



# Universidad Nacional del Sur

TESIS DE DOCTORADO EN CIENCIAS DE LA COMPUTACIÓN

*Estrategias de aprendizaje profundo aplicadas al descubrimiento de fármacos: representación molecular, modelado de bioactividad y analítica visual para cribado virtual.*

Vir Sabando

BAHÍA BLANCA

ARGENTINA

2023

## Prefacio

Esta Tesis se presenta como parte de los requisitos para optar al grado Académico de Doctorado en Ciencias de la Computación, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el ámbito del Departamento de Ciencias e Ingeniería de la Computación durante el período comprendido entre el 1 de Abril de 2018 y el 30 de Noviembre de 2023, bajo la dirección del Dr. Ingacio Ponzoni, Profesor Asociado de la Universidad Nacional del Sur e Investigador Principal de CONICET, y del Dr. Axel J. Soto, Profesor Adjunto de la Universidad Nacional del Sur e Investigador Adjunto de CONICET.

.....  
Vir Sabando

`vir.sabando@cs.uns.edu.ar`

Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur  
Bahía Blanca, 30 de Noviembre de 2023



UNIVERSIDAD NACIONAL DEL SUR  
Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el .../.../..., mercediendo la calificación de ..... (.....)

# Agradecimientos

Soy una firme convencida de que todas las empresas y desafíos, sean grandes o pequeños, sólo son posibles de forma colectiva. Estaré eternamente agradecida con muchas personas que me acompañaron y apoyaron durante todo mi trayecto académico y profesional. La realización de esta tesis no hubiera sido posible sin ellos, por lo que también son autores de este trabajo.

En primer lugar, quiero agradecer profundamente a mis directores, Ignacio y Axel, quienes sin dudar confiaron en mí y generosamente compartieron su tiempo, experiencia y conocimiento, permitiéndome cumplir con un sueño como lo fue el desarrollo de este doctorado. Quiero también agradecer muy especialmente al Dr. Evangelos Milios, cuya colaboración y apoyo me permitieron expandir mis horizontes de investigación y resultaron fundamentales en la culminación de este trabajo, y a la Dra. Ana Maguitman, quien desde 2011 cuando aún cursaba mis estudios de grado me inspiró para emprender el apasionante camino de la investigación. A todos, quiero agradecerles por su impecable labor profesional y su calidez humana, evidencia irrefutable de que es posible hacer ciencia y generar conocimiento con empatía y solidaridad.

Quiero agradecer muy especialmente a quien a lo largo de toda mi carrera de grado y de posgrado me acompañó, aconsejó e inspiró. Este doctorado es tanto suyo como mío. Para Maiso, mi querido compañero de ruta, mi más eterno cariño y gratitud.

También quiero agradecer a la luz de mis ojos, mi adorado Viejo, quien todos los días de mi vida me acompaña y alienta a ser una mejor persona, con su inagotable amor y su admirable temple, ejemplo de lucha y resiliencia. Ansío emprender nuevos caminos y desafíos de tu mano.

Quiero agradecer a mis amigos trolos y compañeres militantes de Mala Junta, Soberana, Casa del Pueblo, La Viejo Trolo, y mis queridos gurises del Merendero Santa María del Camino, que han sido el motor en los momentos más difíciles, me han dado la entereza para continuar y para siempre recordar que hacer investigación en la universidad pública es hacer Patria. También, a mis amigos y compañeres de trabajo, con quienes he compartido innumerables momentos de disfrute, así como

de debate y discusión, que han contribuido a la mejora continua de esta tesis. Y, por supuesto, a mi familia, por su acompañamiento, paciencia y cariño de siempre.

Le debo un agradecimiento especial al Departamento de Ciencias e Ingeniería de la Computación y a todos sus integrantes, que siempre han velado por mi bienestar y hace 12 años me hacen sentir parte de una familia, y al Consejo Nacional de Investigaciones Científicas y Técnicas, por el otorgamiento de la beca doctoral que me ha permitido llevar a cabo mis estudios de posgrado y por poner a la ciencia e investigación públicas como eje central y política de estado en un proyecto grande de soberanía nacional.

Finalmente, quiero agradecer por la educación pública, a la que considero un derecho humano inalienable, una herramienta para transformar la realidad, y el primordial motor de movilidad ascendente, que le ha permitido a este humilde chique de clase baja formarse y soñar con un proyecto de vida improbable en cualquier otro contexto imaginable. En tiempos oscuros donde los derechos humanos más básicos se ponen en cuestionamiento, no hay lugar para tibiezas ni silencios. Quienes luchamos todos los días para existir lo sabemos mejor que nadie. Seguiremos resistiendo y construyendo para alcanzar una Patria en la que nadie se quede afuera.

Vir Sabando

Bahía Blanca, 16 de Noviembre de 2023

Para Diana, Lohana, Tehuel,  
y todas las víctimas de travesticidio social.

# Resumen

El desarrollo de nuevos fármacos constituye un área de investigación fundamental en la medicina moderna. Más allá de los vertiginosos avances científicos en informática molecular y bioquímica que abonan a su mejora continua, la inversión en tiempo y recursos es sumamente elevada, en contraste a su exigua tasa de éxito. Las estrategias computacionales juegan un rol clave en la optimización y eficiencia de las múltiples tareas involucradas en el desarrollo de medicamentos, que abarcan desde la representación molecular y el modelado predictivo de bioactividad hasta el cribado virtual de fármacos y el diseño de nuevas estructuras químicas.

El objetivo de esta tesis se centró en el desarrollo y aplicación de estrategias computacionales novedosas basadas en aprendizaje profundo para contribuir a la optimización de las diversas etapas del descubrimiento de nuevos medicamentos. Las contribuciones de la presente tesis parten de un análisis crítico y permanente del estado del arte en informática molecular e involucran el diseño de nuevas estrategias aplicando conceptos y desarrollos de vanguardia en aprendizaje profundo. Como resultado de este trabajo, se lograron propuestas novedosas alineadas en tres ejes fundamentales del proceso de desarrollo de fármacos: representaciones moleculares, modelado predictivo de bioactividad, y analítica visual aplicada a cribado virtual de fármacos.

En materia de modelado predictivo de bioactividad, desarrollamos enfoques de modelado QSAR capaces de alcanzar rendimientos predictivos superiores a los previamente reportados para un gran número de propiedades de relevancia en el área, sin necesidad de realizar selección de características. Propusimos un enfoque de definición del dominio de aplicabilidad químico para dichos modelos eficaz en la determinación del rango de confiabilidad de las predicciones, y desarrollamos una estrategia para brindar interpretabilidad a modelos QSAR basados en redes neuronales. Además, experimentamos con aprendizaje profundo multi-tarea, logrando un enfoque pionero para el modelado de mutagenicidad de Ames, que permite el aprendizaje conjunto de información de diferentes blancos farmacológicos, superando en rendimiento a los resultados previamente publicados.

En el área de representación molecular, desarrollamos un riguroso trabajo de investigación y análisis comparativo de diversas estrategias de representación molecular tradicionales y basadas en aprendizaje profundo. Propusimos un diseño experimental para la comparación y evaluación del desempeño de dichas representaciones en modelado QSAR, cuyos resultados evidenciaron la importancia de la selección cuidadosa de la representación elegida y proporcionan un marco de referencia para posteriores estudios similares. Por último, presentamos una herramienta integral de analítica visual para cribado virtual que integra diferentes fuentes de información química y representaciones moleculares complementarias. Esta herramienta interactiva demostró ser eficaz en la asistencia a expertos de química medicinal para la exploración visual de patrones de similitud estructural en grandes conjuntos de datos químicos y para el diseño de nuevos compuestos candidatos.

# Abstract

The development of new drugs constitutes a fundamental research area in modern medicine. Beyond the rapid scientific advances in molecular informatics and biochemistry, which contribute to its continuous improvement, the investment in time and resources is extremely high, in contrast to its limited success rate. Computational strategies play a key role in optimizing and streamlining the multiple tasks involved in drug development, ranging from molecular representation and predictive modeling of bioactivity profiles, to virtual drug screening and the design of novel chemical structures.

The goal of this thesis focused on the development and application of novel computational strategies based on deep learning to contribute to the optimization of the many stages involved in the drug discovery process. The contributions of this thesis stem from a critical and ongoing analysis of the state of the art in molecular informatics and involve the design of new strategies by applying recent concepts and developments in deep learning. As a result of this work, we achieved a series of innovative proposals which align to three fundamental cornerstones of the drug development process: molecular representation, predictive modeling of bioactivity profiles, and visual analytics applied to virtual drug screening.

In the field of predictive bioactivity modeling, we developed QSAR modeling approaches that achieved higher predictive performances than those previously reported for numerous relevant biochemical properties, while at the same time overcoming the need for a feature selection step. We proposed an approach to define the chemical applicability domain for these models, effectively determining the reliability range of predictions, and developed a strategy to provide interpretability to QSAR models based on neural networks. Additionally, we experimented with multi-task deep learning, achieving a pioneering approach for modeling Ames mutagenicity that allows the joint learning of information from different pharmacological targets, which outperformed previously published results.

In the field of molecular representation, we conducted a rigorous research and comparative analysis of various traditional and deep learning-based molecular representation strategies. We proposed an



experimental design for the comparison and evaluation of the performance of these representations in QSAR modeling, and the results highlighted the importance of carefully selecting the molecular representation for each task, while also providing a reference framework for subsequent similar studies. Finally, we introduced a comprehensive visual analytics tool for virtual screening that integrates different sources of chemical information and complementary molecular representations. This interactive tool proved to be effective in assisting medicinal chemistry experts in visually exploring structural similarity patterns in large chemical datasets and in the design of new candidate compounds.

# Publicaciones

## Publicaciones en revistas científicas

- Sabando, M.V., Ponzoni, I., Soto, A.J., 2019. *Neural-based approaches to overcome feature selection and applicability domain in drug-related property prediction*. **Applied Soft Computing**, 85, p.105777, doi: <https://doi.org/10.1016/j.asoc.2019.105777>.
- Sabando, M.V., Ulbrich, P., Selzer, M., Byška, J., Mičan, J., Ponzoni, I., Soto, A.J., Ganuza, M.L., Kozlíková, B., 2021. *ChemVA: Interactive Visual Analysis of Chemical Compound Similarity in Virtual Screening*. **IEEE Transactions on Visualization and Computer Graphics**, vol. 27, no. 2, pp. 891-901, doi: <https://doi.org/10.1109/TVCG.2020.3030438>.
- Sabando, M.V., Ponzoni, I., Milios, E.E., Soto, A.J., 2022. *Using molecular embeddings in QSAR modeling: does it make a difference?*. **Briefings in Bioinformatics**, 23(1), p.bbab365, doi: <https://doi.org/10.1093/bib/bbab365>.
- Martínez, M.J.\*, Sabando, M.V.\*, Soto, A.J., Roca, C., Requena-Triguero, C., Campillo, N.E., Páez, J.A., Ponzoni, I., 2022. *Multitask Deep Neural Networks for Ames Mutagenicity Prediction*. **Journal of Chemical Information and Modeling**, 62(24), pp.6342-6351, doi: <https://doi.org/10.1021/acs.jcim.2c00532>.

(\*) Contribuyeron equitativamente al trabajo de investigación.

## Premios

- Sabando M.V., Soto A.J., *Learning Molecular Embeddings for Drug Repurposing*. **Google Latin America Research Awards (LARA) ediciones 2020<sup>1</sup> y 2021<sup>2,3</sup>**.

---

<sup>1</sup>Anuncio Google LARA 2020: <https://blog.google/intl/es-419/noticias-de-la-empresa/de-google/estos-son-los-ganadores-de-la-edicion/>

<sup>2</sup>Anuncio Google LARA 2021 (AR): <https://blog.google/intl/es-419/noticias-de-la-empresa/de-google/premios-lara-novena-edicion-estos-son-los-24-ganadores/>

<sup>3</sup>Anuncio Google LARA 2021 (BR): <https://blog.google/intl/pt-br/novidades/iniciativas/conheca-os-vencedores-do-premio-lara-2021-o-programa-de-bolsas-de-pesquisa-do-google/>

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Estrategias computacionales aplicadas al desarrollo de fármacos . . . . .	1
1.1.1. Desafíos del desarrollo de fármacos asistido por computadora . . . . .	2
1.2. Objetivos y alcance . . . . .	5
1.3. Organización de la tesis . . . . .	6
<b>2. Aprendizaje automático y profundo</b>	<b>8</b>
2.1. Introducción . . . . .	8
2.2. Aprendizaje automático . . . . .	9
2.2.1. Aprendizaje supervisado y no supervisado . . . . .	9
2.2.2. Modelado predictivo . . . . .	10
2.2.3. Modelos de clasificación y regresión . . . . .	13
2.2.4. Modelos de <i>clustering</i> y reducción dimensional . . . . .	14
2.2.5. Algoritmos clásicos de modelado predictivo . . . . .	16
2.3. Particionado de los datos . . . . .	19
2.4. Dimensionalidad de los datos . . . . .	21
2.5. Aprendizaje profundo . . . . .	22
2.5.1. Introducción a las Redes Neuronales Profundas ( <i>Deep Neural Networks</i> ) . . . . .	22
2.5.2. Técnicas empleadas en el entrenamiento de modelos basados en aprendizaje profundo . . . . .	25

2.5.3. Arquitecturas profundas . . . . .	36
2.6. Interpretabilidad y explicabilidad de modelos predictivos . . . . .	48
2.7. Métricas de rendimiento . . . . .	51
2.8. Síntesis y conclusiones . . . . .	55
<b>3. Informática molecular</b>	<b>56</b>
3.1. Introducción . . . . .	56
3.2. Conceptos básicos de química medicinal . . . . .	59
3.2.1. Propiedades farmacocinéticas y farmacodinámicas . . . . .	60
3.2.2. <i>Drug-likeness</i> . . . . .	62
3.2.3. Principio de similitud estructural . . . . .	64
3.3. Informática molecular y diseño de fármacos asistido por computadora . . . . .	65
3.3.1. Representación computacional de compuestos químicos . . . . .	65
3.3.2. Modelado QSAR . . . . .	79
3.3.3. Cribado virtual de fármacos . . . . .	83
3.3.4. Analítica visual en informática molecular . . . . .	86
3.3.5. Diseño <i>de Novo</i> . . . . .	88
3.4. Síntesis y conclusiones . . . . .	90
<b>4. Enfoque integral de modelado QSAR empleando estrategias de aprendizaje profundo</b>	<b>92</b>
4.1. Introducción . . . . .	92
4.2. Enfoque integral para modelado QSAR basado en aprendizaje profundo . . . . .	95
4.3. Trabajo relacionado con la propuesta . . . . .	96
4.4. Nuestra propuesta . . . . .	99
4.5. Casos de estudio, metodología y diseño experimental . . . . .	100
4.5.1. Conjuntos de datos . . . . .	101

4.5.2.	Construcción de los modelos QSAR . . . . .	102
4.5.3.	Dominio de aplicabilidad . . . . .	103
4.5.4.	Análisis de características moleculares <i>a posteriori</i> . . . . .	105
4.6.	Resultados obtenidos . . . . .	107
4.6.1.	Resultados de la técnica de dominio de aplicabilidad integrada . . . . .	110
4.6.2.	Resultados del análisis de características moleculares <i>a posteriori</i> . . . . .	116
4.7.	Síntesis y conclusiones . . . . .	119
<b>5.</b>	<b>Modelado QSAR empleando aprendizaje multi-tarea (MTL)</b>	<b>121</b>
5.1.	Introducción a la mutagenicidad de Ames . . . . .	121
5.2.	Modelado QSAR de la prueba de mutagenicidad de Ames . . . . .	123
5.3.	Trabajo relacionado con la propuesta . . . . .	125
5.4.	Metodología y conjunto de datos empleado . . . . .	128
5.4.1.	Sanitización y etiquetado del conjunto de datos . . . . .	128
5.4.2.	Arquitectura de los modelos propuestos . . . . .	131
5.4.3.	Diseño experimental . . . . .	134
5.5.	Resultados obtenidos . . . . .	138
5.5.1.	Resultados obtenidos mediante el enfoque de etiquetado <i>laxo</i> . . . . .	143
5.6.	Síntesis y conclusiones . . . . .	143
<b>6.</b>	<b>Evaluación de representaciones moleculares</b>	<b>146</b>
6.1.	Representaciones moleculares en diseño de fármacos . . . . .	146
6.2.	Trabajo relacionado con la propuesta . . . . .	150
6.3.	Metodología . . . . .	153
6.3.1.	Conjuntos de datos . . . . .	153
6.3.2.	Métodos de representación molecular . . . . .	155
6.3.3.	Diseño experimental . . . . .	160

6.4.	Evaluación del desempeño de las representaciones moleculares en modelado QSAR . . .	167
6.5.	Análisis del estado del arte . . . . .	173
6.6.	Análisis de dispersión de resultados de clasificación . . . . .	177
6.7.	Síntesis y conclusiones . . . . .	179
<b>7.</b>	<b>Analítica visual aplicada a cribado virtual</b>	<b>181</b>
7.1.	Introducción . . . . .	181
7.2.	Trabajo relacionado con la propuesta . . . . .	184
7.2.1.	Exploración visual de datos multi-dimensionales . . . . .	184
7.2.2.	Visualización de estructuras moleculares . . . . .	186
7.3.	Marco teórico . . . . .	188
7.3.1.	Representaciones moleculares basadas en vectores . . . . .	189
7.3.2.	Características moleculares relacionadas con <i>drug-likeness</i> . . . . .	190
7.4.	Requisitos de nuestra herramienta de analítica visual . . . . .	190
7.5.	Diseño e implementación de <i>ChemVA</i> . . . . .	193
7.5.1.	Diseño general y esquema de distribución de vistas . . . . .	194
7.5.2.	Vistas 2D . . . . .	195
7.5.3.	Vista Tabular . . . . .	202
7.5.4.	Vista 3D . . . . .	204
7.5.5.	Reducción dimensional paramétrica . . . . .	206
7.5.6.	Cómputo de puntajes de correlación . . . . .	207
7.6.	Soporte para diseño <i>de novo</i> . . . . .	209
7.7.	Evaluación de <i>ChemVA</i> y casos de estudio . . . . .	210
7.7.1.	Primera etapa: Casos de estudio . . . . .	210
7.7.2.	Segunda etapa: Evaluación cualitativa de la herramienta por expertos externos	216
7.8.	Resultados y discusión . . . . .	218
7.9.	Síntesis y conclusiones . . . . .	220

<b>8. Conclusiones y trabajo futuro</b>	<b>222</b>
8.1. Resumen de las contribuciones realizadas . . . . .	224
8.2. Trabajo futuro . . . . .	227
<b>A. Hiperparámetros evaluados durante las etapas de selección de modelos en la evaluación de representaciones moleculares (capítulo 6)</b>	<b>229</b>
<b>B. Listado de abreviaciones</b>	<b>232</b>



# Capítulo 1

## Introducción

En este capítulo presentamos de manera sintética la temática abordada por esta tesis doctoral, exponiendo los argumentos fundamentales que sustentan este trabajo de investigación. El propósito de esta sección es proporcionar a lx lectorx una visión general de los contenidos desarrollados a lo largo de sus ocho capítulos y establecer las bases necesarias para su comprensión. Inicialmente, planteamos las problemáticas que motivaron la tesis, para luego enumerar los objetivos generales y específicos perseguidos por la misma. Finalmente, describimos de forma concisa la organización y el contenido de cada uno de los capítulos que conforman el presente trabajo.

---

### 1.1. Estrategias computacionales aplicadas al desarrollo de fármacos

La informática molecular aborda el estudio y desarrollo de métodos computacionales aplicados al análisis de los fenómenos y principios que rigen la interacción de las moléculas. Su aplicación abarca múltiple áreas, desde el diseño racional de fármacos [110, 175] hasta el diseño y estudio de materiales sintéticos y poliméricos para uso industrial y médico [4]. En particular, la informática molecular aplicada al desarrollo de medicamentos constituye un área interdisciplinaria que integra conocimiento experto de ciencias de la computación, bioquímica y medicina, orientada al estudio y modelado de procesos bioquímicos y biológicos a nivel molecular. Desde sus inicios a mediados del siglo pasado [422], esta disciplina se ha transformado significativamente, incorporando avances novedosos en la investigación médica y farmacológica, así como también estrategias de aprendizaje automático e inteligencia artificial [100, 237, 193, 257].

La industria del diseño de fármacos desempeña un papel crucial en el desarrollo global, entrelazando la innovación científica con aspectos comerciales y regulatorios para la introducción de nuevas terapias y tratamientos. La industria farmacéutica se caracteriza por ser de alto impacto y alto riesgo, donde la inversión promedio en el desarrollo de un nuevo medicamento es de miles de millones de dólares y puede llevar más de una década, con una tasa de éxito muy baja. Por lo general, el proceso se inicia con una reserva de compuestos candidatos que supera las decenas de miles de moléculas, de las cuales prosperan solo unas pocas para las etapas finales de ensayos clínicos [90, 362]. El desarrollo de medicamentos constituye un desafío de alta complejidad y múltiples fases, resultando fundamental la aplicación de estrategias computacionales en etapas tempranas del proceso. El diseño de fármacos asistido por computadora permite lograr selecciones precisas y eficientes de compuestos candidatos, reducir el tiempo y los recursos necesarios, y ampliar y diversificar las reservas de compuestos factibles en el proceso de desarrollo de un nuevo fármaco [64, 181, 237, 90, 362].

En la actualidad, existen múltiples tareas en el desarrollo de un nuevo medicamento que involucran estrategias computacionales. Entre ellas, encontramos desde la creación de nuevos algoritmos para el cómputo de representaciones moleculares; los enfoques de simulación de interacciones moleculares; el modelado predictivo de la actividad biológica de compuestos candidatos; la incorporación de técnicas de analítica visual para la exploración de grandes quimiotecas; incluso desafíos ambiciosos como el diseño de nuevas estructuras moleculares [110, 175]. Estas estrategias han mejorado sistemáticamente la probabilidad de éxito en la identificación de nuevos fármacos y han allanado el camino hacia terapias más efectivas y personalizadas.

### 1.1.1. Desafíos del desarrollo de fármacos asistido por computadora

Durante los últimos años ha habido avances tecnológicos significativos en el campo del aprendizaje profundo y la inteligencia artificial, manifestados en una proliferación de modelos y arquitecturas basadas en redes neuronales profundas (DNNs, por sus siglas en inglés) en los más diversos dominios. El auge de los modelos generativos y de los modelos empleados en el dominio del procesamiento de lenguaje natural [148, 382, 327] han dado lugar a un nuevo paradigma de diseño de algoritmos computacionales potentes y versátiles. La flexibilidad de estos enfoques los vuelve propicios para su aplicación en diversas tareas del proceso de desarrollo de medicamentos, lo cual resulta evidente por cuanto nuevas publicaciones sobre la temática surgen cada año, incorporando cada vez más elementos de estos modelos de aprendizaje avanzados [288, 26]. Un ejemplo claro de este fenómeno es el surgimiento de numerosos algoritmos para obtener representaciones moleculares por medio de

estrategias de aprendizaje profundo, las cuales pueden usarse luego en múltiples tareas del proceso de diseño de fármacos [103, 73, 421, 26].

Un desafío interesante con respecto a esta temática radica en la selección del método apropiado para obtener estas representaciones, denominadas *embeddings* moleculares, puesto que un análisis comparativo de distintos *embeddings* requiere un diseño experimental riguroso que tenga en cuenta múltiples escenarios de modelado y la complejidad de las diferentes metodologías en términos de su entrenamiento. El análisis de representaciones moleculares resulta indispensable para garantizar resultados confiables en etapas posteriores del diseño de fármacos y para establecer criterios de selección que permitan explotar su potencial. Previo al desarrollo de esta tesis, y a pesar del evidente interés de la comunidad científica en la temática, no encontramos publicaciones que establecieran un diseño experimental riguroso para la comparación de *embeddings* moleculares en el contexto de modelado QSAR. Asimismo, la mayoría de los trabajos que los emplean lo hacen sin encontrar evidencia contundente de que el desempeño predictivo de los modelos QSAR desarrollados sea superior que al emplear representaciones tradicionales, como descriptores moleculares [182, 176, 124, 122, 440, 439, 69].

Los algoritmos de aprendizaje profundo permiten trabajar con conjuntos de datos químicos de gran magnitud, en espacios de alta dimensionalidad, y modelar relaciones complejas y no lineales entre las características moleculares [100, 237, 193, 257]. Como contrapartida, presentan desafíos relacionados con la interpretabilidad de los modelos, a su capacidad de generalización y a la necesidad de un riguroso proceso de selección de modelos para obtener buenos resultados. Una de las tareas más importantes es la identificación temprana de compuestos candidatos, caracterizados por exhibir propiedades físico-químicas asociadas a un cierto perfil de actividad biológica deseado. El modelado de *Relación Cuantitativa de Estructura-Actividad* (QSAR, por sus siglas en inglés) es una estrategia ampliamente establecida y estudiada en el área, que constituye un pilar en la predicción de propiedades farmacológicas de las moléculas evaluadas [316, 345].

Para entrenar un modelo QSAR, los compuestos químicos suelen representarse en un espacio de alta dimensionalidad, donde cada atributo o característica molecular condensa información estructural y química de los mismos. Generalmente, se debe aplicar un proceso de selección de características, que reduzca el espacio dimensional de representación molecular. Sin embargo, esto implica la necesidad de vasto conocimiento experto, lidiar con la complejidad combinatorial de múltiples selecciones posibles de características, y con el potencial de pérdida de información relevante para el modelado [222], por lo que resulta necesario encontrar estrategias de modelado que puedan lidiar con representaciones moleculares de alta dimensionalidad. Por otra parte, la determinación del

subespacio químico en el cual el modelo QSAR obtenido realiza predicciones confiables, denominado *dominio de aplicabilidad*, no resulta trivial dada la amplitud y diversidad del espacio químico, potencialmente infinito. Estos desafíos se suman a la necesidad de lograr modelos QSAR con un cierto grado de interpretabilidad, ya que son empleados por expertos en el área para la toma de decisiones críticas en el proceso de desarrollo de fármacos. Todos estos desafíos derivan en la necesidad de intersectar múltiples técnicas computacionales y estadísticas para lograr modelos QSAR eficaces y precisos.

Otra arista a considerar en el estudio del modelado QSAR es la complejidad de las interacciones moleculares en los organismos vivos, tema de investigación abierto y en constante desarrollo. Se sabe que un fármaco, una vez administrado e ingresado al torrente sanguíneo, tiene la capacidad de interactuar con múltiples blancos terapéuticos a la vez. Este tipo de comportamiento debe ser estudiado, puesto que acarrea el potencial de generar efectos adversos, de alterar el metabolismo o la excreción de otros fármacos, y de producir interacciones medicamentosas no esperadas [53]. Más aún, existen blancos farmacológicos que interactúan entre sí, cuya activación o inhibición farmacológica está correlacionada y, por ende, debe ser estudiada de forma conjunta y complementaria [12, 138]. La mayoría de las estrategias de modelado QSAR publicadas no prevén mecanismos para incorporar información de múltiples blancos biológicos en la predicción del perfil de bioactividad de un compuesto candidato, lo cual constituye un tópico de investigación prometedor con desafíos interesantes.

Más allá del desarrollo de modelos *in silico* para la predicción de la actividad biológica de un conjunto de compuestos candidatos, los expertos desarrollan exploraciones minuciosas en espacios químicos vastos y diversos, en búsqueda de candidatos prometedores. Estas búsquedas conllevan el análisis de similitud estructural, de similitud en términos de propiedades físico-químicas y de perfiles de bioactividad específicos, entre otros [332, 368, 236]. Este análisis presenta varios desafíos, muchos de ellos asociados al gran volumen de datos químicos a explorar y su alta dimensionalidad. Generalmente, se apela a estrategias de reducción dimensional que permitan procesar de forma eficiente el espacio a explorar; sin embargo, el cómputo subyacente a estas estrategias suele causar distorsiones en el espacio molecular y en las relaciones espaciales entre compuestos, lo que puede llevar a realizar análisis de similitud erróneos o sesgados [21, 80, 294]. En este contexto, resulta fundamental desarrollar estrategias eficientes y eficaces para asistir a expertos en química medicinal, que idealmente integren múltiples fuentes de información química complementarias y faciliten el análisis y la interpretación de las relaciones moleculares en grandes conjuntos de compuestos químicos.

## 1.2. Objetivos y alcance

El trabajo realizado en el marco de esta tesis doctoral se centra en el desarrollo y aplicación de estrategias computacionales basadas en aprendizaje automático y profundo en diferentes etapas del proceso de desarrollo de fármacos. Durante la evolución de esta tesis estudiamos constantemente el estado del arte y analizamos de forma crítica y situada las tendencias prominentes en el área, las cuales naturalmente fueron guiando el proceso, objetivos y preguntas de investigación abordadas por la misma, y las contribuciones realizadas en esta tesis parten de la identificación de limitaciones y desafíos enfrentados por los métodos propuestos.

El modelado QSAR es la subdisciplina de informática molecular en la que más enfáticamente se enmarca la presente tesis. En esta área, planteamos preguntas de investigación vinculadas al desarrollo de modelos predictivos precisos, contando con representaciones moleculares de alta dimensionalidad como datos de entrada, con el objetivo de determinar si, contrariamente a la tendencia contemporánea, evitar un proceso de selección de características moleculares impactaría positivamente en los resultados del modelo. En línea con esta hipótesis, exploramos el impacto de diferentes representaciones moleculares, tradicionales y obtenidas por medio de algoritmos de aprendizaje profundo, en el desempeño de modelos QSAR para diferentes propiedades relevantes al ámbito de la química medicinal, haciendo énfasis en la rigurosidad y diversidad del diseño experimental y del análisis comparativo.

También en el espectro del modelado QSAR, exploramos estrategias de modelado basadas en redes neuronales profundas, siendo su falta de interpretabilidad el desafío más frecuentemente identificado en la literatura del área. En este sentido, planteamos preguntas de investigación orientadas a dilucidar si es posible otorgar un sentido de interpretabilidad a un modelo QSAR basado en redes neuronales, con el objetivo particular de elaborar un enfoque de interpretación en función del impacto de cada una de las características moleculares empleadas como atributos descriptores de los compuestos de entrada. Además, formulamos hipótesis de investigación asociadas a la determinación precisa del dominio de aplicabilidad de un modelo, con el objetivo de cuantificar su impacto en la medición del desempeño y confiabilidad de los resultados obtenidos. Por último, exploramos el desarrollo de estrategias de modelado QSAR multi-tarea, donde el objetivo a predecir no esté determinado por un único valor de bioactividad para cada compuesto, con el objetivo de cuantificar el efecto del aprendizaje conjunto de múltiples tareas de predicción por medio de un modelo basado en aprendizaje profundo.

Por fuera de los desafíos vinculados al modelado QSAR, analizando el estado del arte, identificamos las carencias y limitaciones en la bibliografía en términos de herramientas integrales de asistencia a expertxs para el análisis de grandes conjuntos de datos químicos. En línea con la tendencia contemporánea de la proliferación de métodos de representación molecular basados en aprendizaje profundo, formulamos preguntas de investigación asociadas al impacto de las mismas en la identificación de patrones de similitud estructural entre compuestos, necesaria para guiar el proceso de diseño de nuevos fármacos. En este sentido, y persiguiendo el objetivo de vincular diferentes fuentes de información molecular de forma holística, desarrollamos una herramienta integral de analítica visual para cribado virtual de fármacos que proporciona funcionalidades de avanzada para la comparación y visualización de estructuras moleculares. Siguiendo el objetivo de medir el impacto de la selección de las representaciones moleculares para diferentes tareas en el desarrollo de medicamentos, identificamos la necesidad de un medio de estimación de la confiabilidad de dichas representaciones. Por lo tanto, incorporamos vistas novedosas específicamente diseñadas para contrastar las proyecciones visuales obtenidas de conjuntos de datos químicos expresados por medio de diferentes representaciones moleculares.

Los tópicos abordados por la presente tesis concluyen en desarrollos de carácter interdisciplinario. Las propuestas devenidas de este trabajo contribuyen a mitigar falencias y mejorar la eficacia y eficiencia de diferentes tareas en el complejo proceso de desarrollo de fármacos y han sido rigurosamente validadas por expertxs en el área. En términos metodológicos y algorítmicos, las contribuciones realizadas están fuertemente ancladas en el aprendizaje automático y profundo, proponiendo estrategias novedosas basadas tanto en principios clásicos como de vanguardia mundial en ciencias de la computación.

### 1.3. Organización de la tesis

La presente tesis está organizada en ocho capítulos. En el primer capítulo se brinda una introducción a las temáticas, objetivos y alcances abordados por la tesis. El capítulo 2 presenta de forma sintética los conceptos principales de aprendizaje automático y aprendizaje profundo que dan sustento al desarrollo experimental de la tesis. En el capítulo 3 se brindan las definiciones y explicaciones de los conceptos de informática molecular más relevantes a la presente tesis, orientados a lx lectorx no familiarizadx con el dominio. Luego, en los capítulos 4, 5, 6 y 7 se detallan los aportes realizados en el marco de esta tesis. En el capítulo 4 se propone un enfoque de modelado QSAR integral basado en redes neuronales profundas que integra una estrategia de definición de dominio de

aplicabilidad y de interpretabilidad *a posteriori*. El capítulo 5 presenta una estrategia de modelado QSAR multi-tarea basada en redes neuronales para predecir mutagenicidad, por medio de la cual es posible modelar perfiles de bioactividad complementarios en simultáneo. El capítulo 6 aborda un análisis comparativo de diferentes métodos de representación molecular y su desempeño en el contexto del modelado QSAR para ocho propiedades físico-químicas. En el capítulo 7 se presenta *ChemVA*, una herramienta de analítica visual diseñada para brindar soporte a expertos permitiendo la exploración interactiva de grandes conjuntos de compuestos químicos y la comparación visual de múltiples representaciones moleculares. Finalmente, el capítulo 8 resume los principales aportes y conclusiones de la presente tesis y traza algunas posibles líneas de trabajo futuro asociadas a la misma.

# Capítulo 2

## Aprendizaje automático y profundo

En este capítulo presentamos una introducción a los conceptos básicos de aprendizaje automático, para luego desarrollar en particular los conceptos teóricos relacionados a aprendizaje profundo. El marco teórico sobre aprendizaje automático y profundo presentado en este capítulo busca abonar a una comprensión en alto nivel de abstracción del uso de dichas estrategias, en las cuales se basa el desarrollo experimental y metodológico de esta tesis. Para cada uno de los temas abordados en este capítulo, proporcionamos referencias bibliográficas a las cuales el lector puede remitirse a fin de profundizar los conceptos.

---

### 2.1. Introducción

El *Aprendizaje Automático* (conocido como *Machine Learning* en inglés) y el *Aprendizaje Profundo* (*Deep Learning* en inglés) son dos campos de la inteligencia artificial que han revolucionado la forma en la que abordamos problemas de clasificación, predicción y toma de decisiones en una amplia gama de aplicaciones [178]. A diferencia de las disciplinas de la inteligencia artificial *simbólica*, que se basan en representaciones de alto nivel de abstracción de los problemas inspiradas en la lógica matemática para la construcción y representación de conocimiento, el aprendizaje automático y profundo se centran en la observación de conjuntos de datos, la construcción de modelos basados en dichos datos y su posterior utilización para la resolución de problemas. Las herramientas computacionales derivadas de estas disciplinas se basan en la creación de algoritmos que permitan automáticamente detectar patrones en los datos y desarrollar estrategias de inferencia a partir de los



mismos, donde la mejora en el rendimiento de dichos modelos se da a partir del entrenamiento y no de una programación explícita o basada en reglas sobre una base de conocimiento preestablecida.

## 2.2. Aprendizaje automático

El aprendizaje automático es una rama de la inteligencia artificial que comprende el conjunto de técnicas computacionales y algoritmos que permiten el aprendizaje de patrones a partir de conjuntos de datos y la toma de decisiones, predicción o inferencia basadas en dichos patrones. El rendimiento de estas técnicas computacionales es, en consecuencia, fuertemente dependiente de la cantidad, calidad y características de los datos empleados en su proceso de entrenamiento.

### 2.2.1. Aprendizaje supervisado y no supervisado

Si bien existen diferentes criterios a partir de los cuales elaborar una taxonomía de las diferentes estrategias computacionales comprendidas en el campo del aprendizaje automático [178], podemos identificar dos enfoques primordiales para el aprendizaje automático en términos de la naturaleza de los datos empleados en el entrenamiento [8, 264]. Por un lado, en el enfoque de aprendizaje *supervisado*, el algoritmo es entrenado utilizando un conjunto de datos *etiquetado*, donde cada instancia o dato de entrada está asociado con una etiqueta o salida conocida. El objetivo del enfoque de aprendizaje supervisado es detectar relaciones entre los datos de entrada y sus correspondientes etiquetas de salida, y construir modelos capaces de generalizar estas relaciones a fin de realizar predicciones precisas de los valores de etiqueta para nuevas instancias no vistas anteriormente. Por otro lado, en el enfoque de aprendizaje *no supervisado* el algoritmo se entrena con datos *no etiquetados* y busca identificar patrones ocultos, relaciones o estructuras dentro de los datos. En este enfoque de aprendizaje, las estrategias se pueden clasificar *a priori* en dos instancias principales: *agrupamiento* (*clustering*) y *reducción de dimensionalidad*. Las estrategias de agrupamiento implican la detección de patrones de similitud entre los datos y la separación de los mismos en grupos no preestablecidos, que emergen del proceso de aprendizaje. Por otro lado, las estrategias de reducción de dimensionalidad se enfocan en simplificar la representación de los datos de entrada preservando sus características esenciales y reduciendo la complejidad al disminuir el número de dimensiones. Existe un híbrido que combina elementos de los enfoques supervisado y no supervisado, denominado aprendizaje *semi-supervisado*, que emplea conjuntos de datos parcialmente etiquetados.

Son dignos de mención dos enfoques de aprendizaje automático que se distinguen de los tradicionales. Por un lado, el aprendizaje *por refuerzo* [187], inspirado en los mecanismos de interacción de los organismos vivos con su entorno. Se basa en tres componentes cruciales: un agente autónomo, su entorno y mecanismos de retroalimentación positiva y negativa. La idea fundamental detrás de este enfoque es que el agente desarrolle mecanismos de inferencia y toma de decisiones óptimas a partir de su estado actual y de la interacción con su entorno y, en función de sus acciones, reciba retroalimentación positiva o negativa. En general, el objetivo del agente es aprender una estrategia óptima que le permita maximizar la recompensa acumulada a largo plazo. Por otro lado, los *algoritmos evolutivos* [400, 417], que aunque no necesariamente clasifican como estrategias de aprendizaje automático, han constituido durante años una de las técnicas computacionales estándar para el cribado virtual de fármacos y la selección de características moleculares en quimioinformática [75], disciplina en la que se enmarca esta tesis doctoral. Los algoritmos evolutivos consisten en estrategias de optimización inspiradas en los complejos procesos de evolución biológica, buscando imitar el proceso de selección natural. En líneas generales, se inicia con una población de soluciones candidatas, las cuales se evalúan mediante una función de aptitud que puede ser mono o multi-objetivo. A partir de esta evaluación, se seleccionan las soluciones más aptas para un paso de reproducción, que involucra operaciones de recombinación y mutación, a fin de generar nuevas soluciones (descendencia) que conformen la población candidata de la siguiente generación. Este proceso se repite durante múltiples generaciones, de forma tal que la población evoluciona en pos de hallar soluciones óptimas para el problema dado.

### 2.2.2. Modelado predictivo

El objetivo principal de las estrategias de aprendizaje automático es el *modelado predictivo*, que consiste en el desarrollo de algoritmos y modelos que permitan la detección de patrones y relaciones intrínsecas al conjunto de datos en estudio y, a partir de dichos patrones, realizar predicciones sobre datos futuros o nunca vistos [209, 405]. El aprendizaje automático aplicado al modelado predictivo constituye un pilar fundamental en el campo de la quimioinformática, siendo instrumental en el desarrollo de modelos para la predicción precisa y eficiente de una amplia gama de propiedades químicas y biológicas [342, 273]. El desarrollo de un modelo predictivo comienza por la identificación de la tarea, propiedad o problema de predicción a modelar. Una vez definido el problema, el proceso general de modelado predictivo involucra una serie de pasos clave:

1. Obtención, preparación y preprocesamiento de los datos [115]: en primera instancia, es necesario recolectar los datos, los cuales pueden provenir de fuentes heterogéneas. La calidad, relevancia e integridad de los datos obtenidos resulta crucial para asegurar la confiabilidad en el modelo futuro, por lo que luego se procede a la preparación y preprocesamiento de los mismos. Esta segunda etapa involucra su limpieza, organización, homogeneización, canonicalización, imputación de valores faltantes, manejo de duplicidades, normalización de datos numéricos y conversión de datos categóricos a un formato adecuado para el modelo, entre otros. Dependiendo del dominio particular, en esta etapa pueden realizarse tareas de sanitización especiales o pasos de ingeniería de atributos.
2. Selección de características (*feature selection*): los datos a emplear en la fase de modelado a menudo están descritos por una serie de características o atributos, los cuales inciden directamente en la capacidad predictiva de los modelos finales. Por lo tanto, la elección cuidadosa de las características relevantes para el problema es un paso importante previo al modelado. Para ello, se emplean estrategias manuales o automáticas de selección de características, las cuales pueden estar basadas en técnicas de aprendizaje automático o análisis estadístico, y se eligen las características más relevantes y significativas para la tarea [399, 221]. Sin embargo, es importante notar que el proceso de selección de características es altamente dependiente del dominio. En el caso del modelado de propiedades físico-químicas, generalmente requiere de labor experta y constituye una tarea ardua que se realiza de forma manual, lo cual sumado a la complejidad en la interpretación semántica de la multiplicidad de características moleculares trae aparejado el potencial de pérdida de atributos relevantes para el modelado [354, 222].
3. Particionado de los datos: esta etapa resulta crítica en el desarrollo de modelos predictivos. Consiste en separar el conjunto de datos disponible en subconjuntos disjuntos que posteriormente se utilizan para entrenar y validar los modelos predictivos. Existen diferentes estrategias de particionado, las cuales se adoptan en función del diseño experimental. Profundizaremos sobre esta etapa en la sección 2.3. En líneas generales, se utiliza una mayor proporción de los datos para entrenar el modelo, mientras que los datos restantes son reservados para instancias posteriores al entrenamiento y son empleados para la validación y reporte de resultados del modelo.
4. Selección de modelo: Teniendo en cuenta las particularidades del problema a resolver y las características propias de los datos a emplear, se selecciona un algoritmo de aprendizaje automático específico. Existen diferentes tipos de algoritmos para estrategias de aprendizaje

supervisado y no supervisado, y diferentes algoritmos supervisados dependiendo de las características de la propiedad a predecir: si se trata de etiquetas categóricas se emplean algoritmos de clasificación, mientras que si se trata de etiquetas de valor numérico continuo se emplean algoritmos de regresión. Brindamos más detalles sobre esta etapa en la sección 2.2.3 del presente capítulo.

5. Entrenamiento del modelo: esta etapa consiste en el ajuste de los parámetros del modelo, que le permite aprender patrones y relaciones en los datos. Para ello se emplea la partición de los datos reservada para entrenamiento, y se realiza un proceso iterativo de ajuste y optimización para minimizar la diferencia entre las predicciones realizadas por el modelo y los valores reales a predecir, en el caso del aprendizaje supervisado, o encontrar patrones de forma autónoma para agrupar los datos de forma significativa, en el caso del aprendizaje no supervisado. Este proceso iterativo permite que el modelo aprenda a generalizar a nuevas instancias de datos nunca vistas. Como mencionamos anteriormente, la adecuada selección del algoritmo, el particionado de los datos, la selección de características y el proceso de selección de hiperparámetros del modelo son factores determinantes en los resultados del proceso de entrenamiento [209, 405]. Previa a esta etapa, es fundamental realizar un correcto diseño experimental que permita evaluar la influencia de diferentes parametrizaciones de los algoritmos empleados, teniendo en cuenta aspectos como el potencial desbalance de clases en conjuntos de datos supervisados, la propensión del algoritmo al sobreajuste (*overfitting*), la estocasticidad del modelo y todo factor que pueda introducir sesgos indeseables en el entrenamiento.
6. Evaluación del modelo: una vez concluida la fase de entrenamiento del modelo, se debe determinar el desempeño del mismo en la tarea de predicción. Esto se hace por medio de diferentes métricas, que evalúan la calidad de las predicciones del modelo. Existen métricas adecuadas para escenarios clasificación, de regresión, y para escenarios no supervisados. Brindamos una definición detallada de las métricas más relevantes a la presente tesis en la sección 2.7. La evaluación del modelo se realiza midiendo su desempeño predictivo en la partición de validación externa, la cual no debe haberse empleado de ninguna forma en el proceso de selección de modelos ni de entrenamiento del modelo final, a fin de garantizar independencia de los resultados y de verificar la capacidad de generalización a datos nuevos del modelo final.
7. Análisis estadístico de los resultados: esta etapa, posterior a la evaluación de los modelos por medio de métricas de rendimiento, involucra la aplicación de técnicas estadísticas para interpretar la confiabilidad de los resultados y su significancia estadística [238]. Entre otras

técnicas, se calculan parámetros estadísticos básicos como media, mediana, desviación estándar e intervalos de confianza, para analizar la dispersión entre los múltiples resultados. Además, se emplean pruebas estadísticas de análisis de la varianza para determinar si existen diferencias significativas entre los resultados de modelos diferentes. Esta etapa es crucial para eliminar el potencial sesgo introducido por estocasticidad en la evaluación de los resultados.

### 2.2.3. Modelos de clasificación y regresión

Como discutimos anteriormente, el aprendizaje supervisado y no supervisado constituyen los dos enfoques más importantes de aprendizaje automático. En el aprendizaje supervisado, los modelos son entrenados con datos etiquetados y, por medio de algoritmos de ajuste y optimización, aprenden a asignar etiquetas a nuevos datos de entrada. En función de la naturaleza de dichas etiquetas, podemos referirnos a dos tipos de modelo generados a partir de este enfoque: modelos de *clasificación* y de *regresión*.

Los modelos de *clasificación* son apropiados para cuando la naturaleza de la etiqueta es categórica, es decir, cuando señala la pertenencia de una instancia a una categoría o clase particular de un conjunto de clases. Estos modelos predicen la pertenencia de una instancia a una de dichas clases o categorías. Dependiendo de la cantidad de clases o categorías, puede tratarse de problemas de clasificación *binaria* (dos clases) o *multi-clase* (más de dos clases). Entre los desafíos propios de los problemas de clasificación se destaca el *desbalance de clases*, que se suscita cuando la cantidad de instancias etiquetadas para una clase supera ampliamente a la cantidad de instancias etiquetada para la o las clases restantes. Existen algunas modalidades de entrenamiento que permiten mitigar los efectos del desbalance de clases en el desempeño predictivo, como el particionado estratificado de los datos o la compensación del desbalance durante el ajuste de los parámetros del modelo por medio de funciones de ponderación.

Entre los algoritmos de aprendizaje automático más comúnmente utilizados para clasificación se encuentran *Naïve Bayes* [416], Árboles de decisión (*Decision Trees*) [313], Bosques aleatorios (*Random Forests*) [79], Máquinas de vectores de soporte (*Support Vector Machines* o *SVM*) [239], *Redes Neuronales* [370] y K vecinos más cercanos (*k-Nearest Neighbors* o *k-NN*) [206]. Brindamos una breve explicación de cada uno de estos algoritmos tradicionales de clasificación en la sección 2.2.5.

Por su parte, los modelos de *regresión* son utilizados cuando la etiqueta asignada a los datos es un valor numérico continuo. Dichos modelos son entrenados para predecir un valor numérico en función

de las características que describen a los datos. Si bien en estos casos no se suscita el problema de desbalance de clases, por no trabajar con instancias etiquetadas de forma categórica, el entrenamiento de los modelos de regresión puede resultar desafiante cuando no se cuenta con suficientes datos para cubrir un rango amplio de valores de salida, por lo que tienden a tener problemas de generalización.

Algunos de los algoritmos de regresión más comúnmente empleados mencionamos los modelos de *Regresión lineal* [385], *Bosques aleatorios (Random Forests)* y *Redes Neuronales*. Estos algoritmos tradicionales de regresión se presentan en la sección 2.2.5.

Dentro del aprendizaje supervisado, existen algunas variantes híbridas entre los modelos de clasificación y regresión. Entre ellas, destacamos los modelos de clasificación *difusos*, en los que en lugar de asignar una etiqueta de clase única a cada instancia, se le asignan etiquetas correspondientes con el grado de pertenencia a cada una de las clases posibles. Generalmente, las etiquetas en cuestión se corresponden con valores en el rango  $[0,1]$ , donde 0 indica que la instancia no pertenece a la clase en cuestión en lo absoluto y 1 indica que pertenece a la clase completamente [13]. Otra variante híbrida entre los modelos de clasificación y regresión son los modelos de *regresión ordinal*, adecuados para aquellos escenarios en los que se cuenta con etiquetas categóricas, propias de los problemas de clasificación, pero ordenadas siguiendo algún criterio [141].

#### 2.2.4. Modelos de *clustering* y reducción dimensional

En el contexto del aprendizaje automático no supervisado, los modelos son entrenados a partir de conjuntos de datos no etiquetados y su objetivo es identificar patrones intrínsecos al conjunto de datos y agruparlos a partir de dichos patrones [312, 285]. Entre estos algoritmos, destacan los modelos de *clustering* o agrupamiento, que permiten descubrir estructuras y relaciones subyacentes en conjuntos de datos, y los modelos de *reducción dimensional*, que buscan representar las instancias del conjunto de datos en un espacio de atributos o características inferidas de menor dimensionalidad que el espacio original, buscando preservar la mayor cantidad posible de información relevante.

Los modelos de *clustering* buscan agrupar instancias en conjuntos, también conocidos como clústeres, de acuerdo con la similitud entre ellas. La similitud entre instancias es computada a partir de la exploración automática del espacio de características y atributos, por medio de métricas de similitud o distancia. El objetivo de estos algoritmos es lograr que las instancias dentro de un mismo clúster sean más similares entre sí que con los de otros clústeres, según el criterio de optimización adoptado y la o las métricas de similitud y distancia empleadas. Generalmente, los clústeres son ajustados de forma iterativa por el algoritmo hasta converger a una configuración donde la asignación

de instancias a agrupamientos sea óptima en términos de los criterios de similitud dentro de los clústeres y de disimilitud entre clústeres [312, 285].

Entre los desafíos a abordar en la aplicación de estos algoritmos se incluyen la determinación del número óptimo de clústeres entre los que separar las instancias del conjunto de datos ( $k$ ) y su sensibilidad a la inicialización, por cuanto distintas selecciones iniciales de los centroides de los clústeres pueden derivar en resultados de agrupamiento significativamente diferentes. Por otra parte, algunos algoritmos exhiben dificultades para delimitar clústeres de formas irregulares, no convexos, o en escenarios de densidades significativamente diferentes. Entre los algoritmos de *clustering* más comúnmente utilizados se destacan *K-Means* [7], *DBSCAN (Density-Based Spatial Clustering of Applications with Noise)* [143] y *clustering jerárquico (Hierarchical Clustering)* [274].

Por su parte, los algoritmos de *reducción dimensional* son técnicas empleadas para disminuir la cantidad de características empleadas para representar a un conjunto de datos, buscando preservar la mayor cantidad posible de información relevante. Estos modelos son empleados para mejorar la eficiencia en el procesamiento de los datos, y son fundamentales en tareas como el análisis exploratorio de conjuntos de datos, visualización y clasificación. Los modelos de reducción dimensional transforman el conjunto de datos original, representado por medio de un conjunto de características potencialmente grande, a un conjunto de datos donde cada instancia es expresada por medio de una representación de baja dimensionalidad. Estas representaciones buscan conservar las relaciones esenciales entre las instancias del conjunto, lo cual se logra mediante proyecciones y transformaciones lineales o no lineales que mantienen la estructura y dimensiones de varianza significativa de los datos en menos dimensiones [394].

Uno de los riesgos de aplicar estrategias de reducción dimensional es el potencial de pérdida de información relevante durante la reducción, dado que esto impacta negativamente en el rendimiento de los modelos posteriormente entrenados a partir de tales representaciones [354]. Otros aspectos a tener en cuenta son la elección adecuada del número de dimensiones finales y la correcta configuración de hiperparámetros de dichos algoritmos. Entre las técnicas más prominentes y comúnmente empleadas para reducción dimensional, destacan el Análisis de Componentes Principales (*Principal Component Analysis* o *PCA*) [3], *t-distributed Stochastic Neighbor Embedding (t-SNE)* [393] y los *autoencoders* [297, 29]. Estos últimos, en particular, se enmarcan dentro de las técnicas de aprendizaje profundo.

### 2.2.5. Algoritmos clásicos de modelado predictivo

En esta sección brindamos una breve descripción funcional y matemática de algunos de los algoritmos clásicos de aprendizaje automático supervisado, tanto para tareas de clasificación como regresión. Más allá de que el aprendizaje profundo y los modelos basados en redes neuronales han experimentado un auge significativo en los últimos años, los algoritmos clásicos han constituido pilares fundamentales en el modelado predictivo y siguen siendo ampliamente utilizados y considerados como puntos de referencia [8]. En general, su uso está establecido en la comunidad científica debido a su confiabilidad, interpretabilidad y su relativa simplicidad en contraste con la complejidad inherente a los modelos basados en aprendizaje profundo. En los últimos años, dada la capacidad y el poder predictivo de los modelos basados en aprendizaje profundo, estos algoritmos clásicos aún sientan bases sólidas para comparar y evaluar el desempeño de modelos predictivos, abonando a una mejor comprensión de las ventajas y limitaciones de distintas estrategias de modelado [18].

Más allá de que la amplia mayoría de los modelos predictivos y de reducción dimensional desarrollados en el contexto de la presente tesis se basan primordialmente en arquitecturas de redes neuronales de diversa complejidad, varios de los algoritmos clásicos de clasificación y regresión aquí descritos fueron empleados en nuestro desarrollo experimental. La selección y aplicación específica de estos algoritmos en cada uno de nuestros trabajos se basó en las estrategias de modelado empleadas en la literatura de referencia para los diversos casos de estudio y en los objetivos de investigación particulares a cada trabajo.

#### 2.2.5.1. Regresión lineal

La *regresión lineal* [385] es un modelo matemático de regresión que permite describir la relación entre una variable independiente  $x$  y una variable dependiente  $y$ . Se expresa como  $y = mx + b$ , donde  $m$  es la pendiente de la recta y  $b$  es la ordenada al origen o la intersección con el eje  $y$ . La regresión lineal intenta encontrar la recta que mejor se ajusta a los datos analizados, minimizando la distancia en el eje  $y$  entre las instancias y la recta.

Aunque se trata de un modelo ampliamente utilizado y de gran interpretabilidad, la regresión lineal presenta varias limitaciones importantes, entre ellas la presunción de una relación lineal entre las variables  $x$  e  $y$ , la alta sensibilidad a los valores atípicos (*outliers*) y a las correlaciones entre variables predictoras (multicolinealidad).



### 2.2.5.2. Regresión logística

La *regresión logística* [385] es un modelo matemático empleado principalmente para tareas de clasificación binaria. Permite predecir la probabilidad de que una instancia pertenezca a una de dos clases posibles por medio de una relación expresada en términos de la función logística:

$$P(y = 1|x) = \frac{1}{1 + e^{-(mx+b)}} \quad (2.1)$$

donde  $x$  es la variable de entrada,  $m$  es la pendiente que afecta a la proporción en la que influyen las variaciones en  $x$  en la salida y  $b$  es la intersección de la función con el eje  $x$ .

La función logística transforma la salida a un valor acotado entre 0 y 1, lo que la hace adecuada para modelar probabilidades y, por ende, es empleada para clasificación binaria. Intuitivamente, la salida se corresponde con la probabilidad de pertenencia a cada clase, considerando que si la probabilidad está entre 0,5 y 1 se predice la pertenencia a una clase, mientras que si está entre 0 y 0,5 se predice la pertenencia a la otra clase.

### 2.2.5.3. Clasificador Naïve Bayes

El clasificador *Naïve Bayes* [416] es un modelo probabilístico basado en el teorema de Bayes (ecuación 2.2), que describe la probabilidad condicional de un evento, basándose en el conocimiento previo de otros eventos relacionados:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (2.2)$$

Donde:

- $P(A|B)$  es la probabilidad de que el evento  $A$  ocurra dado que  $B$  ha ocurrido.
- $P(B|A)$  es la probabilidad de que  $B$  ocurra dado que  $A$  ha ocurrido.
- $P(A)$  es la probabilidad a priori del evento  $A$ .
- $P(B)$  es la probabilidad a priori del evento  $B$ .

Basándose en el teorema descrito, el modelo calcula la probabilidad de que una instancia pertenezca a una clase determinada dadas sus características y realiza la predicción en función de dichas probabilidades. Entre las ventajas de este algoritmo se encuentra su velocidad y escalabilidad a grandes conjuntos de datos, además de su interpretabilidad y de ser determinista (no estocástico). Sin embargo, Naïve Bayes asume que las características son condicionalmente independientes entre sí dada la clase, lo cual es una simplificación que generalmente no refleja con exactitud las características del problema a modelar.

#### 2.2.5.4. Árboles de decisión y bosques aleatorios

Los *árboles de decisión* [313] son modelos que emulan una estructura de datos de tipo árbol, en la que los atributos o características que describen a las instancias de datos son representadas por medio de nodos, los posibles valores o salidas de cada característica se representan con ramas, y las hojas del árbol representan las decisiones o predicciones finales. El proceso de construcción del árbol implica la elección de la mejor característica en cada paso para maximizar la ganancia de información.

Por otro lado, los *bosques aleatorios* [79] son conjuntos de árboles de decisión que se entrenan utilizando un proceso llamado *bagging*. Por este procedimiento, cada árbol en el bosque se entrena en una muestra aleatoria del conjunto de datos y luego se promedian las predicciones de todos los árboles para obtener una predicción final.

Los árboles de decisión y bosques aleatorios son aptos para ser entrenados con datos tanto numéricos como categóricos y pueden utilizarse para problemas de clasificación y de regresión. Además, estas técnicas permiten obtener modelos interpretables. Sin embargo, los árboles de decisión profundos pueden ser propensos al sobreajuste, y tienden a ser sensibles a pequeñas variaciones en los datos.

#### 2.2.5.5. Máquinas de Vectores de Soporte (*Support Vector Machines*)

Las Máquinas de Vectores de Soporte, mejor conocidas en el ámbito por su nombre en inglés *Support Vector Machines* o *SVM* [239], constituyen un poderoso modelo de aprendizaje supervisado empleado en problemas de clasificación. El objetivo de este modelo es encontrar un hiperplano óptimo en el espacio de características que separe las instancias de datos pertenecientes a diferentes clases, maximizando el margen entre las clases.

Intuitivamente, las SVMs se basan en la identificación de un hiperplano de separación óptimo, que esté a la máxima distancia posible de las muestras más cercanas de cada clase. Esto permite mejorar las capacidades de generalización del modelo y su resistencia al sobreajuste. El proceso de entrenamiento de una SVM consiste en optimizar una función de costo que busca minimizar el error de clasificación y maximizar el margen entre el hiperplano y las instancias. Cuando las instancias no son linealmente separables, se emplean núcleos o *kernels* de la función de costo que permiten proyectar los datos en espacios de mayor dimensionalidad, donde la separación lineal sí sea posible.

Las SVM son efectivas en espacios de alta dimensionalidad y pueden lidiar con conjuntos de datos no lineales. Sin embargo, son altamente sensibles a la elección del *kernel* y su entrenamiento puede resultar computacionalmente costoso en grandes conjuntos de datos.

## 2.3. Particionado de los datos

Realizar un adecuado particionado de los datos en aprendizaje automático resulta crucial, ya que impacta de forma directa en la evaluación y desempeño del modelo [235, 309]. Un buen particionado es determinante en el aprendizaje y la capacidad de generalización a datos no vistos por el modelo. Además, la división del conjunto de datos empleado en particiones disjuntas para entrenar y validar los resultados es necesaria para realizar una justa calibración de hiperparámetros del modelo y así evitar sesgos indeseables a la hora de evaluar la eficacia del modelo predictivo.

La elección de la estrategia de particionado de los datos depende de múltiples factores, entre ellos el tamaño del conjunto de datos, la naturaleza del problema a modelar y la cantidad de instancias de datos etiquetadas para cada una de las clases a predecir, en el caso de los modelos de clasificación. Algunas estrategias de particionado son más adecuadas cuando se desea hacer una evaluación preliminar del impacto de diferentes ajustes de hiperparámetros del modelo, mientras que otras aportan robustez y son preferibles al realizar una comparación entre diferentes algoritmos de modelado. A grandes rasgos, podemos distinguir dos estrategias de particionado predominantes:

- Particiones fijas: esta estrategia de particionado consiste en la separación del conjunto de datos en tres particiones disjuntas: Entrenamiento (*Train*), Validación interna (*Internal validation*) y Validación externa (*External validation* o *Held-out set*).

La partición de entrenamiento es la más grande del conjunto de datos y se utiliza para entrenar el modelo. En general, cuanto más grande y diversa sea la partición de entrenamiento mayor es

la probabilidad de que el modelo aprenda de manera más efectiva y generalice mejor a nuevos datos.

La partición de validación interna es utilizada para ajustar hiperparámetros y evaluar el desempeño del modelo en la fase de selección de modelos. El objetivo de la validación interna es proporcionar una evaluación imparcial de los resultados obtenidos por el modelo en datos no vistos durante el entrenamiento.

Por último, la partición de validación externa se mantiene completamente separada y no es utilizada en ninguna etapa de entrenamiento o validación preliminar del modelo. Esta partición es únicamente utilizada una vez hallado el modelo final para evaluar su rendimiento real en datos nunca vistos, por lo que su objetivo es proporcionar una visión imparcial de las capacidades de generalización del mismo.

La proporción de los datos del conjunto asignada a cada partición dependerá del tamaño del conjunto y de las características del problema. Generalmente, se destina entre el 70 % y el 80 % de los datos a entrenamiento, dejando el 20 % a 30 % restante a las particiones de validación interna y externa.

- Validación cruzada por pliegos (*Cross-fold validation*): en esta estrategia de particionado se procede a dividir el conjunto de datos en  $k$  secciones o *pliegos* de igual cantidad de instancias. Luego se entrenan  $k$  réplicas del modelo, empleando  $k - 1$  pliegos como partición de entrenamiento y 1 pliego (restante) como partición de validación, variando el pliego empleado para validación del modelo en cada réplica. Finalmente, se evalúa el desempeño del modelo promediando el rendimiento obtenido en cada una de las réplicas sobre el pliego de validación correspondiente.

Las estrategias de particionado fijo son empleadas para realizar una exploración preliminar de algoritmos de modelado, así como en las fases finales en las que se debe proporcionar un modelo único para predicción de datos novedosos. Si bien este particionado permite analizar el rendimiento de la estrategia de modelado sin necesidad de entrenar múltiples réplicas del modelo, tiene la desventaja de que vuelve al modelo proclive al sesgo inherente al particionado, por cuanto una partición de validación externa relativamente similar a la de entrenamiento en términos de los valores de sus características o atributos obtendría naturalmente mejor desempeño que una partición más diversa.

Por su parte, la estrategia de validación cruzada permite realizar una evaluación más robusta del modelo, ya que evalúa su rendimiento al iterar sobre diferentes conjuntos de datos, mitigando el impacto de los sesgos inherentes a una partición específica. Sin embargo, tiene como resultado  $k$

réplicas que *a priori* no han sido validadas en una partición de validación externa, por lo que a la hora de proporcionar un modelo predictivo final es necesario trabajar con particiones fijas y solo resulta adecuado modelar a partir de validación cruzada en las etapas de selección de modelos o cuando se realiza un análisis comparativo entre algoritmos de modelado para un caso de estudio particular.

En un diseño experimental pueden coexistir varias estrategias de particionado, siempre cuidando la integridad de las particiones y no introduciendo datos de validación en la etapa de entrenamiento. Un aspecto a tener en cuenta es la elección de las proporciones de datos asignadas a cada partición, a fin de evitar el sobreajuste del modelo. Por otra parte, en problemas de clasificación con alto desbalance entre clases el particionado debe hacerse de manera estratificada, de manera que la distribución de clases sea similar en todas las particiones o pliegos. Por último, es importante que todas las particiones incluyan instancias significativas del conjunto de datos en términos de los valores de sus características, de forma tal de evitar excluir instancias clave del proceso de entrenamiento, o de sesgar negativamente los resultados con una partición de validación externa disímil de los datos empleados para entrenar el modelo.

## 2.4. Dimensionalidad de los datos

Un aspecto crítico en el desempeño y el tiempo que insume el entrenamiento de los modelos predictivos es la dimensionalidad de los datos, la cual se define tanto en términos de la cantidad de características o atributos que describen cada instancia de datos como del rango de valores que cada atributo puede tomar. En líneas generales, los espacios de alta dimensionalidad implican una mayor dispersión de las instancias, lo cual puede dificultar la capacidad de detectar patrones significativos entre los datos y requiere de conjuntos de datos más grandes para entrenar modelos precisos. Además, la alta dimensionalidad aumenta la propensión del modelo al sobreajuste, especialmente en aquellos casos en los que la relación entre las características y el resultado es no lineal. Más aún, modelar en espacios de alta dimensionalidad implica una mayor complejidad computacional, debido a la necesidad de cálculos intensivos.

Para mitigar estos problemas se emplean estrategias de reducción dimensional, que buscan conservar la información relevante del conjunto de datos disminuyendo el número de características. No obstante, la reducción dimensional conlleva un riesgo de pérdida de información relevante para el modelado, puesto que una mayor dimensionalidad permite una representación más detallada y expresiva de los datos, lo que puede influir considerablemente en el rendimiento predictivo.

## 2.5. Aprendizaje profundo

El *aprendizaje profundo*, mejor conocido como *Deep Learning* en inglés, es una rama específica del aprendizaje automático que se concentra en el estudio y desarrollo de estrategias de aprendizaje basadas en redes neuronales artificiales profundas (*Deep Neural Networks* o *DNNs*). La estrategia de entrenamiento adoptada por los modelos basados en aprendizaje profundo está inspirada en el cerebro humano, conformado por redes de neuronas interconectadas que procesan y transmiten la información en forma de estímulos eléctricos [216, 128].

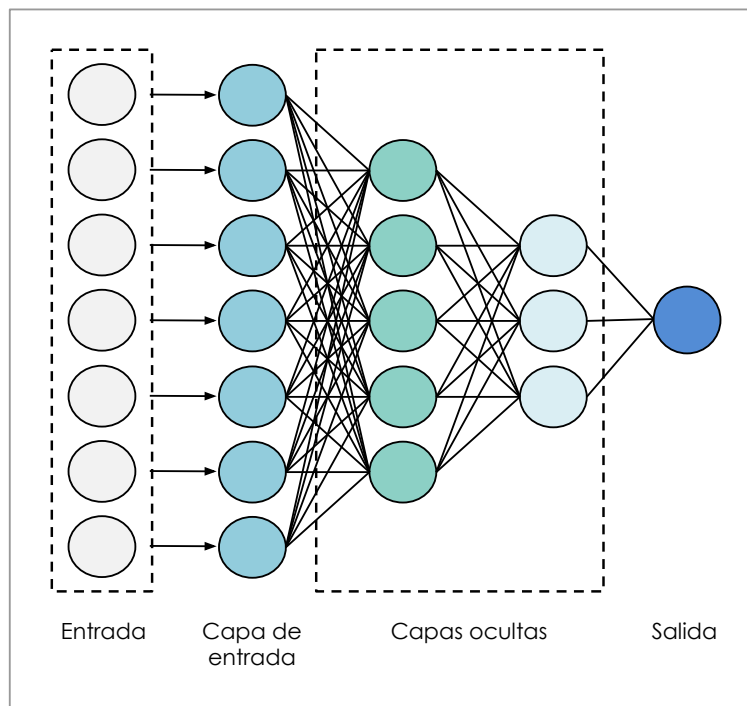
Desde la conceptualización de la primera red neuronal a mitades del siglo pasado [254] hasta las diversas y complejas arquitecturas con las que contamos en la actualidad, la investigación en aprendizaje profundo ha crecido exponencialmente y constituye un pilar fundamental en el desarrollo científico-tecnológico de las más diversas áreas de investigación interdisciplinarias. El auge del aprendizaje profundo se debe en parte al éxito de esta estrategia en resolver problemas de gran complejidad, como el procesamiento del lenguaje natural, el reconocimiento de imágenes y la traducción automática de textos, entre otros [93, 330]. Las DNNs pueden ser entrenadas a partir de casi cualquier tipo de dato, sin necesidad de llevar a cabo procesos complicados de ingeniería de atributos o de selección de características, por lo que dan lugar a representaciones enriquecidas y flexibles de los datos y, consecuentemente, obtienen mejores rendimientos predictivos. Además, el avance en el desarrollo de distintas técnicas de entrenamiento de DNNs han llevado a arquitecturas variadas y flexibles, aptas para diversas tareas en los más variados dominios, a la vez siendo capaces de lidiar con escenarios de aprendizaje complejos y multidimensionales y con conjuntos de datos masivos [216, 128].

El aprendizaje profundo ha sido un pilar fundamental en el desarrollo de la presente tesis, en la que hemos explorado una gran diversidad de arquitecturas y estrategias de entrenamiento de DNNs. En esta sección enumeramos y describimos de forma no exhaustiva algunos de los conceptos básicos relacionados a redes neuronales profundas y describimos técnicas importantes de entrenamiento, así como brindamos detalles sobre algunas de las arquitecturas de DNN más prominentes en la literatura.

### 2.5.1. Introducción a las Redes Neuronales Profundas (*Deep Neural Networks*)

Las redes neuronales profundas, comúnmente denominadas DNNs por sus siglas en inglés (*Deep Neural Networks*) son modelos predictivos muy flexibles de aprendizaje supervisado, que permiten

modelar tanto problemas de clasificación como de regresión. La arquitectura básica de una DNN sienta las bases para la comprensión del funcionamiento y diseño de las complejas arquitecturas neuronales modernas. Una DNN básica consiste en una serie de capas de nodos interconectados, donde podemos distinguir una *capa de entrada*, una o más *capas ocultas* y la *capa de salida*, tal y como se puede ver en la figura 2.1. Generalmente, la cantidad de nodos disminuye capa a capa, lo que permite aprender patrones de progresivamente mayor nivel de abstracción en cada capa. Los datos ingresan por la capa de entrada, en la que cada uno de los nodos representa una de las características o atributos del conjunto de datos. Cada nodo tiene un valor asociado, el cual es propagado hacia la siguiente capa por medio de conexiones ponderadas. Esta operación se repite capa por capa y, en este proceso, las capas ocultas detectan patrones y aprenden características más abstractas conforme se avanza hacia la capa de salida. Finalmente, la capa de salida cuenta con nodos que proporcionan los resultados de la predicción hecha por la DNN.



FFNN

Figura 2.1: Arquitectura básica de una red neuronal profunda, conocida por su nombre en inglés *Feed-Forward Neural Network* (FFNN).

Las conexiones entre los nodos tienen un peso y un sesgo (*bias*) asociados, los cuales son ajustados durante el entrenamiento. Estos dos parámetros determinan la influencia de un nodo sobre los

nodos en la siguiente capa de la arquitectura y, en esencia, el aprendizaje de la DNN depende del entrenamiento y ajuste de los mismos. Durante el entrenamiento, los parámetros entrenables son iterativamente ajustados mediante un algoritmo de optimización que minimiza una función de pérdida previamente especificada. A grandes rasgos, el proceso de entrenamiento de una DNN consta de los siguientes pasos:

1. Se inicializan los parámetros (pesos y *biases*) de manera aleatoria o siguiendo alguna función de inicialización. Estos valores constituyen el punto de partida del entrenamiento.
2. Se realiza un paso de *propagación hacia adelante*, donde los datos de entrada son transmitidos a través de la red neuronal capa por capa. En cada capa, se aplican transformaciones lineales y no lineales a los datos.
3. En la capa de salida se calcula el error por medio de una función de pérdida, la cual indica cuán diferentes son las predicciones de la red respecto a las etiquetas reales de los datos. La selección de la función de pérdida está determinada por el tipo de problema que se está modelando, sea de clasificación o de regresión.
4. Una vez computado el error en la predicción, se procede a ajustar los parámetros de la red para compensar o corregir dicho error. Esto se realiza por medio de un paso de *retropropagación*. En primer lugar, desde la capa de salida y atravesando cada una de las capas ocultas, se calculan las derivadas parciales de la función de pérdida con respecto a cada peso y *bias* de la red, las cuales intuitivamente indican cómo afectaría un pequeño cambio en dichos parámetros al error computado.
5. Por medio de las derivadas parciales calculadas durante la *retropropagación*, se ajustan los parámetros de la DNN utilizando algoritmos de optimización.

Estos pasos son repetidos durante múltiples iteraciones o *épocas* sobre el conjunto de datos de entrenamiento, lo que conlleva una optimización gradual de las predicciones de la red. El entrenamiento se realiza hasta alcanzar un criterio de parada determinado, el cual puede estar basado en el tiempo de entrenamiento, por ejemplo, al fijar una cantidad de iteraciones máxima, o en criterios de convergencia, por ejemplo, al llegar a iteraciones en las que solo se alcancen mejoras marginales o mínimas en los resultados.



## 2.5.2. Técnicas empleadas en el entrenamiento de modelos basados en aprendizaje profundo

En esta sección exploraremos algunas técnicas esenciales que sustentan el aprendizaje de las DNNs. Entre dichas técnicas, detallaremos algunas estrategias de inicialización de los parámetros entrenables de la DNN y enumeraremos algunos algoritmos de optimización, los cuales permiten regular la forma en la que se ajustan dichos parámetros. Luego enumeraremos las funciones de activación más utilizadas en la actualidad, las cuales son empleadas para aplicar transformaciones no lineales a los valores procesados por la red durante el entrenamiento. Además, exploraremos algunas técnicas de normalización, que colaboran en la convergencia temprana de la red y combaten problemas frecuentes como el desvanecimiento y explosión de gradientes. Finalmente, enumeraremos algunas estrategias de regularización, necesarias para controlar el sobreajuste.

Estas técnicas constituyen pilares fundamentales en el entrenamiento y desarrollo de DNNs, siendo factores determinantes en la capacidad de convergencia y en el desempeño predictivo de las mismas. Sin ánimos de profundizar en la explicación detallada de los fundamentos detrás de cada técnica, brindaremos referencias a los trabajos de investigación que las sustentan y fundamentos sobre la importancia de realizar una elección adecuada de estos elementos en el entrenamiento de DNNs para lograr modelos de aprendizaje profundo eficaces y generalizables.

### 2.5.2.1. Inicialización

La inicialización adecuada de los parámetros de una DNN es un aspecto crucial para lograr modelos estables y entrenamientos exitosos. La estrategia de establecimiento de los pesos iniciales en una red neuronal no solamente afecta significativamente a los tiempos y la capacidad de convergencia de los modelos, sino que también se ha demostrado sistemáticamente su impacto en el rendimiento predictivo final de los mismos [276].

Un problema recurrente en el entrenamiento de las DNNs es el desvanecimiento y la explosión de gradientes. El desvanecimiento de gradientes ocurre cuando los gradientes disminuyen exponencialmente durante el paso de retropropagación en el entrenamiento, lo cual dificulta el ajuste de los parámetros en las primeras capas de la DNN, especialmente en arquitecturas muy profundas. La explosión de gradientes, por su parte, se suscita cuando los gradientes crecen exponencialmente y resulta en ajustes drásticos en los parámetros. Ambos fenómenos atentan contra la convergencia del modelo, además de afectar su estabilidad y desempeño [128, 136]. La adecuada inicialización de pesos ayuda a mitigar estos problemas en el entrenamiento de redes profundas.

Enumeramos tres estrategias de inicialización de pesos empleadas en el marco de esta tesis doctoral, las cuales a su vez se encuentran entre las más comúnmente adoptadas en la literatura.

- **Inicialización Aleatoria:** la inicialización aleatoria es la estrategia más sencilla para establecer los pesos iniciales en una DNN. Tal y como su nombre lo indica, consiste en asignar valores aleatorios a los pesos en la primera iteración del entrenamiento. Además de su simplicidad, constituye una estrategia de inicialización computacionalmente eficiente. Sin embargo, una inicialización de pesos puramente aleatoria puede derivar en la saturación de las funciones de activación, con el consecuente potencial de desvanecimiento o explosión de gradientes, por lo que generalmente en la actualidad no es utilizada.
- **Inicialización de *Xavier/Glorot* [121]:** la inicialización de Xavier/Glorot es una estrategia hoy en día muy comúnmente utilizada para establecer los pesos iniciales en una red neuronal. El objetivo de esta estrategia de inicialización es mantener la varianza inicial de las activaciones constantes a través de las capas de la DNN, lo que contribuye a mitigar el problema del desvanecimiento o explosión de gradientes. Se basa en una fórmula matemática que calcula la varianza para inicializar los pesos de manera óptima, teniendo en cuenta el número de nodos de cada capa. Para una determinada capa, siendo  $n_{\text{in}}$  el número de nodos de la capa anterior y  $n_{\text{out}}$  el número de nodos de la capa en cuestión, los pesos se inicializan a partir de una distribución uniforme en el rango  $[-a, a]$ , donde  $a = \sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}$ .
- **Inicialización de *He* [151]:** esta estrategia de inicialización es similar a la inicialización de Xavier/Glorot [121], y es especialmente efectiva en los casos de DNNs que emplean funciones de activación no lineales no acotadas como *ReLU* (ver ecuación 2.12). El objetivo de la inicialización de He es evitar la desaparición o explosión de gradientes, manteniendo la varianza inicial de las activaciones constantes en los nodos de la red durante el paso de propagación hacia adelante. En esta estrategia se inicializan los pesos de forma aleatoria a partir de una distribución con  $\mu = 0$  y varianza  $\frac{2}{n_{\text{in}}}$ , donde  $n_{\text{in}}$  es el número de entradas a una capa determinada, coincidente con el número de nodos de la capa anterior.

### 2.5.2.2. Optimización

El proceso de optimización durante el entrenamiento de un modelo basado en redes neuronales implica ajustar los parámetros de la DNN para minimizar una *función de pérdida* [179, 408]. La función de pérdida permite medir la discrepancia entre las predicciones del modelo y los valores

de etiqueta de los datos de entrenamiento. La elección de la función de pérdida depende del tipo de problema supervisado a resolver, sea este de clasificación o de regresión. Entre las funciones de pérdida más comúnmente utilizadas enumeramos:

- *Entropía cruzada (Cross-Entropy)*: mide la discrepancia entre dos distribuciones de probabilidad, siendo comúnmente utilizada en problemas de clasificación multi-clase [45]. Para dos distribuciones  $\mathbf{y}$  y  $\hat{\mathbf{y}}$ , se define como:

$$H(\mathbf{y}, \hat{\mathbf{y}}) = - \sum y \cdot \ln(\hat{y}) \quad (2.3)$$

donde  $y$  son las etiquetas reales de los datos de entrenamiento,  $\hat{y}$  son las predicciones del modelo, y  $\ln$  es el logaritmo natural.

- *Entropía cruzada binaria (Binary Cross-Entropy)*: esta función de pérdida constituye una variante de la función de entropía cruzada específica para clasificación binaria [45]. Dada la etiqueta real del dato de entrada  $y$  y la predicción  $\hat{y}$  del modelo, se define como

$$\text{BCE}(y, \hat{y}) = -(y \cdot \ln(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})) \quad (2.4)$$

donde  $y$  es la etiqueta real (0 o 1) y  $\hat{y}$  es la predicción del modelo (probabilidad de pertenencia a la clase positiva o 1).

- *Hinge Loss*: es una función de pérdida comúnmente utilizada en problemas de clasificación, tanto binaria como multi-clase [408]. Dada la etiqueta real  $y$  y la predicción  $\hat{y}$ , se define como:

$$\text{Hinge Loss}(y, \hat{y}) = \text{máx}(0, 1 - y \cdot \hat{y}) \quad (2.5)$$

donde  $y$  representa las etiquetas reales de los datos de entrenamiento e  $\hat{y}$  representa las predicciones del modelo.

- *Error Cuadrático Medio o Mean Squared Error (MSE)*: esta función de pérdida, empleada en problemas de regresión, mide la diferencia cuadrática entre las predicciones y las etiquetas verdaderas [177]. Esta misma función es también empleada como métrica de rendimiento de modelos de regresión (ver sección 2.7). Se define como:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.6)$$

donde  $y$  representa las etiquetas reales de los datos de entrenamiento e  $\hat{y}$  representa las predicciones del modelo.

- *Error Absoluto Medio o Mean Absolute Error (MAE)*: similar a MSE, esta función de pérdida para problemas de regresión mide la media de las diferencias absolutas entre las predicciones y los valores de etiqueta reales [45, 304]. Al igual que en el caso de MSE, MAE es también empleada como métrica de rendimiento de modelos de regresión (ver sección 2.7). Se define como:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.7)$$

- *Divergencia de Kullback-Leibler (KL Divergence)*: la divergencia de Kullback-Leibler, comúnmente empleada en tareas de clasificación, mide la diferencia entre dos distribuciones probabilísticas  $P$  y  $Q$  definidas en  $x$  [395]. La ecuación correspondiente a esta función de pérdida es:

$$D_{KL}(P||Q) = \sum P(x) \log \left( \frac{P(x)}{Q(x)} \right) \quad (2.8)$$

Luego de la inicialización de los parámetros de la red se comienza el entrenamiento, durante el cual se calculan, en cada iteración, los gradientes de la función de pérdida con respecto a los parámetros de la DNN por medio de la retropropagación. Estos gradientes indican tanto la dirección como la tasa de cambio de la función de pérdida. En este punto crucial entran en juego los algoritmos de optimización [364], que emplean estos gradientes para actualizar los parámetros, ajustándolos en la dirección que minimice la función de pérdida. Este proceso se repite iteración tras iteración hasta que se termina el entrenamiento, sea por algún criterio de convergencia o de parada definido.

Los algoritmos de optimización en DNNs enfrentan desafíos particulares asociados a la alta dimensionalidad y la no convexidad de la función de pérdida [179, 408]. Generalmente, las funciones de pérdida a minimizar tienen múltiples óptimos locales, es decir, puntos en los que la función alcanza mínimos locales que no son el mínimo global de la función. En estas situaciones, se corre el riesgo de que el algoritmo de optimización se estanque en óptimos locales, existiendo otro punto en el espacio de parámetros al cual se debería llegar en el proceso de optimización, en el que la función de pérdida alcanza un valor aún más bajo.

Además de los óptimos locales, la naturaleza no convexa de la mayoría de las funciones de pérdida en modelos de aprendizaje profundo da lugar a la existencia de puntos de silla o *saddle*

*points*, llamados así en asociación a la función matemática tradicionalmente conocida como *silla de montar* ( $f(x,y) = ax^2 - by^2$ ). En los puntos de silla de una función de pérdida, algunas direcciones tienen gradientes cero o cercanos a cero en varias direcciones, así como también pueden tener gradientes ascendentes o descendentes en diferentes direcciones, lo que puede causar que el algoritmo de optimización se estanque en el entrenamiento.

Otro de los problemas que enfrentan los algoritmos de optimización es el desvanecimiento y explosión de gradientes, al cual nos referimos en la sección 2.5.2.1. Para lidiar con gradientes explosivos, en el proceso de optimización se suelen emplear técnicas como *recorte de gradientes* (*Gradient Clipping*) [446, 256, 292], que limita la magnitud de los gradientes durante el entrenamiento a un valor predefinido. En el caso del desvanecimiento de gradientes, entra en juego la adecuada selección de funciones de activación de los nodos de la DNN, sobre las cuales discutiremos en la sección 2.5.2.3. Asimismo, suele emplearse la estrategia de *normalización por lotes* (*Batch Normalization*) [173, 329], sobre la cual también elaboraremos una explicación en la sección 2.5.2.4.

En los algoritmos de optimización, un hiperparámetro crucial es la *tasa de aprendizaje*, universalmente denominada por su nombre en inglés: *learning rate* ( $\alpha$ ) [215, 216, 128]. El *learning rate* controla la proporción en la que se actualizan los parámetros de la DNN durante el proceso de optimización, y su adecuada selección constituye un paso vital en desarrollo de un modelo basado en redes neuronales. Por un lado, un *learning rate* demasiado alto puede causar problemas de convergencia en el entrenamiento, ya sea por ajustes drásticos en los parámetros de la red que vuelvan inestable el proceso, o por perder el punto óptimo (mínimo global) de la función de pérdida que se está intentando minimizar. Por otro lado, un *learning rate* demasiado bajo podría ocasionar que el proceso de entrenamiento sea excesivamente lento o se detenga prematuramente.

Entre los algoritmos de optimización más comúnmente empleados se destacan:

- *Descenso de gradiente por lotes* o *Batch Gradient Descent*: es el método clásico de optimización de modelos neuronales [320], al cual nos hemos referido intuitivamente durante el desarrollo del capítulo. En cada iteración, se calcula el gradiente de la función de pérdida con respecto a los parámetros de la red. Dicho gradiente indica la dirección en la que la función de pérdida crece, por lo que el objetivo del método es ajustar los parámetros en la dirección opuesta para hallar el mínimo global de dicha función. La actualización de los parámetros se realiza mediante la fórmula:

$$\theta_{i+1} = \theta_i - \alpha \cdot \nabla J(\theta_i) \quad (2.9)$$

donde  $\theta_i$  representa los parámetros en la iteración  $i$ -ésima,  $\alpha$  es el *learning rate* y  $\nabla J(\theta_i)$  es el gradiente de la función de pérdida en la iteración  $i$ -ésima.

Este algoritmo ajusta iterativamente los parámetros hasta alcanzar un punto donde el gradiente es cercano a cero, lo que indica que se ha alcanzado un mínimo local o global de la función de pérdida. Sin embargo, este método suele ser lento por emplear todos los datos de entrenamiento, además de que puede quedar atrapado en óptimos locales y no garantizar la convergencia al óptimo global en funciones no convexas.

- **Descenso de gradiente estocástico o *Stochastic Gradient Descent SGD*** [49]: este algoritmo de optimización clásico constituye una mejora con respecto al descenso de gradiente clásico. En SGD, se actualizan los parámetros del modelo de forma iterativa, empleando muestras aleatorias individuales del conjunto de datos en cada iteración. Esta diferencia no solo hace que SGD sea computacionalmente más eficiente que el algoritmo clásico, sino que además mitiga el estancamiento en óptimos locales. Al igual que el algoritmo de descenso de gradiente clásico, la actualización de los parámetros en SGD se realiza de acuerdo a la ecuación 2.9.
- ***Adam (Adaptive Moment Estimation)*** [198]: este algoritmo de optimización, uno de los más populares en la actualidad, combina ideas del algoritmo SGD, pero agrega un ajuste del *learning rate* para cada parámetro entrenable. En el algoritmo *Adam* se calculan dos promedios móviles: el *momento de primer orden*, que aproxima al valor medio móvil de los gradientes, y el *momento de segundo orden*, que aproxima a la media móvil de los cuadrados de los gradientes. La fórmula de actualización de los parámetros de una DNN por medio del algoritmo *Adam* es:

$$\begin{aligned} m_i &= \beta_1 m + (1 - \beta_1) \nabla J(\theta_i) \\ v_i &= \beta_2 v + (1 - \beta_2) (\nabla J(\theta_i))^2 \\ \theta_{i+1} &= \theta_i - \frac{\alpha}{\sqrt{v_i} + \epsilon} \cdot m_i \end{aligned} \tag{2.10}$$

donde  $m_i$  representa el momento de primer orden de la iteración  $i$ -ésima,  $v_i$  representa el momento de segundo orden de la iteración  $i$ -ésima,  $\alpha$  es el *learning rate*, y  $\nabla J(\theta_i)$  es el gradiente de la función de pérdida en la iteración  $i$ -ésima. Por su parte,  $\beta_1$  y  $\beta_2$  son parámetros que controlan la influencia de los momentos del primer y segundo orden  $m_i$  y  $v_i$ , y  $\epsilon$  es un parámetro infinitesimal agregado al denominador para evitar la división por cero.

- ***RMSProp (Root Mean Square Propagation)***: al igual que en el caso del algoritmo de optimización *Adam*, aborda el problema de desvanecimiento y explosión de gradientes que disminuyen o

explotan adaptando el *learning rate* para cada parámetro entrenable de la red. En el caso de *RMSProp* [375], se mantiene un promedio móvil del cuadrado de los gradientes, equivalente al momento de segundo orden en *Adam*, y el *learning rate* es dividido por la raíz cuadrada de este promedio. De esta manera, ante la ocurrencia de gradientes altos, el algoritmo ajusta automáticamente el valor del *learning rate* y evita ajustes drásticos en dichos parámetros. La fórmula de actualización de los parámetros en *RMSProp* es:

$$\begin{aligned} v_i &= \beta v_i + (1 - \beta)(\nabla J(\theta_i))^2 \\ \theta_{i+1} &= \theta_i - \frac{\alpha}{\sqrt{v_i} + \epsilon} \cdot \nabla J(\theta_i) \end{aligned} \quad (2.11)$$

donde  $v_i$  es el momento de segundo orden de la iteración  $i$ -ésima,  $\beta$  es un hiperparámetro de decaimiento, generalmente cercano a 1, que controla cuánta influencia tienen los gradientes de la iteración  $i - 1$  en el valor  $v_i$ ,  $\alpha$  es el *learning rate*, y  $\nabla J(\theta_i)$  es el gradiente de la función de pérdida en la iteración  $i$ -ésima. Al igual que en la ecuación de actualización de *Adam*,  $\epsilon$  es un parámetro infinitesimal agregado al denominador para evitar la división por cero.

### 2.5.2.3. Funciones de activación

Las funciones de activación permiten introducir transformaciones no lineales a los parámetros entrenables de una DNN, lo cual es esencial para aprender patrones complejos y representar relaciones no lineales entre las entradas y las salidas [15, 97, 190]. Intuitivamente, una función de activación no lineal computa la suma ponderada de todas las entradas conectadas a un nodo y les aplica una operación no lineal, la cual constituye la salida de dicho nodo que será propagada a los nodos de la capa siguiente de la DNN.

Como hemos discutido anteriormente, la correcta elección de las funciones de activación resulta crucial no solamente para mitigar los efectos del desvanecimiento o explosión de gradientes, sino que además es determinante en la convergencia del modelo durante el entrenamiento, permitiendo a la red neuronal ajustar sus parámetros de forma precisa y eficiente [97]. Un desafío adicional en la selección de las funciones de activación es el riesgo de introducir saturación o sesgos en los pesos, lo cual dificulta la optimización del modelo.

Las funciones de activación en una DNN cumplen diferentes roles según se apliquen en las capas ocultas, de entrada o de salida de la misma. Generalmente, en la capa de entrada la función de activación empleada suele ser la función identidad o una normalización de los atributos [282, 97].

Entre las funciones de activación empleadas más comúnmente en las capas ocultas de una DNN enumeramos:

- *ReLU (Rectified Linear Unit)* [275]: esta función de activación es una de las más utilizadas en el entrenamiento de DNNs debido a su eficiencia computacional y su capacidad para mitigar el problema de desvanecimiento de gradientes. Devuelve la entrada  $x$  si esta es positiva y cero si es negativa.

$$f(x) = \text{máx}(0, x) \quad (2.12)$$

- *Leaky ReLU (Rectified Linear Unit con fuga)*: se trata de una variante de la función de activación *ReLU*, ideada para mitigar uno de los problemas que puede exhibir dicha función: cuando la entrada es negativa, la función *ReLU* convencional produce la salida cero, lo que puede causar que los nodos de la DNN nunca se activen para ciertos datos [96]. La fórmula de la función *Leaky ReLU* está definida por casos, permitiendo para las entradas negativas que una pequeña fracción del valor pase a la salida del nodo.

$$f(x) = \begin{cases} x, & \text{si } x \geq 0 \\ \gamma x, & \text{si } x < 0 \end{cases} \quad (2.13)$$

donde  $\gamma$  es una constante, generalmente entre 0,01 y 0,3, que representa la fuga de la función de activación para las entradas negativas.

- *Tangente hiperbólica (Tanh)*: La función de activación *Tanh* produce salidas dentro del rango  $[-1,1]$ , lo cual resulta de utilidad para lograr parámetros entrenables normalizados y minimizar la explosión y desvanecimiento de gradientes. Asimismo, al estar centrada en cero y ser inherentemente simétrica, abona a la convergencia y estabilidad de la red.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.14)$$

En el caso de la capa de salida de una DNN, la elección de la función de activación depende del tipo de modelo supervisado que se esté desarrollando. En el caso de los problemas de regresión, la capa de salida puede emplear la función identidad o emplear una función lineal, ya que se busca obtener un valor de salida continuo que represente la predicción del modelo. En el caso de los problemas de clasificación, las dos funciones de activación más comúnmente empleadas son:



- Sigmoidea: La función sigmoidea transforma el valor de entrada a una salida en el rango (0,1), la cual es interpretada como la probabilidad de pertenencia a una de dos clases en problemas de clasificación binaria, donde el umbral de decisión entre las dos clases se sitúa en 0,5.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.15)$$

- Softmax: esta función de activación se emplea en problemas de clasificación multi-clase. Permite asignar probabilidades de pertenencia a cada clase, asegurando que la suma de estas probabilidades sea 1.

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^N e^{x_j}} \quad (2.16)$$

#### 2.5.2.4. Normalización

Las técnicas de normalización tienen como objetivo estandarizar o regularizar los datos empleados para y durante el entrenamiento, a fin de ajustar los valores de la DNN a un rango o distribución específica [363, 230]. A grandes rasgos, podemos distinguir dos tipos de etapas de normalización igualmente importantes: la *normalización de entradas*, y la *normalización de activaciones*. Ambas etapas de normalización tienen como objetivo estabilizar y acelerar el proceso de entrenamiento, agilizando la convergencia del proceso de optimización y mitigando los efectos negativos del desvanecimiento o explosión de gradientes.

La normalización de entradas consiste en la transformación de los datos de entrada por medio de una operación de estandarización, a fin de ajustarlos a una cierta escala y distribución antes de que se propaguen a través de la red. Este paso se realiza, por lo tanto, antes de iniciar el entrenamiento de la DNN, y permite garantizar que todas las características o atributos del conjunto de datos contribuyan *a priori* de manera similar al proceso de entrenamiento. La normalización de entradas posibilita entrenamientos más rápidos y eficaces, logrando modelos de mejor desempeño.

La normalización de activaciones, por otro lado, hace referencia a las técnicas de ajuste de los valores de activación, obtenidos a la salida de cada nodo de la DNN, a una distribución y escala determinadas. Esta etapa de normalización se realiza durante el entrenamiento, después de que los datos atraviesan cada capa de la DNN y son transformados por medio de las funciones de activación de dicha capa. Los valores de dichas activaciones se ajustan para que tengan una media y varianza específicas, lo que contribuye a mantener la estabilidad del modelo, acelerar la convergencia

y mejorar sus capacidades de generalización. Las estrategias de normalización de activaciones juegan un rol fundamental en la prevención del desvanecimiento y explosión de gradientes, por cuanto escalan y estabilizan la magnitud de las activaciones [363, 230]. Al garantizar gradientes estables, en consecuencia, las DNNs que implementan normalización de activaciones admiten el uso de *learning rates* más altos, lo que acelera significativamente la convergencia [325].

Existen una variedad de estrategias de normalización de activaciones. Se puede implementar una *normalización por lotes* (*Batch Normalization, BN*) [173]; por *capas* (*Layer Normalization, LN*) [23]; por *instancias* (*Instance Normalization, IN*) [390], o por *grupos* (*Group Normalization, GN*) [429]. La diferencia entre cada una de estas estrategias radica fundamentalmente en las porciones de datos a partir de las cuales se toman las estadísticas para realizar la estandarización.

De las técnicas mencionadas, la más universalmente utilizada en diferentes dominios del aprendizaje profundo es la *normalización por lotes* (*Batch Normalization, BN*) [173]. Esta técnica normaliza las activaciones de cada capa por lotes (*batches*) de datos durante el entrenamiento, ajustando su media y desvío estándar. Siendo  $\mathbb{X}$  el vector de activaciones de una capa determinada, la ecuación de *Batch Normalization* se define como:

$$\text{BN}(\mathbb{X}) = \frac{\mathbb{X} - \mu}{\sqrt{\sigma^2 + \epsilon}} \cdot \gamma + \beta \quad (2.17)$$

donde  $\mu$  es la media aritmética de  $\mathbb{X}$  en un lote (*batch*) de datos,  $\sigma^2$  es la varianza de  $\mathbb{X}$  en dicho lote,  $\epsilon$  es una pequeña constante para evitar la división por cero, y  $\gamma$  y  $\beta$  son los parámetros de escalado y sesgo, respectivamente. Tanto  $\gamma$  como  $\beta$  son aprendidos durante el entrenamiento a partir de un valor inicial elegido al inicio del entrenamiento.

### 2.5.2.5. Regularización

En el contexto del entrenamiento de DNNs, las técnicas de regularización resultan esenciales para evitar el sobreajuste y promover modelos con mayor capacidad de generalización [129, 373, 263]. Entre las estrategias más comúnmente utilizadas se destacan las siguientes:

- **Regularización L1 y L2:** estas dos estrategias de regularización introducen términos de penalización en la función de pérdida computada por la DNN, los cuales se computan a partir de la magnitud de los pesos de la red.

La regularización L1, también conocida como *Regularización de Lasso* (*Least Absolute Shrinkage and Selection Operator*), agrega una penalización a la función de pérdida proporcional

a la suma de los valores absolutos de los pesos  $\theta$  de la DNN. En términos matemáticos, la regularización L1 se expresa como:

$$\text{L1} = \lambda \sum_{i=1}^n |\theta_i| \quad (2.18)$$

donde  $\lambda$  constituye el factor de penalización y  $n$  es el número total de pesos en la red. Intuitivamente, la regularización L1 tiene un efecto de supresión sobre los pesos de la red, favoreciendo la dispersión de los pesos hacia cero, lo cual deviene en un proceso de selección de características automático y, por ende, abona a la simplicidad del modelo.

La regularización L2, también conocida como *Regularización de Ridge*, funciona de forma similar a la regularización L1, agregando una penalización a la función de pérdida proporcional a la suma de los cuadrados de los pesos  $\theta$  de la DNN. Matemáticamente, se expresa como:

$$\text{L2} = \lambda \sum_{i=1}^n \theta_i^2 \quad (2.19)$$

donde  $\lambda$  es el factor de penalización y  $n$  es el número total de pesos en la red. A diferencia de la regularización L1, esta estrategia tiene un efecto de reducción suave sobre los pesos, reduciendo la magnitud de los mismos sin forzarlos a ser exactamente cero. Esto evita que se vuelvan excesivamente grandes y contribuyan al sobreajuste.

- *Dropout*: esta técnica, destacada por su simpleza y efectividad, consiste en la desactivación aleatoria de un porcentaje de nodos neuronales en determinadas capas ocultas de la DNN durante el entrenamiento [27]. Los nodos a desactivar son elegidos aleatoriamente en cada iteración, lo que evita que la red aprenda desarrollando dependencias fuertes a ciertas conexiones en la arquitectura y, por ende, mejora su capacidad de generalización y robustez.

Durante el entrenamiento, se inhibe o apaga aleatoriamente una fracción  $p$  de los nodos neuronales en cada capa oculta, lo que significa que se eliminan temporalmente de la red. Durante el paso de propagación hacia adelante del entrenamiento de la DNN, la salida de un nodo  $i$  en una capa dada es inhibida con probabilidad  $1 - p$ . Luego, durante el paso de retropropagación, solamente aquellos nodos que no fueron desactivados temporalmente contribuyen al gradiente computado, por lo que la derivada parcial del error con respecto a los pesos ( $\frac{\partial \text{error}}{\partial w_i}$ ) debe ser multiplicada por  $p$  a fin de contabilizar los nodos activos. Es importante notar que una vez finalizado el entrenamiento de la DNN, todos los nodos neuronales son utilizados en la inferencia, a fin de mantener la coherencia en la salida de la red.

- Parada Temprana o *Early Stopping*: si bien no se trata de una estrategia que regulariza directamente los parámetros de la red, *Early Stopping* es considerada una técnica de regularización, ya que colabora en la prevención del sobreajuste. Consiste en detener el entrenamiento de la DNN de forma temprana, cuando hay indicios de que el modelo comienza a sobreajustarse a los datos de entrenamiento. En esta estrategia, se supervisa el desempeño en una partición de validación interna durante el entrenamiento y, en lugar de completar todas sus iteraciones o épocas, se detiene el proceso cuando el valor de la métrica monitoreada computado sobre la partición de validación interna no exhibe mejoras significativas.

Durante el proceso de entrenamiento de la red, se guarda el modelo en cada época (*checkpoint*) y se evalúa el desempeño del modelo guardado en una partición de validación interna. Si el rendimiento del modelo no mejora al cabo de una cantidad de iteraciones determinada, se detiene el entrenamiento y se selecciona el *checkpoint* del modelo que haya obtenido el mejor desempeño en los datos de validación.

Para implementar *Early Stopping* se requiere establecer una serie de hiperparámetros, entre los cuales se destacan la *paciencia* (*patience*), que representa la cantidad de iteraciones consecutivas de tolerancia que se le da al modelo al cabo de las cuales se detiene su entrenamiento, y el *diferencial* ( $\delta$ ), que representa la cantidad mínima de cambio en la métrica monitoreada considerada una mejora significativa en el desempeño. Además, se debe seleccionar la métrica a monitorear, la cual puede ser una métrica de desempeño a maximizar (por ejemplo, exactitud o precisión) o a minimizar (por ejemplo, la función de pérdida de la DNN).

La aplicación de *Early Stopping* evita que el modelo se ajuste demasiado a los datos de entrenamiento al monitorear su desempeño en un conjunto de datos no visto, como lo es la validación interna. No obstante, pueden obtenerse modelos subóptimos si se detiene el entrenamiento demasiado temprano, por lo que resulta crucial realizar una parametrización adecuada y equilibrada de la técnica.

### 2.5.3. Arquitecturas profundas

La creación, diseño y proliferación de las diferentes arquitecturas de redes neuronales profundas que observamos en la actualidad responden a años de investigación en aprendizaje profundo. El éxito de las DNNs en tareas diversas y complejas, así como la disponibilidad de grandes conjuntos de datos y los avances tecnológicos y en poder de cómputo, fueron factores determinantes para el avance exponencial en esta área de investigación [414, 411].

En esencia, las arquitecturas de redes neuronales profundas son estructuras que definen la forma en la que los nodos neuronales y las capas se organizan y conectan dentro de la red. Además, en algunos casos involucran nodos o celdas especiales, con funciones de activación propias a la arquitectura, especialmente diseñadas para una cierta tarea. Las arquitecturas de DNN evolucionan constantemente para abordar distintos problemas y tareas, a la vez proponiendo nuevos desafíos de interpretabilidad, eficiencia y poder de cómputo. La elección de la arquitectura depende de la naturaleza de los datos y la tarea que se desea realizar, siendo además frecuente la integración de elementos de múltiples arquitecturas en modelos híbridos en el abordaje de problemas de modelado multimodales o complejos.

En esta sección enumeramos y describimos brevemente algunas de las arquitecturas de DNN más notables, concentrándonos en aquellas que hemos empleado durante las fases experimentales de esta tesis. Existen numerosos trabajos bibliográficos que abordan en mayor profundidad cada una de dichas arquitecturas, los cuales dejamos a consideración de lx lectorx [164, 54, 11, 250].

### 2.5.3.1. Redes neuronales de alimentación hacia adelante - *Feed-Forward Neural Networks (FFNN)*

Las redes neuronales de alimentación hacia adelante, universalmente conocidas como *Feed-Forward Neural Networks* (FFNN), constituyen la arquitectura fundacional en el campo del aprendizaje profundo [370]. La topología característica de las FFNN está conformada por nodos neuronales dispuestos en capas, donde cada nodo está conectado a todos los nodos de la capa anterior y siguiente. En las FFNN no hay retroalimentación, sino que los datos fluyen desde la capa de entrada hacia la capa de salida durante la fase de propagación hacia adelante, y los parámetros entrenables se ajustan durante el paso de retropropagación.

La estructura básica de una FFNN consta de una capa de entrada, que recibe los datos de entrada y consta de un nodo por cada característica o atributo del conjunto de datos; una serie de capas ocultas, capas intermedias que transmiten los datos de entrada y los transforman por medio de las funciones de activación no lineales; y finalmente una capa de salida, que produce los resultados de la DNN basándose en las transformaciones aplicadas a lo largo de la DNN. Un ejemplo de la topología de una FFNN se presenta en la figura 2.1.

Como hemos descripto anteriormente a lo largo del presente capítulo, cada conexión entre los nodos tiene un peso y un *bias* asociado, los cuales son ajustados durante el entrenamiento, y los nodos implementan funciones de activación específicas para transformar los datos. En particular,

los nodos en la capa de salida implementan una función de activación específica de acuerdo a la naturaleza del problema (clasificación o regresión).

Las FFNN son empleadas en una amplia variedad de tareas, dada su versatilidad y flexibilidad. Debido a su diseño modular y gran capacidad de aprendizaje y detección de patrones complejos, suelen ser fácilmente adaptables y acoplables a arquitecturas más complejas, por lo general como subestructuras de salida para dar soporte a tareas de clasificación y regresión a aquellas arquitecturas diseñadas con otros propósitos. A partir de este tipo de arquitecturas pueden diseñarse diversos modelos, tales como las arquitecturas de aprendizaje multi-tarea, que comparten un núcleo de capas en común y luego se subdividen en núcleos de capas específicas a diferentes tareas de predicción [319]. Este tipo de modelo es explicado en mayor detalle en el capítulo 5 de la presente tesis.

### 2.5.3.2. Redes neuronales convolucionales - *Convolutional Neural Networks (CNN)*

Las redes neuronales convolucionales (CNN) constituyen una arquitectura de gran importancia en el aprendizaje profundo aplicado a procesamiento de imágenes y videos [223]. Las CNNs son ampliamente utilizadas en tareas como clasificación de imágenes, segmentación semántica y detección de objetos, en las que han demostrado un desempeño excepcional vinculado a su capacidad para aprender representaciones jerárquicas de características complejas a partir de representaciones de datos con bajo preprocesamiento.

Inicialmente diseñadas para procesar datos expresados como matrices bidimensionales, las CNN emplean *capas convolucionales* que aplican *filtros* a subregiones de la entrada. Estos filtros están conformados por matrices que se aplican siguiendo una metodología de ventana deslizante sobre la matriz de entrada, calculando productos escalares locales que les permiten detectar patrones específicos. Además de las capas convolucionales, las CNN suelen incluir capas de *pooling*, que agrupan subregiones de los datos en ventanas y por ende actúan como paso de reducción dimensional de la salida de las capas convolucionales. Mostramos un diagrama esquemático de una CNN en la figura 2.2.

### 2.5.3.3. Redes neuronales recurrentes - *Recurrent Neural Networks (RNN)*

Las redes neuronales recurrentes (RNN) constituyen un tipo de arquitectura de redes neuronales inicialmente diseñado para el procesamiento de datos secuenciales, como series de tiempo o texto [444]. En contraste con las FFNNs, las RNN poseen conexiones retroalimentadas; es decir, las salidas de una capa en un instante de tiempo son suministradas como entrada a la misma capa en el

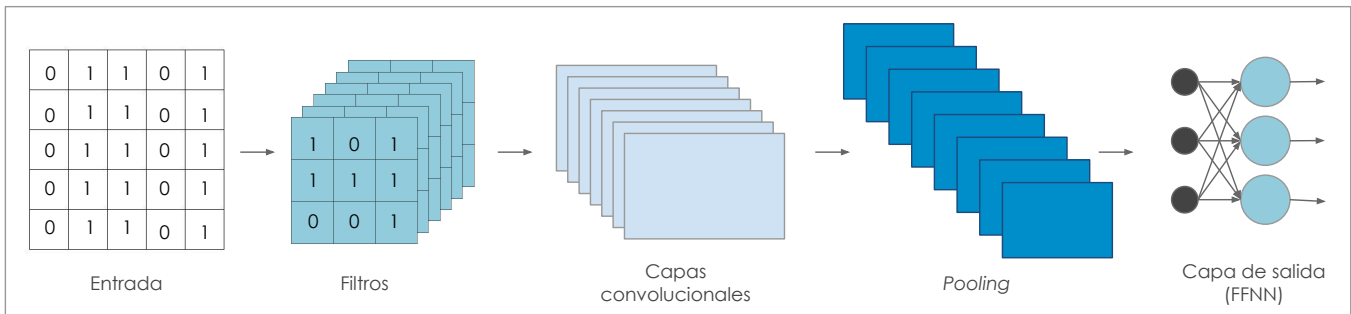


Figura 2.2: Diagrama esquemático de una red neuronal convolucional (CNN).

siguiente instante de tiempo. Este diseño, el cual esquematizamos en la figura 2.3, permite que las RNNs mantengan un estado interno o memoria, que les permite almacenar información previa en la secuencia y tener en cuenta el contexto al procesar nuevos datos.

Al procesar una secuencia de datos en una RNN, la secuencia es en primera instancia separada en unidades atómicas o *tokens*, los cuales son suministrados a la RNN uno a uno en distintos instantes de tiempo consecutivos. El proceso de entrenamiento de una RNN consiste en ajustar los parámetros que representan el estado interno del modelo iterativamente en cada instante de tiempo, el cual almacena la información procesada hasta el momento. A partir del procesamiento de la secuencia completa, la RNN produce una salida que se computa en función de la entrada y de sus estados internos en el instante de tiempo final. Su capacidad para capturar dependencias o patrones temporales vuelve a las RNNs en una arquitectura idónea para tareas de procesamiento de lenguaje natural y análisis de series temporales. No obstante, son propensas al desvanecimiento o explosión de gradientes al procesar secuencias largas, por lo que se han desarrollado dos variantes populares de RNN que permiten mitigar este problema: las redes basadas en celdas *Long Short-Term Memory (LSTM)* [159] y las redes basadas en celdas *Gated Recurrent Units (GRU)* [70].

Las celdas LSTM (*Long Short-Term Memory*) [159] constituyen un tipo de unidad recurrente empleado para construir RNNs. Las celdas LSTM incorporan compuertas que controlan la forma en la que fluyen los datos en la RNN, las cuales constan de unidades internas que implementan funciones de activación sigmoidea (explicada en la sección 2.5.2.3). Intuitivamente, estas unidades determinan en cada instante de tiempo qué información contextual de la secuencia ya procesada debe ser recordada o descartada, lo que contribuye a la conservación de información ligada a dependencias de largo plazo en las secuencias. El diseño de las celdas LSTM permite priorizar la información crítica y mejorar la capacidad de generalización de la RNN.

Tal y como se ilustra en la figura 2.4, una celda LSTM está conformada por tres compuertas claves: la

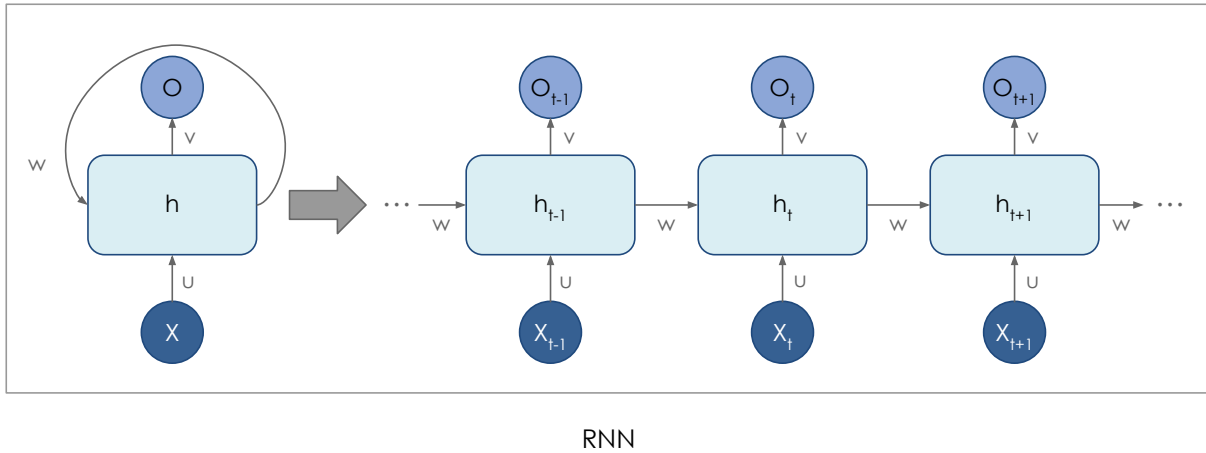


Figura 2.3: Diagrama esquemático de una red neuronal recurrente (RNN), donde  $X_t$  denota el *token*  $t$ -ésimo de la secuencia de entrada,  $u, v$  y  $w$  son los pesos de las celdas recurrentes,  $h_t$  denota el  $t$ -ésimo estado interno de la RNN y  $O_t$  denota la  $t$ -ésima salida de la RNN.

compuerta de Olvido ( $f_t$ ), que controla cuánta información debe descartarse del estado de memoria anterior ( $C_{t-1}$ ); la compuerta de Entrada ( $i_t$ ), que controla cuánta de la nueva información se debe almacenar en la celda de memoria ( $C_t$ ); y la compuerta de Salida ( $o_t$ ), que controla cuánta información de la celda de memoria debe usarse para producir la salida ( $h_t$ ).

Las ecuaciones para cada una de las componentes de una celda LSTM son:

- Celda de Memoria ( $C_t$ ):  $C_t = f_t \odot C_{t-1} + i_t \odot \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$
- Compuerta de Olvido ( $f_t$ ):  $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
- Compuerta de Entrada ( $i_t$ ):  $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
- Compuerta de Salida ( $o_t$ ):  $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
- Salida de la Celda ( $h_t$ ):  $h_t = o_t \odot \tanh(C_t)$

En estas ecuaciones,  $x_t$  representa la entrada en el instante de tiempo actual  $t$ ,  $h_{t-1}$  es la salida de la celda obtenida en el instante de tiempo anterior ( $t-1$ ),  $W$  y  $b$  representan los pesos y *biases* de la celda LSTM,  $\sigma$  es la función de activación sigmoidea y  $\tanh$  es función de activación tangente hiperbólica. Finalmente,  $\odot$  denota el producto elemento por elemento, y  $[h_{t-1}, x_t]$  denota la concatenación de  $h_{t-1}$  y  $x_t$ .



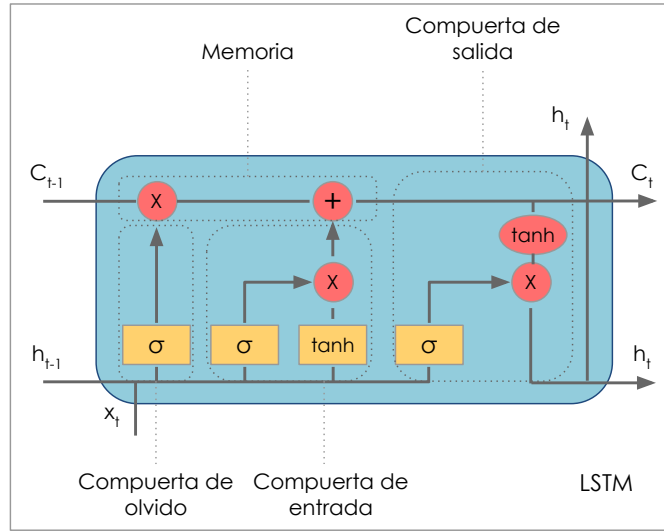


Figura 2.4: Diagrama esquemático de una celda *Long-Short Term Memory* (LSTM).

Las celdas GRU (*Gated Recurrent Units*) [70] son otro tipo de celda recurrente que, al igual que en el caso de las celdas LSTM, fueron diseñadas para mejorar la memoria de las RNNs ante dependencias de largo plazo y mitigar los efectos negativos del desvanecimiento de gradientes. El diseño de las celdas GRU permite capturar dependencias temporales en datos secuenciales por medio de dos compuertas principales: la compuerta de Reinicio ( $r_t$ ), que controla cuánto de la información del estado anterior  $h_{t-1}$  debe olvidarse o conservarse para el cálculo de la nueva salida, y la compuerta de Actualización ( $z_t$ ), que determina cuánta de la información del estado actual debe incorporarse en el cómputo de la salida candidata. La figura 2.5 muestra la estructura básica de una celda GRU. Las ecuaciones que describen una celda GRU son las siguientes:

- Compuerta de Reinicio ( $r_t$ ):  $r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$
- Compuerta de Actualización ( $z_t$ ):  $z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$
- Salida candidata ( $\tilde{h}_t$ ):  $\tilde{h}_t = \tanh(W \cdot [r_t \odot h_{t-1}, x_t])$
- Salida de la celda GRU ( $h_t$ ):  $h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$

En estas ecuaciones,  $x_t$  es la entrada en el instante de tiempo actual  $t$ ,  $h_{t-1}$  es la salida de la celda en el instante de tiempo anterior ( $t - 1$ ),  $W_r$ ,  $W_z$ , y  $W$  son los pesos asociados a la celda GRU,  $\sigma$  es la función sigmoidea,  $\tanh$  es la tangente hiperbólica, y  $\odot$  denota el producto elemento por elemento.

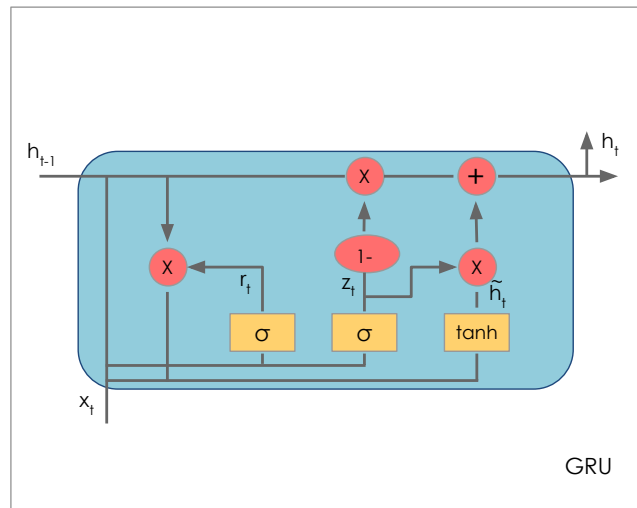


Figura 2.5: Diagrama esquemático de una celda *Gated Recurrent Unit* (GRU), donde  $r_t$  representa la compuerta de reinicio,  $z_t$  representa la compuerta de actualización y  $\tilde{h}_t$  representa la salida candidata  $t$ -ésima de la secuencia procesada.

Una de las desventajas de las RNNs es su costo computacional: especialmente en el caso de secuencias largas, su entrenamiento es lento e insume muchos recursos, pues se deben mantener estados internos para almacenar información de toda la secuencia. Los avances en la investigación de arquitecturas profundas devinieron en el desarrollo del mecanismo de *atención* (*attention*) [25], al cual nos referiremos en la sección 2.5.3.4 que constituye el pilar fundamental detrás de la vasta mayoría de los modelos actuales y fue inicialmente diseñado para mitigar este problema.

#### 2.5.3.4. Mecanismo de atención y *Transformers*

A partir de la publicación científica seminal publicada por Bahdanau et al. [25], seguida por el innovador trabajo de Vaswani et al. [397], que presentó el modelo denominado *Transformer* y es considerado un hito en el procesamiento del lenguaje natural, la introducción de los mecanismos de atención ha revolucionado el desarrollo de arquitecturas profundas durante la última década. El mecanismo de atención se inspira en el modo en el que los humanos identifican patrones y aprenden de la información observando y concentrando su atención en diferentes partes de la información en simultáneo.

Coloquialmente, el mecanismo de atención permite analizar todas las partes de una entrada secuencial en simultáneo y asignar ponderaciones dinámicas a cada una de ellas. Dichas ponderaciones

o pesos se calculan basándose en la relevancia de cada una de dichas partes de la secuencia con respecto a la tarea. Estos pesos son luego empleados para ajustar la representación de entrada de los datos, lo que permite obtener nuevas representaciones enriquecidas con información contextual de toda la secuencia. Estas representaciones capturan la información relevante del dato y de las interrelaciones entre las diferentes partes de la secuencia de entrada.

En principio, podemos distinguir dos tipos de implementación del mecanismo de atención. Por un lado, el mecanismo puede estar aplicado sobre una misma secuencia, en donde los elementos o partes de dicha secuencia se entienden como relacionados entre sí, lo que se conoce como *Self-Attention* [397]. Por otro lado, el mecanismo de atención puede aplicarse entre dos secuencias diferentes o entre dos representaciones diferentes de una misma secuencia, lo que se denomina *Cross-Attention* [397]. A su vez, las arquitecturas que implementan el mecanismo de atención pueden ser *de único cabezal (Single-Head Attention)* o *de múltiples cabezales (Multi-Head Attention)* [397]. La distinción entre ambos tipos de arquitectura radica en la cantidad de proyecciones paralelas entre los elementos o partes de la secuencia, computadas para obtener los conjuntos de pesos de atención. En otras palabras, en las arquitecturas de múltiples cabezales se obtienen  $r$  conjuntos de pesos que vinculan las partes de la secuencia, siendo  $r$  la cantidad de cabezales de atención, con  $r > 1$ . En esta sección nos referimos en particular a *Self-Attention* y *Multi-Head Attention* por su relevancia en el desarrollo de modelos modernos y por ser dos de las estrategias centrales en el abordaje experimental de múltiples trabajos presentados en el contexto de esta tesis doctoral.

En el mecanismo de *Self-Attention* [397], la atención se aplica sobre una misma secuencia de entrada. Cada elemento de la secuencia tiene una representación propia y una posición dentro de la secuencia, a partir de los cuales se calculan múltiples pesos de atención para cada par de elementos, los cuales son luego combinados para construir una nueva representación de la secuencia. Este cálculo se realiza en paralelo para todos los elementos, lo cual constituye una mejora notable en términos de tiempo de cómputo con respecto al procesamiento de secuencias con redes neuronales recurrentes (RNNs). Cada uno de los pesos vincula a un elemento de la secuencia con todos los restantes, lo cual intuitivamente establece vínculos de atención o correlación entre todas las partes de la secuencia.

Suponiendo una secuencia de entrada  $X = \{x_1, x_2, \dots, x_n\}$ , para cada elemento  $x_i$  en la secuencia se computan tres representaciones:

1. Consulta (*Query*):  $Q_i = XW_Q$
2. Clave (*Key*):  $K_i = XW_K$

3. Valor (*Value*):  $V_i = XW_V$

donde  $W_Q, W_K, W_V$  son matrices de parámetros entrenables del modelo. Luego de computar las consultas, claves y valores de cada elemento, se calculan los pesos de atención  $\alpha_{ij}$  entre cada par de elementos  $x_i$  y  $x_j$ . Estos pesos de atención se obtienen mediante la función de atención:

$$\alpha_{ij} = \text{softmax} \left( \frac{Q_i K_j^T}{\sqrt{d_k}} \right)$$

donde  $d_k$  es la dimensionalidad de las claves  $K$ . Finalmente, la salida para cada elemento  $O_i$  se calcula combinando los valores  $V_j$  ponderados por los pesos de atención  $\alpha_{ij}$ :

$$O_i = \sum_{j=1}^n \alpha_{ij} V_j$$

Esta salida constituye la representación de un elemento de la secuencia en función de los demás elementos de la misma secuencia. Este proceso se realiza para cada elemento de la entrada, lo que da como resultado la representación final de la secuencia.

El mecanismo de *Multi-Head Attention* [397] aplica el mismo principio de cómputo de pesos de atención para cada elemento de una secuencia de entrada, con la salvedad de que se computan múltiples conjuntos de pesos de atención en simultáneo, cada uno a partir de una proyección lineal diferente de la secuencia de entrada. Dichos conjuntos de pesos de atención tienen sus propias representaciones, las cuales son finalmente concatenadas y proyectadas para obtener la salida final. La finalidad de emplear múltiples cabezales de atención es detectar diferentes tipos de interacción entre los elementos de la secuencia, lo que da lugar a representaciones ricas en información contextual. La figura 2.6 presenta una abstracción gráfica de una capa de atención de múltiples cabezales (*Multi-Head Attention*) y una vista en detalle del cómputo de los pesos de atención, la cual fue reproducida a partir de una figura análoga presentada por Vaswani et al. [397] en su trabajo.

### 2.5.3.5. *Autoencoders* y *Embeddings*

Los *autoencoders* conforman una categoría única de modelos basados en redes neuronales que se emplean en el contexto de aprendizaje no supervisado [297, 29]. A grandes rasgos, constan de dos partes: un *codificador* o *encoder*, que recibe el dato de entrada en alta dimensionalidad y lo transforma a una representación de baja dimensionalidad, denominada *embedding* [37], y un *decodificador* o *decoder*, el cual es entrenado para reconstruir la representación original en alta dimensionalidad del dato de entrada a partir del *embedding* obtenido por el *encoder*. El objetivo de un *autoencoder* es la obtención de representaciones comprimidas de las entradas, que capturan la información más significativa de los datos, intentando evitar la pérdida de información relevante.

El entrenamiento de un *autoencoder* busca minimizar la diferencia entre la entrada original y la salida, la cual en principio debe ser una reconstrucción idéntica de la entrada. Las arquitecturas del *encoder* y *decoder* pueden ser diversas, por lo que en la bibliografía es común encontrar *autoencoders* basados en CNNs, RNNs, *Self-Attention*, FFNNs y más [297, 29]. La función de pérdida a emplear durante el proceso de entrenamiento depende del tipo de arquitectura elegida, debiendo siempre permitir cuantificar la discrepancia entre la entrada y la salida. Como en cualquier DNN, el entrenamiento del modelo se basa en el ajuste de los parámetros entrenables de las capas del *autoencoder* para minimizar la función de pérdida. La figura 2.7 presenta una arquitectura simple de un *autoencoder* basado en FFNN.

Entre las aplicaciones de los *autoencoders* se destacan la reducción dimensional y la obtención de *embeddings*, representaciones enriquecidas de los datos que constan generalmente de menos dimensiones que la representación original. Además, los parámetros de un *autoencoder* previamente entrenado pueden ser empleados para inicializar los pesos de otra DNN, lo cual mejora el desempeño y convergencia de la misma [297].

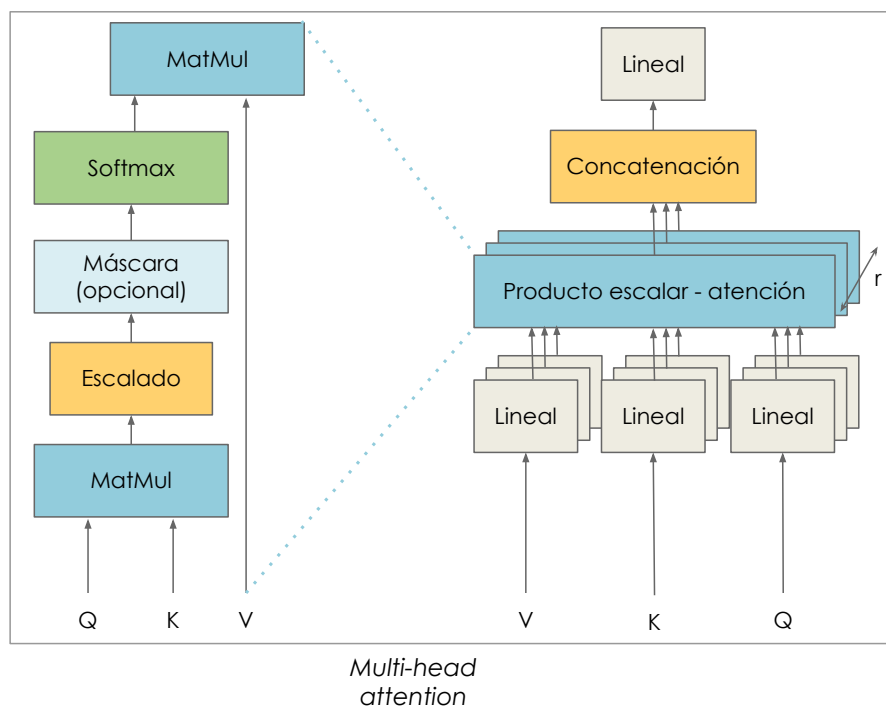


Figura 2.6: Diagrama esquemático de una capa de atención de  $r$  cabezales (*Multi-Head Attention*) y una vista en detalle del cómputo de los pesos de atención. Esta figura ha sido reproducida a partir de la originalmente propuesta por Vaswani et al. [397].

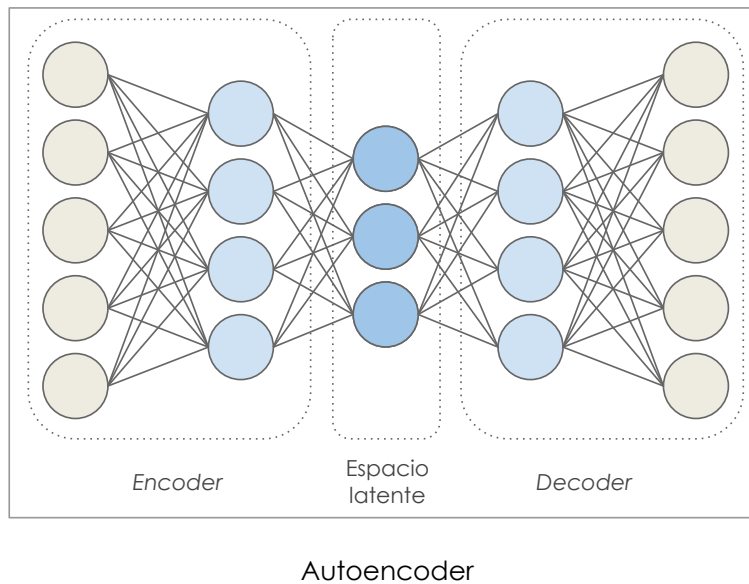


Figura 2.7: Arquitectura básica de un *autoencoder* basado en FFNN.

En particular, los *embeddings* [37] son representaciones vectoriales que capturan la estructura y semántica de los datos en un espacio numérico, generalmente de dimensionalidad reducida, el cual se denomina *espacio latente*. Dependiendo del *autoencoder* empleado para su aprendizaje, los *embeddings* presentan propiedades matemáticas especiales que les permiten expresar relaciones y significado entre los datos y sus partes por medio de operaciones sencillas, como el producto escalar o la suma. Al tratarse de representaciones numéricas ricas en información relevante, pueden ser empleados a su vez como entradas a modelos predictivos, los cuales pueden basarse en DNNs o en otros algoritmos de aprendizaje automático para clasificación o regresión.

### 2.5.3.6. Modelos generativos

Los modelos generativos basados en DNNs constituyen una clase avanzada de modelos capaces de generar instancias nuevas de datos de similares características y estructura a los datos de entrenamiento [288, 148, 327]. Estos modelos se caracterizan por sus topologías, generalmente dotadas de múltiples capas, que por diversos mecanismos de entrenamiento capturan propiedades e información semántica de los datos en distintos niveles de abstracción. Los modelos generativos han revolucionado profundamente el panorama presente y futuro de la inteligencia artificial, estando presentes en aplicaciones de éxito mundial como la serie GPT de OpenAI [52, 286], el modelo *DeepDream* de Google [265], o el modelo BERT [89], públicamente accesible a través de *Hugging Face*

*Transformers* [426]. Existen muchos tipos y arquitecturas de modelos generativos, entre los cuales destacan los modelos basados en redes neuronales recurrentes [140], los *autoencoders variacionales* (VAE, por sus siglas en inglés) [199], las *redes adversarias generativas* (GAN, por sus siglas en inglés) [127], y los modelos basados en *aprendizaje por refuerzo* [133, 272]. En particular, hacemos mención a los *autoencoders variacionales* (VAE) y a las *redes adversarias generativas* (GAN), con las cuales actualmente estamos experimentando para futuros trabajos derivados de la presente tesis doctoral.

Los *autoencoders* variacionales (VAE) [199] combinan elementos constitutivos de los *autoencoders* y conceptos de inferencia variacional. En contraste con los *autoencoders* tradicionales, un VAE no solamente condensa la información de entrada en un espacio latente de baja dimensionalidad, sino que además aprende una distribución probabilística gaussiana en dicho espacio. Durante su entrenamiento, el proceso de optimización consiste en aprender a reconstruir los datos de entrada por medio del par *encoder-decoder*, y a la vez la función de codificación produce dos vectores por cada instancia de datos en el espacio latente: un vector de medias  $\mu$  y un vector de desvíos estándar  $\sigma$ . Estos vectores definen una distribución gaussiana en el espacio latente alrededor de cada instancia de datos. La introducción de una distribución probabilística gaussiana en el espacio latente permite muestrear puntos de dicho espacio una vez entrenado el VAE. Al decodificar dichos puntos es posible obtener nuevas instancias de datos diferentes a las empleadas en el entrenamiento. Asimismo, dicha distribución asegura que puntos cercanos en el espacio latente preserven propiedades semánticas y estructurales similares, por lo que este tipo de modelos generativos es especialmente útil para la exploración de conjuntos potencialmente enormes de datos. La figura 2.8 muestra una representación abstracta de la arquitectura de un VAE basado en FFNN.

Las redes adversarias generativas (GAN) [127], ilustradas en la figura 2.9, constituyen un enfoque novedoso para el modelado generativo, en el que dos redes neuronales compiten entre sí: la red *generador* tiene por objetivo crear datos sintéticos a partir de ruido aleatorio y engañar a la red *discriminador*, cuyo objetivo es distinguir entre datos reales y generados por la red *generador*. El mecanismo de competencia entre las redes generador y discriminador optimiza iterativamente la capacidad del generador para crear instancias de datos que respeten las características semánticas y estructurales de los datos reales.

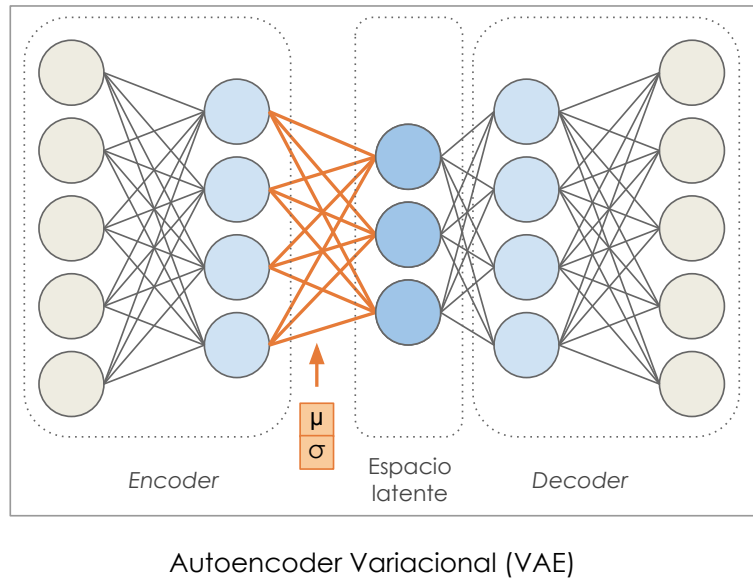


Figura 2.8: Arquitectura básica de un *autoencoder variacional* (VAE) basado en FFNN. El espacio latente codifica, además de la representación en baja dimensionalidad de la entrada, una distribución probabilística gaussiana con media  $\mu$  y desvío estándar  $\sigma$ .

## 2.6. Interpretabilidad y explicabilidad de modelos predictivos

En términos generales, cuando nos referimos a la explicabilidad o interpretabilidad de los modelos de aprendizaje automático y de las técnicas de inteligencia artificial, intuitivamente aludimos a la capacidad de entender los mecanismos que dichos modelos implementan para inferir o predecir a partir de los datos de entrada. Esta cualidad resulta importante por múltiples motivos, que abarcan la confianza en los resultados, la transparencia en el proceso de inferencia, la detección temprana de posibles sesgos, la mejora continua de las estrategias de modelado, la asistencia en el proceso de toma de decisiones críticas, e incluso el cumplimiento de normativas.

La explicabilidad de modelos de inteligencia artificial (XAI, por sus siglas en inglés) constituye una disciplina de investigación emergente y muy prolífica, por cuanto nuevos trabajos de investigación son publicados todos los años en la temática [16, 139, 58, 82, 225, 243, 188]. En esta disciplina, la terminología empleada para referirnos a la capacidad de interpretar o entender los modelos de inteligencia artificial no está totalmente estandarizada, siendo común encontrar los términos *explicabilidad* e *interpretabilidad* de modelos en contextos similares o incluso siendo utilizados como sinónimos [139, 404, 16, 94]. Esto se debe a que, taxonómicamente, las diferencias entre dichos



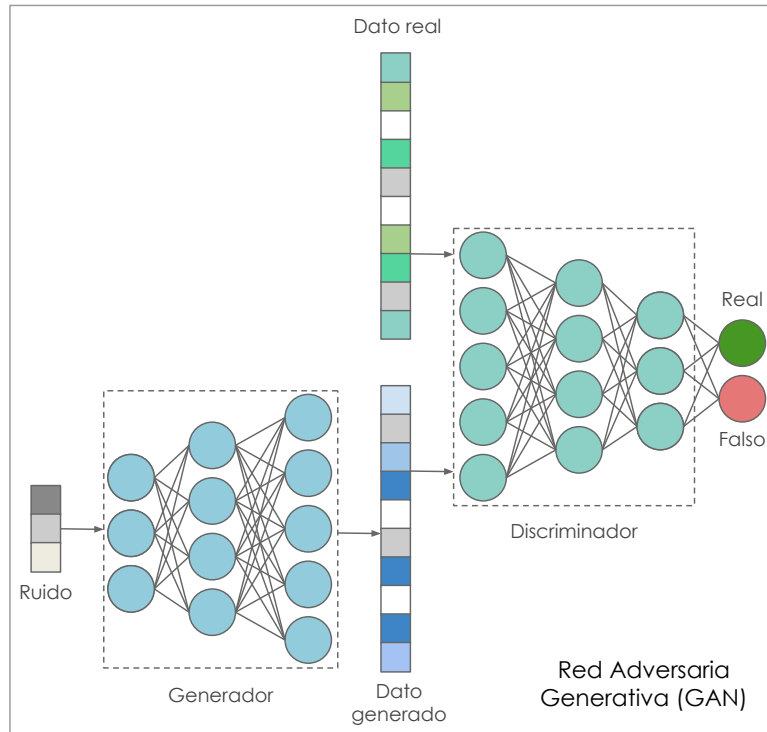


Figura 2.9: Arquitectura básica de una *red adversaria generativa* (GAN).

términos no están completamente definidas. Dichos conceptos resultan amplios y difusos en su definición, pudiendo ser analizados desde varios niveles de abstracción.

Aquellxs autorxs que trazan una distinción entre estos dos términos se basan en el planteo de que un modelo es *interpretable* cuando es capaz de proporcionar interpretaciones humanamente comprensibles de sus predicciones en términos matemáticos, algorítmicos o probabilísticos, deduciendo los alcances del resultado a partir de la comprensión de la estrategia de aprendizaje del modelo y de la forma en que el mismo transforma y opera internamente con los atributos de entrada para computar una predicción [16, 243]. Un modelo interpretable, según esta definición, permite analizar los mecanismos internos de aprendizaje e inferencia para lograr una comprensión a un bajo nivel de abstracción de las predicciones, por lo que, en síntesis, no es considerado un modelo de “caja negra”.

Por otro lado, un modelo *explicable* es definido como aquel cuyas predicciones pueden entenderse más allá de una comprensión algorítmica del proceso de inferencia [16, 243]. En este sentido, nos situamos en un nivel de abstracción mayor, en el que es posible interpretar la inferencia automática en términos de cómo los atributos o características que describen a las instancias de datos inciden en la predicción final, entendiendo al modelo como una “caja negra” cuyo proceso algorítmico nos

es, *a priori*, indistinto. Los modelos entendidos como explicables bajo esta definición admiten ser analizados con enfoques *a posteriori*, en función de sus variables de entrada y de los resultados obtenidos.

Más allá de los mecanismos que el modelo ofrezca para su análisis y comprensión, la interpretación de los resultados alcanzados por un modelo predictivo a nivel humano o experto requiere de un análisis contextual de los datos de entrada y de los atributos en términos de los cuales dichos datos están descritos, tarea que escapa a las capacidades propias de la estrategia de aprendizaje automático empleada, y constituye un nivel aún más alto de abstracción desde el cual entender la interpretabilidad de los modelos predictivos. En este sentido, resulta claro que las estrategias de reducción dimensional y de selección de características abonan a la comprensión de los modelos, por cuanto reducen el número de características que describen a los conjuntos de datos, facilitando la identificación de potenciales atributos determinantes en el proceso de modelado [98, 95, 108]. No obstante, la selección de características conlleva el potencial de pérdida de información relevante para el buen desempeño de los modelos, además de que, en determinados dominios, su semántica puede resultar oscura o ambigua, dificultando el proceso de selección [354].

Históricamente, los modelos de aprendizaje profundo han sido cuestionados en términos de interpretabilidad debido a su naturaleza altamente compleja y no lineal [16, 14, 158]. La complejidad de sus arquitecturas y topologías, junto con la gran cantidad de parámetros entrenables, la no linealidad de las transformaciones aplicadas a los datos por las funciones de activación, y la gran abstracción de las representaciones obtenidas por medio de las DNNs, atentan contra la interpretabilidad en bajo nivel de los mecanismos de inferencia. La interpretación precisa a nivel algorítmico de este tipo de modelos sigue siendo un área de investigación abierta y esencial para su aplicación en dominios críticos [14, 186]. Sin embargo, los modelos basados en DNNs tienen la capacidad de ser estudiados *a posteriori*, posibilitando visualizar la activación de las capas a partir de los datos de entrada y cuantificar el impacto de las características en la predicción final [280].

Eludiendo la disquisición sobre las diferencias entre los conceptos de *explicabilidad* e *interpretabilidad*, sobre las cuales no hay consenso en la comunidad experta en el área, en el contexto de la presente tesis empleamos mayoritariamente el término *interpretabilidad* para referirnos a los diferentes niveles de abstracción en los cuales podemos analizar la capacidad de describir el comportamiento o aprendizaje de un modelo de aprendizaje automático. La elección de dicho término se basa únicamente en la frecuencia de uso en la bibliografía científica.

## 2.7. Métricas de rendimiento

La selección de las métricas de rendimiento adecuadas es un paso fundamental para la evaluación de la efectividad de los modelos y la toma de decisiones en función de sus resultados [48, 165, 346, 305]. La evaluación de un modelo por medio de métricas inadecuadas puede sesgar la interpretación de los resultados obtenidos. Existen diferentes métricas que permiten cuantificar el rendimiento de un modelo, dependiendo de los diferentes aspectos del rendimiento que se deseen medir, del tipo de problema abordado (clasificación, regresión o *clustering*) y de las características particulares del conjunto de datos empleado en el modelado.

Para los problemas de clasificación, comúnmente se utilizan métricas que evalúan las predicciones en términos de cantidad de clasificaciones correctas y falsos positivos o falsos negativos. Por su parte, en los problemas de regresión las métricas buscan cuantificar la diferencia entre las predicciones del modelo y los valores reales de las instancias de datos. En el caso de problemas de *clustering* (aprendizaje no supervisado) se emplean métricas de distancia, densidad y entropía, que miden similitud entre instancias en el espacio de características, cantidad de instancias agrupadas y homogeneidad de los agrupamientos [5, 111].

En esta sección describimos el subconjunto de métricas de rendimiento empleadas en el desarrollo experimental de esta tesis, focalizando principalmente en métricas de evaluación para modelos de clasificación binaria y regresión. Además, proporcionamos una breve descripción de algunas métricas de distancia, comúnmente empleadas en problemas de aprendizaje no supervisado, las cuales fueron empleadas en etapas de análisis preliminares de conjuntos de datos o en la evaluación de proyecciones de baja dimensionalidad.

### Métricas para problemas de clasificación binaria

- Matriz de confusión o *Confusion Matrix*: en problemas de clasificación binaria, las dos clases a predecir se pueden asociar a los valores *positivo* (1) o *negativo* (0). Una matriz de confusión está conformada por cuatro componentes: verdaderos positivos o *True Positives* (**TP**), verdaderos negativos o *True Negatives* (**TN**), falsos positivos o *False Positives* (**FP**) y falsos negativos o *False Negatives* (**FN**). **TP** denota la cantidad de casos positivos correctamente clasificados, **TN** la cantidad de casos negativos correctamente clasificados, **FP** se corresponde con los casos negativos clasificados incorrectamente como positivos y **FN** con los casos positivos clasificados incorrectamente como negativos. La mayoría de las métricas descritas a continuación se computan a partir de los valores de la matriz de confusión.

- Exactitud o *Accuracy* ( $Acc$ ): expresa la proporción de predicciones correctas sobre el total de predicciones. Está acotada entre 0 y 1.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.20)$$

- Sensibilidad o *Sensitivity* ( $Sn$ ): se corresponde con la exactitud para los casos positivos, y expresa la capacidad del modelo de detectar TPs con respecto a todas las instancias etiquetadas como positivas, entre las que se incluyen FPs y TPs. Una denominación alternativa y equivalente para esta métrica es *Cobertura* o *Recall*, la cual es también ampliamente utilizada en la literatura. Esta métrica está acotada entre 0 y 1.

$$Sn = \frac{TP}{TP + FN} \quad (2.21)$$

- Especificidad o *Specificity* ( $Sp$ ): se corresponde con la exactitud para los casos negativos, y expresa la capacidad del modelo de detectar TNs con respecto a todas las instancias etiquetadas como negativas. Está acotada entre 0 y 1.

$$Sp = \frac{TN}{TN + FP} \quad (2.22)$$

- Precisión o *Precision*: esta métrica corresponde a la proporción de TPs detectados por el modelo con respecto a la totalidad de instancias predichas como positivas por el modelo, entre las que se incluyen TPs y FPs. Está acotada entre 0 y 1.

$$\text{precision} = \frac{TP}{TP + FP} \quad (2.23)$$

- Exactitud Balanceada, Tasa de No-Error o *Balanced Accuracy, Non-Error Rate* ( $BAcc, NER$ ): esta métrica se computa como la media aritmética entre la sensibilidad ( $Sn$ ) y la especificidad ( $Sp$ ), está acotada entre 0 y 1. Es empleada principalmente en escenarios de desbalance de clases, en contraposición a la métrica de exactitud ( $Acc$ ), que no resulta adecuada en dichos escenarios.

$$BAcc = \frac{Sn + Sp}{2} \quad (2.24)$$

- Puntaje F1 o *F1-score* ( $F1$ ): se computa como la media armónica entre la precisión (P) y sensibilidad o cobertura (Sn o *recall*), está acotada entre 0 y 1. Es adecuada para escenarios de desbalance de clases.

$$F1 = \frac{2 \times \text{precision} \times \text{Sn}}{\text{precision} + \text{Sn}} \quad (2.25)$$

- Puntaje H1 o *H1-score* ( $H1$ ): esta métrica no es una de las métricas estándar de la literatura en aprendizaje automático, sino que fue propuesta por nuestro equipo de investigación en el marco de la presente tesis doctoral [322]. Se computa como la media armónica entre la sensibilidad o cobertura (Sn o *recall*) y la especificidad (Sp), está acotada entre 0 y 1.

$$H1 = \frac{2 \times \text{Sn} \times \text{Sp}}{\text{Sn} + \text{Sp}} \quad (2.26)$$

- Área bajo la curva ROC (*AUC-ROC*): la curva ROC (*Receiver Operating Characteristic*) representa la proporción de TPs ( $Sn$ ) con respecto a la tasa de FPs ( $1 - Sp$ ) para diferentes umbrales de decisión. Intuitivamente, el área bajo la curva ROC mide la capacidad del modelo para distinguir entre las clases positiva y negativa. La métrica AUC-ROC está acotada entre 0 y 1. Un valor de AUC-ROC cercano a 0 indica que el modelo predice la mayoría de las instancias de forma incorrecta. Un valor de AUC-ROC cercano a 0,5 sugiere un rendimiento aleatorio, donde el modelo azarosamente asigna etiquetas a las instancias de datos, mientras que cuanto más se acerca a 1 representa un modelo de mejor rendimiento predictivo.

## Métricas para problemas de regresión

- Error Cuadrático Medio o *Mean Squared Error* ( $MSE$ ): corresponde a la media de las diferencias al cuadrado entre las predicciones realizadas por el modelo y los valores de etiqueta reales de las instancias. Penaliza fuertemente los errores grandes.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.27)$$

- Raíz del Error Cuadrático Medio o *Root Mean Squared Error* ( $RMSE$ ): se computa como la raíz cuadrada del MSE y proporciona un valor expresado en la misma unidad que el valor a predecir, lo cual facilita su interpretación.

$$RMSE = \sqrt{MSE} \quad (2.28)$$

- Error Absoluto Medio o *Mean Absolute Error (MAE)*: es la media de las diferencias absolutas entre las predicciones realizadas por el modelo y los valores de etiqueta de las instancias. Es robusta en escenarios donde hay presencia de valores atípicos en el conjunto de datos, en contraposición al MSE.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.29)$$

- Coeficiente de Determinación o *R-squared ( $R^2$ )*: esta métrica indica la proporción de la varianza en el valor a predecir que puede ser efectivamente predicha por el modelo a partir de las características o atributos del conjunto de datos. Esta métrica está acotada entre 0 y 1, donde 1 indica un ajuste perfecto.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (2.30)$$

En estas ecuaciones,  $n$  es el número total de instancias;  $y_i$  es el valor de etiqueta real de la instancia  $i$ -ésima;  $\hat{y}_i$  es la predicción del modelo para la instancia  $i$ -ésima; y  $\bar{y}$  es la media aritmética de los valores de etiqueta reales de todas las instancias del conjunto de datos.

## Métricas de distancia

- Distancia Euclídea: Intuitivamente, puede interpretarse como la distancia *en línea recta* entre dos puntos en un espacio euclidiano de  $n$  dimensiones. La inversa de la distancia euclídea es empleada como métrica de similitud entre dos instancias. La fórmula de la distancia euclídea en  $n$  dimensiones para dos instancias  $\mathbf{A} = (a_1, a_2, \dots, a_n)$  y  $\mathbf{B} = (b_1, b_2, \dots, b_n)$  es:

$$\text{Distancia Euclídea} = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \quad (2.31)$$

donde  $a_i$  y  $b_i$  representan el valor de la característica  $i$ -ésima que describe a las instancias de datos  $\mathbf{A}$  y  $\mathbf{B}$ , característica coincidente con la dimensión  $i$ -ésima del espacio euclídeo.

- Distancia Coseno: esta métrica de distancia representa la diferencia angular entre dos vectores en un espacio  $n$ -dimensional, conformado por las  $n$  características que describen al conjunto de datos. La distancia coseno brinda una medida de similitud entre dos instancias de datos en

términos de la orientación de sus vectores  $n$ -dimensionales, en lugar de analizar su magnitud. Un valor de distancia 1 indicaría que los vectores tienen idéntica orientación, mientras que un valor 0 indicaría que los vectores son ortogonales. Para dos vectores  $\mathbf{A}$  y  $\mathbf{B}$ , la distancia coseno se calcula como:

$$\text{Distancia Coseno} = 1 - \frac{A \cdot B}{\|A\| \|B\|} \quad (2.32)$$

donde  $A \cdot B$  es el producto escalar de  $A$  y  $B$ , y  $\|A\|$  y  $\|B\|$  son las magnitudes de los vectores  $A$  y  $B$ .

- Distancia de *Jaccard* o Índice de *Tanimoto*: esta métrica de distancia se emplea comúnmente para medir la diferencia entre agrupamientos. La distancia de Jaccard varía entre 0 y 1, donde 0 indica conjuntos idénticos y 1 indica conjuntos disjuntos. Para dos conjuntos  $A$  y  $B$ , la distancia de Jaccard se calcula como:

$$\text{Distancia de Jaccard} = 1 - \frac{|A \cap B|}{|A \cup B|} \quad (2.33)$$

donde  $|A \cap B|$  es la cardinalidad del conjunto resultante de la intersección entre  $A$  y  $B$ , mientras que donde  $|A \cup B|$  es la cardinalidad del conjunto resultante de la unión entre  $A$  y  $B$ .

## 2.8. Síntesis y conclusiones

En este capítulo brindamos definiciones y explicaciones en un alto nivel de abstracción de los conceptos básicos de aprendizaje automático y profundo que dan sustento al diseño y desarrollo experimental de la presente tesis. En el capítulo 3 presentamos conceptos y definiciones asociados a informática molecular, los cuales pretenden abonar a la comprensión básica del contexto particular de aplicación de los desarrollos presentados en esta tesis a aquellxs lectorxs no familiarizadxs con las especificidades del área.

# Capítulo 3

## Informática molecular

En este capítulo exploramos de forma sintética los conceptos fundamentales de la informática molecular, enumerando principios en los que se basa el desarrollo teórico-experimental de la presente tesis doctoral. Formulamos algunos de los fundamentos básicos de la química medicinal que sustentan la toma de decisiones expertas, para luego adentrarnos en el estudio de diferentes subdisciplinas de la informática molecular, enfatizando en cómo estos enfoques son empleados para acelerar y optimizar el proceso de desarrollo de nuevos fármacos.

---

### 3.1. Introducción

La *informática molecular* es una disciplina que fusiona conceptos y elementos de informática, química medicinal, farmacia y biología para comprender y modelar fenómenos biológicos y bioquímicos a nivel molecular. Esta área se basa en el estudio y aplicación de algoritmos y métodos computacionales para la simulación de interacciones entre estructuras moleculares, la predicción de actividad biológica de los compuestos químicos, el análisis de procesos biológicos complejos y el modelado de funciones moleculares en los organismos vivos, entre otras tareas [110, 175]. La informática molecular constituye hoy en día un pilar fundamental en el proceso de diseño de medicamentos, mejorando su precisión, eficacia y eficiencia.

Desde sus orígenes en la década de 1960, con el surgimiento de técnicas informáticas para el análisis de datos biológicos y moleculares [422], la informática molecular ha evolucionado de manera significativa y vertiginosa hasta convertirse en una de las áreas de investigación interdisciplinarias más prominentes de la actualidad. Esto se debe primordialmente al rápido avance en la investigación



médica y farmacológica, que ha permitido generar enormes cantidades de datos bioquímicos, sumado a la explosión de nuevas y eficaces estrategias computacionales de aprendizaje automático e inteligencia artificial, capaces de procesar grandes volúmenes de datos y modelar relaciones complejas y no lineales entre datos heterogéneos [100, 237, 193, 257].

El desarrollo de fármacos se ha transformado drásticamente a lo largo de las últimas décadas y, en la actualidad, consiste en un proceso complejo y desafiante que integra conocimiento experto de múltiples disciplinas. A grandes rasgos podemos identificar una serie de etapas clave en el proceso de desarrollo de fármacos, ilustradas a modo sintético en la figura 3.1:

1. Descubrimiento de blancos terapéuticos: esta primera etapa involucra la identificación, estudio y validación de moléculas, proteínas o genes relacionados con un proceso biológico determinado, para el cual resulta necesario el desarrollo de un fármaco. En esta instancia, las estrategias computacionales asisten en el análisis de datos genómicos y proteómicos y en la simulación de las estructuras proteínicas asociadas a los mismos. Por medio de esta etapa es posible descubrir sobre qué blancos terapéuticos debería actuar un potencial fármaco y qué tipo de acciones farmacológicas debería ejercer sobre el mismo.
2. Descubrimiento de compuestos candidatos: en esta etapa, el objetivo es identificar compuestos químicos candidatos, capaces de interactuar efectivamente con el blanco terapéutico identificado previamente. Considerando la magnitud potencialmente infinita del espacio molecular a explorar, en esta etapa resulta indispensable la aplicación de técnicas computacionales de *cribado virtual de fármacos* y de diseño de moléculas asistido por computadora, como el diseño *de novo*. Estas técnicas, explicadas en mayor detalle en las secciones 3.3.3 y 3.3.5 del presente capítulo, indudablemente requieren de la integración de conocimiento experto de forma transversal y emplean enormes bases de datos de compuestos químicos.
3. Optimización de compuestos: una vez identificados, los compuestos candidatos son sometidos a una serie de evaluaciones intensivas y pasos de optimización que permiten validar su viabilidad farmacológica, eficacia y seguridad. En esta etapa, las herramientas computacionales de aprendizaje automático juegan un rol fundamental en la predicción de propiedades farmacocinéticas de los compuestos candidatos, así como en la predicción de su perfil de bioactividad en relación con el blanco terapéutico en estudio. El modelado predictivo de la *Relación Cuantitativa entre Estructura y Actividad* (modelado *QSAR*, por sus siglas en inglés) constituye una de las áreas más prolíficas de investigación en quimioinformática [68, 316], y

es uno de los ejes centrales del desarrollo de esta tesis doctoral, siendo abordada en la sección 3.3.2 del presente capítulo.

4. Fase preclínica: una vez seleccionados y validados los compuestos candidatos, se procede a sintetizarlos químicamente en laboratorios y a realizar experimentos *in vitro* (empleando cultivos de tejidos vivos o bacterias, fuera de un organismo vivo) e *in vivo* (dentro de organismos vivos intactos) para estudiar su posología y asegurar su eficacia y seguridad. Las estrategias computacionales aplicadas en las etapas previas, comúnmente denominadas experimentación *in silico*, son determinantes para obtener una mayor cantidad de compuestos candidatos y para garantizar mejores resultados en esta etapa. La fase preclínica es fundamental para la toma de decisiones sobre la viabilidad de realizar ensayos clínicos en humanxs.
5. Ensayos clínicos: si los compuestos candidatos pasan la fase preclínica, se procede a realizar una serie de ensayos clínicos en humanxs. Estos ensayos consisten en tratamientos experimentales con el fármaco candidato, los cuales son realizados en múltiples fases bajo supervisión médica y siguiendo rigurosos protocolos que buscan proteger la integridad del procedimiento.
6. Aprobación y comercialización: si los ensayos clínicos son exitosos, el compuesto candidato se convierte efectivamente en un medicamento y se somete a regulación y aprobación. Los resultados del proceso se presentan a las agencias gubernamentales reguladoras, como la *FDA* en Estados Unidos o la *EMA* en Europa [74, 396], las cuales revisan el procedimiento y, de otorgar su aprobación, el medicamento puede ser comercializado de forma pública. Esta etapa implica además la producción, distribución y promoción del medicamento.
7. Farmacovigilancia: la farmacovigilancia es el nombre que recibe el proceso de seguimiento y evaluación continua de la seguridad de los medicamentos una vez que se lanzan al mercado. El objetivo de las estrategias de farmacovigilancia es prevenir y detectar efectos adversos no previstos de los medicamentos recién aprobados. En esta etapa, se emplean herramientas computacionales que permiten recopilar grandes volúmenes de datos de distintas fuentes para su análisis y la predicción de efectos adversos.

El proceso de desarrollo de fármacos no está exento de desafíos. Tal y como se puede apreciar en la figura 3.1, cada una de las etapas descritas requiere de mucho tiempo y recursos económicos, para una tasa de éxito muy baja. La inversión promedio en el proceso completo de desarrollo de un nuevo medicamento es del orden de los 2.500 millones de dólares y de entre 10 y 15 años [64, 181, 90, 362]. Además, tiene una *tasa de deserción* muy alta: la mayoría de los compuestos candidatos fracasan

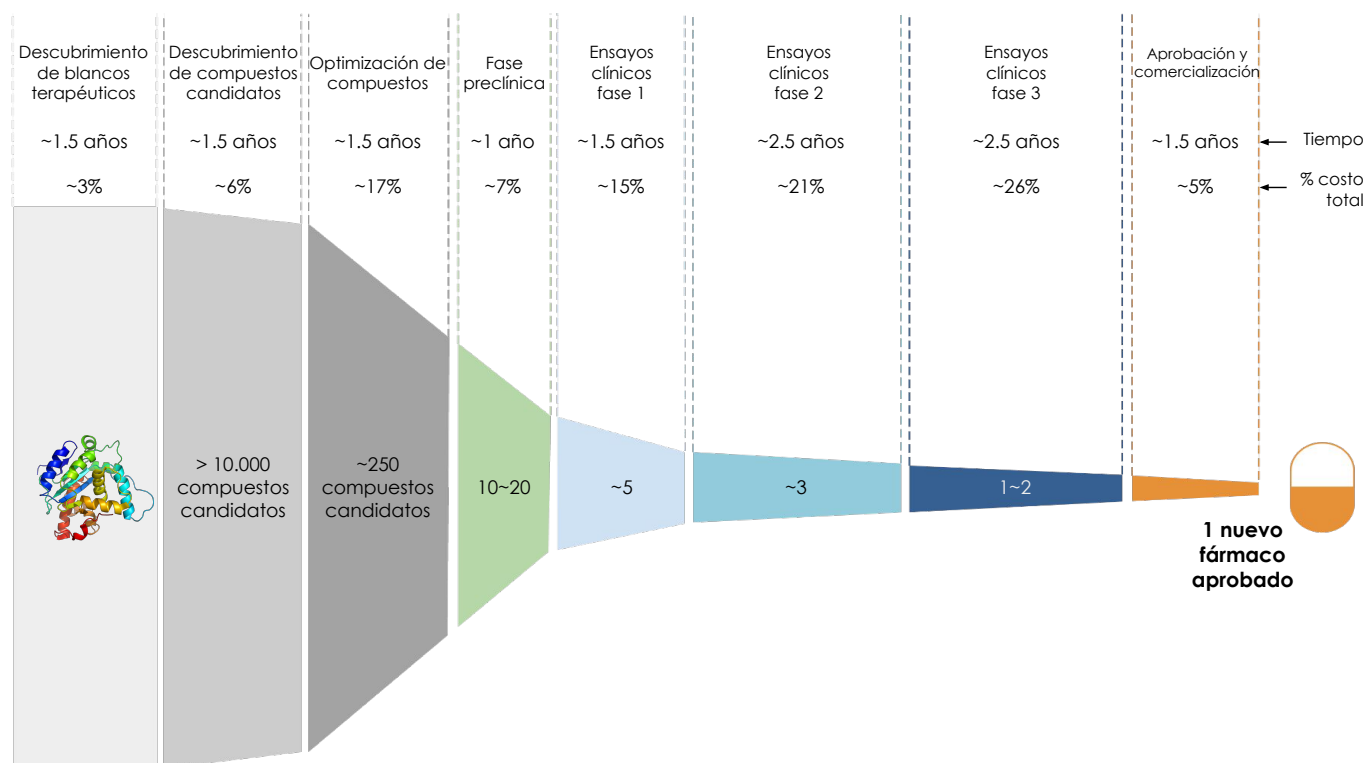


Figura 3.1: Representación esquemática del proceso de descubrimiento de fármacos, donde se muestra la aguda tasa de deserción y el tiempo y costos involucrados. Según estudios recientes en la materia, el costo estimado de desarrollo de un nuevo fármaco es del orden de los 2.500 millones de dólares [362].

en las fases preclínicas y clínicas, lo que implica cuantiosas pérdidas económicas y de esfuerzos de investigación. Gracias a la aplicación de estrategias computacionales en etapas tempranas del proceso, se comienza con números del orden de los miles de compuestos candidatos, de los cuales con suerte prosperan uno o dos para la etapa de ensayos clínicos [237, 362]. En este contexto, resulta claro que la introducción de estrategias computacionales constituye un punto de inflexión en la historia del desarrollo de fármacos y es un pilar fundamental para lograr procesos eficientes, precisos, más rápidos y menos onerosos.

## 3.2. Conceptos básicos de química medicinal

En esta sección brindaremos definiciones breves y en un alto nivel de abstracción de algunos conceptos básicos del campo de la química medicinal, los cuales sirven de sustento a muchas de

las decisiones tomadas en el diseño experimental de los trabajos enmarcados en la presente tesis doctoral. En primera instancia, describiremos las propiedades *farmacocinéticas* y *farmacodinámicas* de los compuestos químicos, vinculadas a los mecanismos de interacción de los medicamentos con los organismos vivos y su acción terapéutica. Además, brindaremos una definición breve del concepto de *bioactividad* y de las posibles acciones farmacológicas de los medicamentos. Luego describiremos algunos determinantes químicos de la afinidad de un compuesto candidato con las características deseables en un fármaco (*drug-likeness*) y por último discutiremos el principio de similitud estructural, el cual sustenta muchos de los algoritmos de representación molecular y modelado predictivo en el dominio.

### 3.2.1. Propiedades farmacocinéticas y farmacodinámicas

La *farmacocinética* y la *farmacodinámica* son dos áreas cruciales de la farmacología y la química medicinal que estudian los mecanismos de interacción de los fármacos con los organismos vivos [53]. El objetivo de la *farmacocinética* es estudiar cuáles son los factores que inciden en los niveles de concentración de un fármaco a lo largo del tiempo y en diferentes tejidos o partes del organismo, lo cual permite analizar la dosificación y garantizar su eficacia y seguridad. La farmacocinética estudia cinco propiedades fundamentales de los compuestos químicos, denominadas *propiedades ADME-Tox*: Absorción, Distribución, Metabolismo, Excreción y Toxicidad [101].

- Absorción: propiedad vinculada al proceso de ingreso de un fármaco al torrente sanguíneo desde la vía de administración, fuertemente asociada a la *biodisponibilidad* del fármaco. Factores tales como la solubilidad y la estabilidad del medicamento en medios adversos (como el tracto intestinal y estomacal, por ejemplo) son determinantes para esta propiedad.
- Distribución: propiedad asociada a la diseminación del medicamento en los diferentes medios, tejidos y fluidos corporales. El peso molecular y la polaridad molecular de los compuestos son factores influyentes en esta propiedad de relevancia, que determina el alcance, dosificación y efectividad de la droga.
- Metabolismo: propiedad asociada a la transformación química de un fármaco en el organismo, que determina su actividad, vida media y eliminación. En el organismo, el metabolismo de algunos medicamentos puede verse afectado si existen otros fármacos que alteren las enzimas responsables de dicho proceso, por lo que esta propiedad está vinculada al estudio de la interacción medicamentosa.

- **Excreción:** propiedad relacionada con la capacidad de eliminación del fármaco del organismo, generalmente a través de la orina o las heces. Esta propiedad resulta importante, puesto que, de no ser completamente eliminable, un medicamento podría acumularse en ciertos tejidos orgánicos causando efectos adversos o toxicidad.
- **Toxicidad:** esta propiedad se vincula con el potencial de un medicamento para causar efectos perniciosos en el organismo. La toxicidad de un fármaco puede estudiarse a nivel del organismo vivo o al nivel de tejidos o células. Asimismo, hay diferentes formas de evaluar la toxicidad de una sustancia, teniendo en cuenta los niveles de concentración en tejidos, la exposición, la severidad de los efectos adversos o el tipo de daño que producen, entre otros factores.

La *farmacodinámica* estudia los mecanismos de interacción del fármaco con los blancos terapéuticos y biológicos en el organismo. Estos blancos terapéuticos pueden ser receptores, enzimas o canales iónicos, cuya identificación y validación es fundamental en la primera etapa del proceso de descubrimiento de fármacos. La farmacodinámica analiza cómo la interacción entre el fármaco y estos receptores modifica distintas funciones biológicas, lo que se traduce en efectos terapéuticos o adversos [53].

La forma en la que un fármaco interactúa con uno o más blancos biológicos en el organismo determina su perfil de *bioactividad*. La bioactividad de un compuesto está vinculada no solo a su capacidad de ligar efectivamente con un receptor, sino a la acción terapéutica o farmacológica que ejerce sobre dicho receptor. Sin ánimos de ser exhaustivos, listamos a continuación algunas de las acciones farmacológicas más estudiadas:

1. **Agonismo:** un fármaco agonista *activa* un receptor o enzima, desencadenando una respuesta biológica similar a la producida por una molécula endógena, propia del organismo.
2. **Antagonismo:** un fármaco antagonista *bloquea o inhibe* la actividad de un receptor o enzima, impidiendo la respuesta biológica normal.
3. **Inhibición enzimática:** en estos casos, el fármaco inhibe la actividad de una enzima específica, lo que interfiere con ciertos procesos metabólicos.
4. **Modulación alostérica:** en estos casos el fármaco se une a un sitio *alostérico* del receptor o enzima, que son secciones de la proteína alejadas de los sitios activos (donde normalmente el fármaco debería unirse) alterando su actividad sin afectar el sitio activo.

5. Bloqueo de canal iónico: los canales iónicos en la membrana celular intervienen en la transmisión de señales eléctricas, por lo que su bloqueo modifica la actividad de las células musculares o neuronas.
6. Interferencia de la síntesis de ADN/ARN: se trata de fármacos que inhiben la replicación, transcripción o traducción de ADN o ARN en las células, lo cual es fundamental en tratamientos antivirales.

Dos conceptos clave vinculados a farmacocinética y farmacodinámica son la *biodisponibilidad* y la *concentración* del fármaco. La biodisponibilidad se vincula con la velocidad a la que un fármaco o compuesto químico es absorbido por los tejidos y está presente en el torrente sanguíneo luego de su administración. Este factor es crítico en la determinación de la eficacia de los medicamentos y en la toma de decisiones sobre su vía de administración (oral, intravenosa, tópica, etc.) [384]. La concentración de un fármaco se relaciona con la determinación de la cantidad de dicho fármaco que debe ser administrada para causar el efecto terapéutico deseado. Existen múltiples métricas de concentración ampliamente utilizadas, entre las que destacamos la *Concentración Inhibitoria 50* (*IC50*, por sus siglas en inglés), y la *Concentración Efectiva 50* (*EC50*, por sus siglas en inglés), las cuales son empleadas para determinar la concentración de un fármaco necesaria para inhibir o activar el 50% de una respuesta biológica específica, respectivamente. Cuanto más bajo sea el nivel de *IC50* o *EC50*, según se trate de un fármaco antagonista o agonista, respectivamente, más potente será considerado el mismo [244].

### 3.2.2. *Drug-likeness*

El término en inglés *drug-likeness* se refiere a la afinidad de un compuesto químico con aquellas características deseables en un fármaco [6]. Un compuesto en principio tiende a exhibir ciertas propiedades físico-químicas que lo vuelven más probable a ser un buen candidato a medicamento. Estas propiedades han sido extensamente estudiadas en la literatura del área [406, 403, 391, 372] y se relacionan con la farmacocinética y farmacodinámica del compuesto. Existen reglas o heurísticas que los expertos siguen a la hora de diseñar nuevos fármacos, orientadas a garantizar su *drug-likeness*. Entre las más importantes enumeramos:

- Regla de los Cinco de Lipinski (*Lipinski's RO5*) [227]: La regla de los cinco de Lipinski es un conjunto de criterios formulados por Lipinski [227] para evaluar la afinidad de un compuesto químico con las características propias de un candidato viable a fármaco. Si bien no se trata

de una regla rígida, considerando que muchos fármacos exitosos no cumplen con todos sus criterios, esta regla es universalmente aceptada en el ámbito de la investigación farmacéutica y se utiliza para filtrar y priorizar compuestos candidatos en etapas tempranas del proceso de desarrollo de fármacos. Los criterios señalados por la regla de los cinco de Lipinski involucran cuatro características moleculares, a saber:

1. Cantidad de aceptadores de puentes de hidrógeno (HBA): los aceptadores de puentes de hidrógeno son grupos químicos en una molécula capaces de aceptar átomos de hidrógeno de otras sustancias en interacciones químicas. Un exceso de grupos aceptadores puede afectar la capacidad del compuesto para ser absorbido y distribuido en el cuerpo. La RO5 establece que un compuesto candidato debe tener menos de cinco grupos aceptadores.
  2. Cantidad de donantes de puentes de hidrógeno (HBD): los donantes de puentes de hidrógeno son grupos funcionales que tienen la capacidad de donar átomos de hidrógeno en interacciones químicas. Un exceso de grupos donantes también puede influir en la biodisponibilidad y en la capacidad del compuesto para cruzar membranas celulares. Según la RO5, un fármaco candidato debe contar con menos de diez grupos donantes.
  3. Peso molecular (MW): el peso molecular es la suma de las masas atómicas de los átomos que conforman una molécula, y se expresa en unidades de masa atómica (*uma*) o daltons (*Da*). Los compuestos con altos pesos moleculares son más propensos a la acumulación en tejidos, vinculada a la toxicidad, además de dificultar su distribución, absorción y solubilidad. Un compuesto candidato debe exhibir un peso molecular menor a 500*Da*, según la RO5.
  4. Logaritmo del coeficiente de partición octanol-agua ( $\log P$ ): el coeficiente de partición octanol-agua se utiliza para medir la lipofilia de un compuesto, como indicador de cuán soluble es una sustancia en octanol (una sustancia lipofílica) en comparación con su solubilidad en agua (una sustancia hidrofílica). La lipofilia de las sustancias afecta la forma en la que éstas traspasan las distintas membranas y medios acuosos de las células, lo que influye en su absorción y biodisponibilidad. Los valores de  $\log P$  aceptables dependen de la propiedad o bioactividad en estudio; no obstante como regla general se establece que los valores de  $\log P$  menores a 5 están asociados a buenos compuestos candidatos.
- Puntaje QED (*Quantitative Estimate of Drug-likeness*) [44]: es un indicador que se utiliza para evaluar *drug-likeness* basado en modelos y cálculos computacionales que consideran múltiples propiedades del compuesto, tales como su peso molecular, su solubilidad y lipofilia, su

polaridad y otros factores. Un puntaje QED alto indica una mayor probabilidad de que el compuesto sea farmacológicamente viable y seguro. El puntaje QED oscila entre 0 y 1.

- Puntaje de accesibilidad sintética (*Synthetic Accessibility score*) [105]: es una métrica que permite estimar la viabilidad de síntesis de un compuesto químico en el laboratorio. Este puntaje se computa en términos de la complejidad estructural de un compuesto y la disponibilidad de reactivos y rutas de síntesis conocidas. La sintetizabilidad de un compuesto implícitamente está asociada a sus propiedades farmacocinéticas y farmacodinámicas, puesto que las estructuras moleculares complejas exhiben a menudo mayores dificultades para ser distribuidas y absorbidas por los tejidos. Un compuesto es considerado más fácil de sintetizar cuanto menor sea su puntaje de accesibilidad sintética, variando este puntaje entre 1 y 10.
- Constante de disociación ácida ( $K_a$ ) y Constante de disociación básica ( $K_b$ ): son medidas cuantitativas de la fuerza de un ácido (o base) en una solución química. Son buenos indicadores de la capacidad de un compuesto de fármaco para ingresar al torrente sanguíneo y acumularse en tejidos o secreciones.

### 3.2.3. Principio de similitud estructural

El *principio de similitud estructural*, concepto fundamental en el diseño de fármacos, establece que compuestos químicos con estructuras moleculares similares tienden a exhibir propiedades físico-químicas y perfiles farmacológicos similares [35, 31]. La similitud estructural entre compuestos químicos puede ser entendida desde de la presencia y disposición de ciertos átomos en sus estructuras hasta la presencia de grupos funcionales específicos. Este principio resulta esencial para identificar compuestos candidatos con potencial farmacológico, ya que el análisis de las estructuras moleculares de fármacos aprobados o de compuestos que son ligandos conocidos a ciertos blancos terapéuticos permite guiar la exploración de nuevos compuestos candidatos [249].

No obstante, la determinación de la similitud entre compuestos químicos constituye un proceso complejo y no es infalible. Los determinantes estructurales de similitud están siempre supeditados al blanco farmacológico objetivo, a la complejidad de las interacciones biológicas en el organismo y a las posibles interacciones no esperadas que dichos compuestos puedan tener con otros blancos terapéuticos. El estudio de similitud entre compuestos químicos constituye un área de investigación prominente, en la que las estrategias computacionales juegan un rol fundamental, existiendo una gran variedad de métodos computacionales y de modelado molecular que permiten analizar y comparar las estructuras químicas y predecir la similitud estructural entre compuestos.



### 3.3. Informática molecular y diseño de fármacos asistido por computadora

Como hemos discutido en el transcurso del presente capítulo, la aplicación de estrategias computacionales ha revolucionado el proceso de descubrimiento de nuevos medicamentos. Estas estrategias aprovechan los avances tecnológicos en inteligencia artificial y aprendizaje automático para acelerar y optimizar el proceso de diseño de fármacos, consolidando una plataforma eficaz, eficiente y rentable para el diseño de nuevas terapias farmacológicas. En esta sección exploramos una serie de subdisciplinas en auge, las cuales constituyen, de forma independiente, áreas de investigación prolíficas y necesarias en la carrera por la mejora continua del proceso de diseño de fármacos asistido por computadora.

La *representación computacional de compuestos químicos* constituye una de las piedras angulares del área, necesaria para condensar información heterogénea de datos químicos en formatos manipulables por los algoritmos de aprendizaje. A partir de dichas representaciones es posible desarrollar modelos predictivos, fundamentales para determinar el perfil de bioactividad de compuestos candidatos. El modelado de *Relación Cuantitativa Estructura-Actividad* (*Quantitative Structure-Activity Relationship* o QSAR, por sus siglas en inglés) constituye uno de los pilares sobre los que se sustenta la informática molecular y el proceso moderno de descubrimiento de fármacos.

Por su parte, las técnicas de *cribado virtual de fármacos* (*virtual screening*) permiten la exploración de espacios potencialmente infinitos de compuestos químicos de forma eficiente, facilitando la identificación de subespacios moleculares prometedores. Como soporte a estas técnicas, las estrategias de analítica visual habilitan la exploración visual de espacios químicos de alta dimensionalidad y asisten a los expertos en la toma de decisiones y el proceso de diseño *de novo* de nuevas estructuras moleculares.

#### 3.3.1. Representación computacional de compuestos químicos

La representación de compuestos químicos constituye una disciplina de investigación histórica. Desde la introducción de las fórmulas estructurales a mediados del siglo XIX, pasando por las proyecciones de Fischer [266] y por los más recientes modelos tridimensionales [26], se han realizado numerosos esfuerzos por encontrar abstracciones para representar la complejidad del espacio químico. Con el auge de la informática, surgieron múltiples algoritmos que permiten computar representaciones moleculares ricas en información estructural, funcional y química [20, 421]. Las representaciones

computacionales de compuestos químicos capturan información sobre la topología, la geometría y la carga de las moléculas, y resultan necesarias para poder suministrar información de entrada a cualquier modelo de aprendizaje automático, sea cual sea su propósito específico, constituyendo un factor determinante en la confiabilidad y el desempeño de dichos modelos [421].

El enfoque convencionalmente adoptado en el diseño de representaciones moleculares involucra la ingeniería de atributos, proceso que consiste en la identificación o cómputo manual o semi-automatizado de características y propiedades relevantes. Otros enfoques tradicionalmente empleados consisten en algoritmos que analizan la estructura molecular como si fuera un grafo, cuyos nodos son átomos y sus enlaces arcos, para detectar patrones estructurales que luego son codificados en atributos de la representación final. Estos enfoques han dado lugar a representaciones moleculares ampliamente utilizadas en el dominio del diseño de fármacos [378, 62], denominadas representaciones *tradicionales*, entre las que destacan las representaciones lineales [418, 207, 152], los descriptores moleculares [378] y los *fingerprints* [311, 99, 86].

Más allá de la popularidad y la convención de uso de dichas representaciones tradicionales en la literatura [421], el proceso de ingeniería de atributos requiere de la aplicación de conocimiento experto en el dominio químico. Por ejemplo, la elaboración de representaciones basadas en descriptores moleculares requiere de experticia en química medicinal para lograr una correcta interpretación semántica y química de los descriptores. Más aún, cada representación molecular tradicional codifica información de distinta naturaleza, por lo que resultan complementarias y no existe una única representación óptima para todas las tareas [134, 332].

En los últimos años, han proliferado los métodos de representación de compuestos químicos derivados de algoritmos de aprendizaje automático y aprendizaje profundo [103, 73, 20, 421, 26]. Estas representaciones, denominadas *embeddings moleculares*, resultan versátiles y adaptables a diferentes tareas, ya que al ser obtenidas por medio de procesos automáticos de aprendizaje pueden computarse a partir de fuentes de información de diversa naturaleza, condensando distintos aspectos de la estructura y función molecular en una representación compacta [73].

En esta sección exploraremos las representaciones tradicionales más ampliamente utilizadas y el concepto de *embeddings* moleculares, analizándoles desde sus algoritmos y estrategias de cómputo, su viabilidad para atravesar procesos de selección de características y su *invertibilidad*, es decir, la posibilidad de reconstrucción del compuesto original a partir de la representación obtenida.

### 3.3.1.1. Representaciones lineales

Las representaciones lineales de compuestos químicos codifican la estructura molecular en secuencias de caracteres. Estas representaciones son las más ampliamente utilizadas en informática molecular, siendo posible encontrar bases de datos moleculares comprendiendo millones de compuestos químicos almacenados en estos formatos [360, 196, 116]. Esto se debe a múltiples factores, entre ellos a su simplicidad e interpretabilidad, a su eficiencia computacional y a su interoperabilidad en múltiples aplicaciones y plataformas de química medicinal. En las representaciones lineales, la información molecular se presenta de manera secuencial y unidimensional, por lo que no son capaces de representar información espacial o de coordenadas tridimensionales. Existen además representaciones que codifican la topología 3D de la molécula, tales como los archivos **SDF** (Structure-Data File) [331].

La representación lineal más ampliamente utilizada es la notación **SMILES** (*Simplified Molecular Input Line Entry System*) [418]. Los códigos o fórmulas SMILES son un sistema de notación lineal que representan las estructuras moleculares de los compuestos químicos en forma de cadenas de caracteres alfanuméricos de forma concisa y legible. En un código SMILES, cada átomo y enlace en una molécula se representa mediante símbolos y caracteres específicos. Los átomos son representados por su símbolo químico (por ejemplo, el caracter **C** para el carbono, o el caracter **N** para el nitrógeno), y los enlaces entre átomos en la estructura son representados por medio de símbolos (por ejemplo, el caracter **-** para enlaces simples, **=** para enlaces dobles, o **#** para enlaces triples). Los átomos de hidrógeno no se escriben explícitamente, sino que se representan siguiendo convenciones particulares a la notación. Por otra parte, se emplean corchetes **[ ]**, paréntesis **( )** y números naturales para denotar grupos atómicos, anillos aromáticos e isótopos, y los caracteres **@**, **\** y **/** para denotar isomería y orientación en enlaces dobles.

El cómputo de una fórmula SMILES se realiza partiendo del grafo asociado a la estructura molecular del compuesto, donde los átomos son entendidos como nodos y los enlaces como arcos no dirigidos. Partiendo de un nodo, se realiza un recorrido en profundidad del grafo, listando las componentes de la estructura de acuerdo a la notación fijada por el estándar. Considerando que no existe un único recorrido posible ni una única forma de comenzar el listado de nodos en un grafo, pueden obtenerse múltiples representaciones SMILES a partir de una misma molécula. Sin embargo, las fórmulas SMILES son invertibles, siendo siempre posible reconstruir la estructura molecular asociada. Existen múltiples librerías de informática molecular que permiten el cómputo de códigos SMILES canónicos (siguiendo siempre un mismo algoritmo de recorrido, que da como resultado representaciones unívocas del compuesto), entre las que destaca *RDKit* [211] como la

más popularmente empleada en el área. La figura 3.2 demuestra de forma sintética el algoritmo de construcción de un código SMILES a partir de un grafo molecular.

Derivados de la codificación SMILES, los patrones **SMARTS** (*SMILES Arbitrary Target Specification*) [85] son una notación que permite identificar patrones estructurales en compuestos químicos mediante la disposición de átomos y enlaces en una estructura molecular. Mayormente, siguen el conjunto de reglas establecido por el estándar SMILES; por lo que las representaciones resultantes consisten en cadenas de caracteres alfanuméricos de longitud variable. Los patrones SMARTS son empleados para identificar subestructuras específicas, por lo que resultan adecuados para el análisis de similitud estructural.

Otra representación lineal ampliamente utilizada es el estándar **InChI** (*IUPAC International Chemical Identifier*) [152]. Esta notación introducida por la *Unión Internacional de Química Pura y Aplicada* (IUPAC, por sus siglas en inglés) [46], consiste en una fórmula identificadora lineal que, al igual que SMILES, codifica la estructura molecular en una secuencia de caracteres alfanuméricos. A diferencia del estándar SMILES, bajo el cual es posible representar un mismo compuesto por medio de múltiples fórmulas, el estándar InChI permite representaciones unívocas y también invertibles, lo que facilita la búsqueda de compuestos en bases de datos. Esta codificación, sin embargo, resulta menos interpretable a nivel humano que los códigos SMILES, razón por la cual su uso no es tan extendido.

Un código InChI siempre comienza con la subcadena **InChI=** seguida del número de versión del algoritmo empleado para su cómputo, y luego continúa con la codificación de la estructura, la cual comprende múltiples capas representadas en subcadenas. La primera capa representa información sobre la topología y conectividad atómica del compuesto, describiendo cómo se enlazan los átomos en la molécula. En las capas adicionales se representa la información de isomería, estereoquímica, cargas y otros atributos. A grandes rasgos, la generación de un código InChI consiste en tres pasos: normalización de la estructura, que remueve información redundante; canonicalización, para generar una secuencia única; y serialización, para obtener una cadena de caracteres. Se toma la estructura representada de forma estandarizada y se aplican reglas y transformaciones que generan cada una de las capas de información. Finalmente, las capas computadas son ensambladas en la cadena final.

Por último, dentro de las representaciones lineales destacamos la codificación **SELFIES** (*SELF-referencing Embedded Strings*), introducida en 2019 por Krenn et al. [207]. Esta técnica de codificación permite representar las estructuras moleculares de manera unívoca e invertible por medio de cadenas de caracteres alfanuméricas. Una diferencia notable entre los códigos SELFIES y los SMILES radica en la representación de ramificaciones en la estructura: dado su algoritmo

de construcción, los códigos SMILES pueden exhibir dependencias de largo rango entre caracteres en la cadena, con ramificaciones que comienzan en porciones tempranas de la cadena y terminan en las últimas posiciones de la misma. Esto hace que ciertos modelos de aprendizaje automático tengan dificultades para identificar ciertas subestructuras moleculares. Las codificaciones SELFIES resuelven las dependencias de largo rango en la cadena utilizando una estrategia de anidamiento: cada cadena SELFIES representa una subestructura química, y estas cadenas son anidadas para formar estructuras más complejas y reflejar las relaciones topológicas en la estructura. Esta estrategia de anidamiento permite codificar dependencias de largo rango entre partes de la cadena de una manera jerárquica y eficiente. La figura 3.2 muestra un ejemplo de este tipo de representación molecular, en contraste con la construcción de un código SMILES.

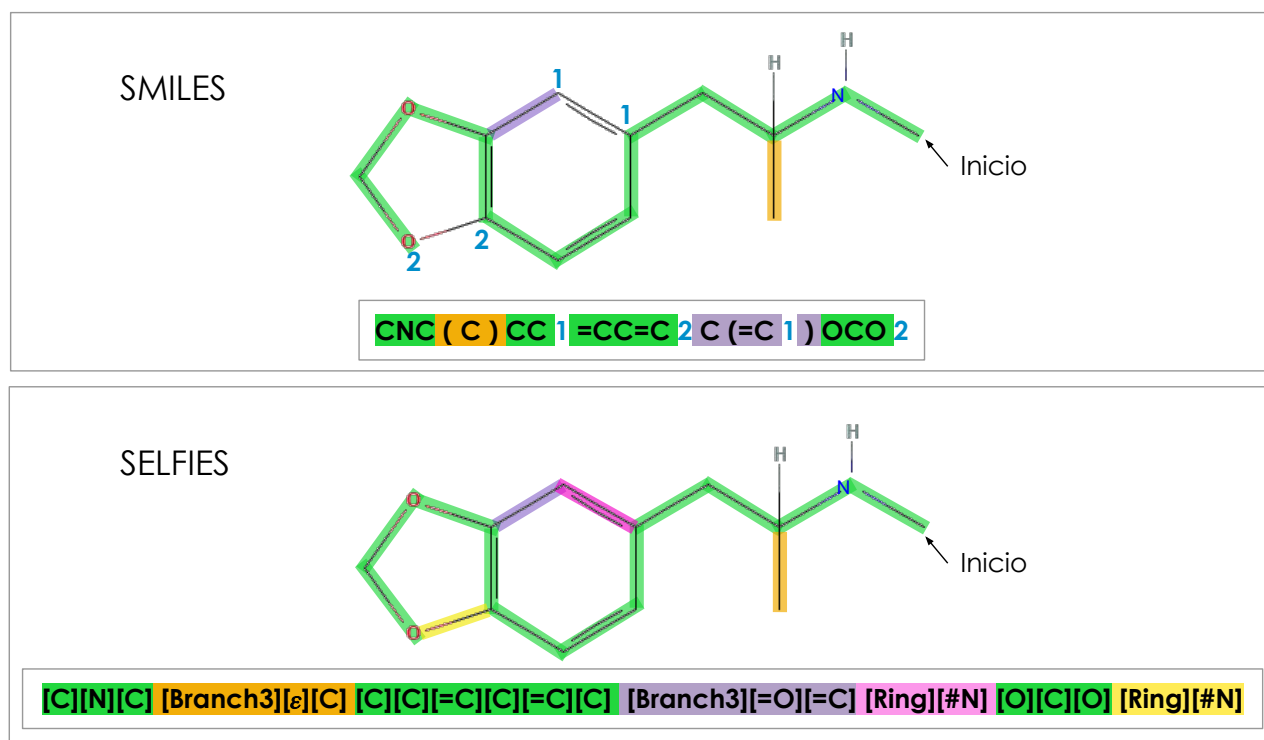


Figura 3.2: Representación esquemática del algoritmo de cómputo de las fórmulas SMILES y SELFIES de un compuesto químico. Como se puede apreciar, las fórmulas SELFIES resuelven las dependencias de largo rango en la secuencia por medio de anidamiento de subestructuras, introduciendo *tokens* especiales para denotar la existencia de ramificaciones y anillos aromáticos en la estructura. Las fórmulas SMILES denotan estos mismos grupos químicos por medio de números, donde dos números iguales son empleados para notar la apertura y cierre de un grupo o fragmento.

### 3.3.1.2. Descriptores moleculares

Los descriptores moleculares son valores numéricos que representan diversas características y propiedades de un compuesto químico [378, 252]. Son computados a partir de su estructura molecular, generalmente expresada en términos de su código SMILES o InChI. Considerados una representación molecular tradicional, los descriptores moleculares han constituido una de las representaciones estándar en el ámbito de la informática molecular durante años, especialmente como representación molecular para el modelado predictivo de propiedades o actividad biológica [252].

A grandes rasgos, los descriptores moleculares pueden categorizarse según su dimensionalidad o según el tipo de información sobre el compuesto químico que codifican. En términos de su dimensionalidad, podemos definir las siguientes categorías:

- Descriptores 0D: estos descriptores son independientes de la estructura bidimensional o tridimensional de la molécula y son calculados a partir de su fórmula química, en función de la presencia o ausencia de ciertos átomos en la misma. Ejemplos de este tipo de descriptores incluyen el número de átomos de carbono, hidrógeno, oxígeno, etc., así como el peso molecular o la cantidad de aceptadores o donantes de puentes de hidrógeno.
- Descriptores 1D: estos descriptores representan información unidimensional de la estructura molecular y se computan a partir del procesamiento de la estructura de la cadena o la secuencia lineal que codifica al compuesto. Entre los ejemplos más notables de este tipo de descriptores se encuentran el número de enlaces simples, dobles o triples o la longitud de las cadenas de carbono.
- Descriptores 2D: estos descriptores representan información bidimensional, codificando información de la estructura planar del compuesto. Entre estos descriptores se encuentran aquellos vinculados a conectividad, descriptores de grupos funcionales, fragmentos, solubilidad e índices topológicos.
- Descriptores 3D: estos descriptores codifican la información tridimensional del compuesto químico y se vinculan a su estructura y conformación en el espacio. Generalmente, solo aportan información significativa si son computados a partir de representaciones tridimensionales de la molécula. En esta categoría encontramos descriptores tales como la energía conformacional, la polarizabilidad, los momentos dipolares y la superficie molecular.

Por otra parte, es posible categorizar a los descriptores moleculares según el tipo de información que representan del compuesto químico. En líneas generales podemos definir las siguientes categorías:

- Estructurales: representan características moleculares vinculadas a la conectividad atómica, la estructura y la topología de la molécula, tales como la cantidad de anillos aromáticos, fragmentos y longitud de enlaces.
- Físico-químicos: se relacionan con propiedades físicas y químicas del compuesto químico, como la solubilidad, la polaridad, la acidez y la basicidad.
- De carga: codifican información sobre la distribución de cargas en la molécula, incluyendo momentos dipolares, carga parcial de átomos y coeficientes de polarización.
- De forma: representan información sobre la forma y el volumen de la molécula y son especialmente adecuados en aquellas tareas vinculadas al análisis de interacciones moleculares, como los descriptores de campo molecular.
- Cuánticos: emplean cálculos cuánticos para representar propiedades electrónicas y estructurales de la molécula, como la energía electrónica, la densidad de electrones y los orbitales moleculares.

Para obtener una representación de un compuesto químico basada en descriptores, en primera instancia, se deben seleccionar aquellos descriptores moleculares específicos necesarios en función de la tarea a resolver o de los objetivos de investigación. Luego, se emplean soluciones de software o bibliotecas especializadas, coloquialmente denominadas *calculadoras de descriptores moleculares*, que permiten computar los valores de los descriptores seleccionados a partir de las representaciones lineales de los compuestos. Estos programas aplican algoritmos y ecuaciones matemáticas específicas para cada descriptor. Finalmente, a partir de los descriptores computados, se elabora un vector de valores continuos, donde cada descriptor ocupa una posición fija y pasa a constituir una de las características o atributos en términos de los cuales se describe al compuesto. Estos vectores, ilustrados en la figura 3.3, pueden ser utilizados como datos de entrada en la mayoría de los algoritmos de aprendizaje automático y profundo.

La elección del conjunto de descriptores moleculares a emplear en una representación constituye un paso crucial para el éxito de la tarea a desarrollar. Esta selección es fuertemente dependiente de la propiedad específica que se desea estudiar, y generalmente se emplean combinaciones de distintos tipos de descriptores a fin de capturar múltiples aspectos de la estructura molecular [379]. No obstante, esta tarea no resulta trivial: la selección manual de descriptores requiere de vasto conocimiento experto, considerando que la semántica química de los mismos puede resultar oscura y poco interpretable [379]. Por otro lado, la selección automática de características no está exenta de sesgos introducidos por los propios algoritmos empleados para la tarea. Además, los algoritmos

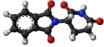
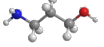
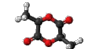
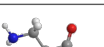
Compuesto	MW	LogP	HBD	nRings	ABCIndex	...	ZagrebIndex
	132	10,63	4	3	24,17	...	8,75
	95	5,31	1	0	15,43	...	5,66
	85	15,78	2	1	22,27	...	8,25
	110	4,73	1	0	19,74	...	10,15

Figura 3.3: Representación ilustrativa de una base de datos de compuestos químicos representados por medio de descriptores moleculares.

de selección automática de características pueden no capturar las complejas interacciones entre las mismas, lo cual, sumado al proceso inherentemente automático de selección, puede atentar contra la interpretabilidad de la representación molecular obtenida y su posterior adopción por parte de los expertos. Por otra parte, los algoritmos empleados para computarlos realizan suposiciones y simplificaciones sobre la estructura molecular y existen miles de descriptores computables, por lo que se trata de un problema combinatorial complejo con el potencial de pérdida de información valiosa [222]. Otra desventaja del uso de este tipo de representación es la complejidad en la tarea de imputar valores faltantes para moléculas de estructuras complejas, y la necesidad de normalizar los valores para lograr desempeños adecuados de los algoritmos de aprendizaje automático. Por último, las representaciones obtenidas a partir de descriptores moleculares no son unívocas ni invertibles, resultando inviable reconstruir la estructura molecular a partir del vector de descriptores obtenido.

### 3.3.1.3. *Fingerprints* moleculares

Los *fingerprints moleculares* son representaciones moleculares que condensan información estructural y físico-química del compuesto por medio de una secuencia de valores numéricos, por lo general mediante bits (0 y 1) [62]. La adopción de *fingerprints* como representación molecular para diversas tareas en informática molecular es amplia, siendo considerados una representación tradicional y estándar en múltiples dominios. Esto se fundamenta en que permiten realizar una comparación y análisis eficiente de los compuestos y en que son ricos en información a pesar de su notación simple y compacta.

Existen diferentes tipos de *fingerprint* y cada uno de ellos codifica diferentes aspectos del compuesto químico [62]. La elección del *fingerprint* a emplear se realiza en función de la tarea específica, y a menudo se emplean múltiples representaciones *fingerprint* o en combinación con



otras representaciones moleculares de forma complementaria. Entre las categorías más prominentes encontramos:

- Basados en subestructuras: en este tipo de *fingerprints*, los valores de los bits en la representación se establecen en función de la presencia o ausencia de ciertas subestructuras clave o características específicas, las cuales están listadas por medio de claves estructurales de forma estandarizada. El número de bits en la representación es determinado por la cantidad de claves estructurales, y cada bit se relaciona con la presencia o ausencia de una subestructura en el compuesto. La figura 3.4 ilustra el proceso de obtención de un *fingerprint* basado en subestructuras. Este tipo de *fingerprints* resulta idóneo para configurar alertas estructurales en compuestos químicos, que permiten detectar subestructuras con potencial carcinogénico, mutagénico o tóxico. Una desventaja de este tipo de *fingerprints* radica en que las representaciones de compuestos que no presentan muchas de las claves estructurales listadas son ralas y no muy informativas. En esta categoría, uno de los ejemplos más notables es el de las claves MACCS (*Molecular ACCess System*) [99], que presenta dos variantes: una de 960 y otra de 166 claves estructurales basadas en patrones SMARTS. La versión de 166 bits es la más frecuentemente empleada por ser compacta pero rica en información estructural, abarcando un gran número de características químicas relevantes al descubrimiento de fármacos.

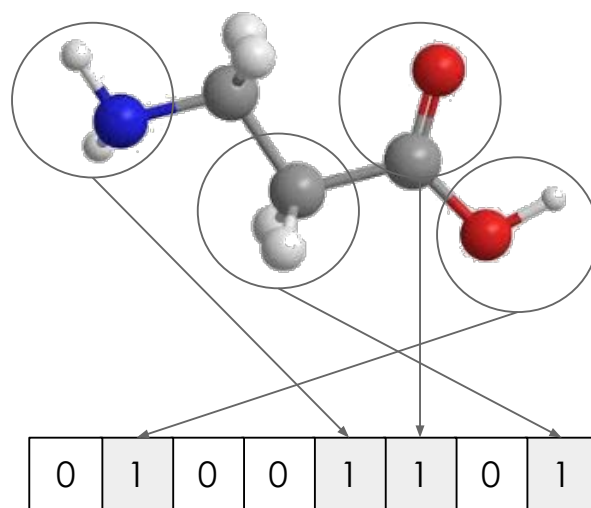


Figura 3.4: Representación esquemática del cómputo de un *fingerprint* basado en subestructuras, donde la presencia o ausencia de ciertas subestructuras o grupos químicos se codifica en bits unívocamente definidos para cada una de ellas.

- Topológicos: los fingerprints topológicos, también denominados como *basados en caminos*, analizan todos los fragmentos de la estructura molecular elaborando una ruta lineal hasta un número establecido de enlaces, para luego computar una función de cifrado *hash* de cada uno de dichos caminos que determina cada uno de los bits o valores de la representación. La ventaja de este tipo de *fingerprint* es que cualquier molécula genera una representación significativa. Sin embargo, su longitud puede ser variable y, por tratarse de *fingerprints* obtenidos por medio de una función *hash*, no es posible rastrear a qué característica específica corresponde cada bit, existiendo además la posibilidad de colisiones, donde múltiples características moleculares se mapeen a un mismo bit. La figura 3.5 ilustra el modo en el que se computan los *fingerprints* topológicos. Un ejemplo saliente de esta categoría es el *fingerprint Daylight* [86], que puede consistir de hasta 2.048 bits y codifica todas las posibles rutas de conectividad entre los átomos de una molécula hasta una longitud determinada.

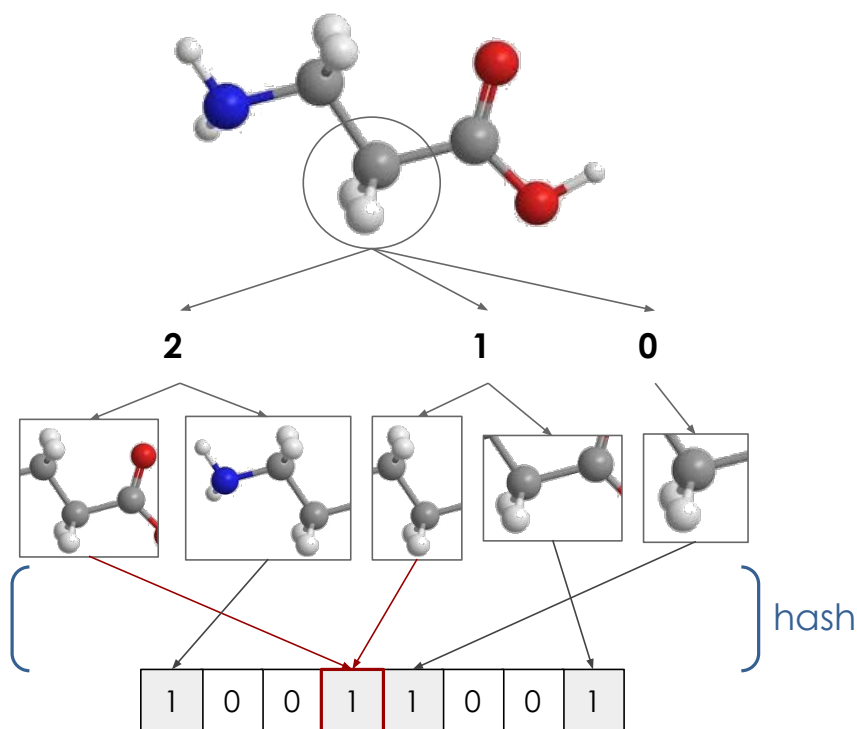


Figura 3.5: Representación esquemática de la estrategia de cómputo de un *fingerprint* topológico, que detecta patrones de conectividad en la estructura molecular y les aplica una función de cifrado *hash* para codificar su presencia en la secuencia. Notar que es posible que se susciten colisiones en la secuencia resultante, lo que supone una pérdida de información.

- De conectividad: son representaciones moleculares que analizan la conectividad atómica y los

patrones de enlaces presentes en la estructura molecular. Al igual que en los *fingerprints* topológicos, para el cómputo de los bits en la secuencia obtenida generalmente se emplean funciones *hash*, lo que deriva en posibles colisiones y en que las representaciones obtenidas no sean invertibles. Los *fingerprints* circulares, una subcategoría de este tipo de representación, ilustrada en la figura 3.6, se computan situándose sobre cada uno de los átomos de la molécula y analizando el entorno de dicho átomo en la estructura, registrando patrones de conectividad en un radio determinado. El ejemplo más paradigmático de esta categoría, devenido en un estándar en informática molecular, son los *Fingerprints de Conectividad Extendida* (*Extended Connectivity FingerPrints* o ECFP) [311], basados en el algoritmo de Morgan [267]. Los *fingerprints* ECFP fueron diseñados específicamente para su uso en modelado predictivo de relaciones estructura-actividad [62]. En ellos, cada bit codifica vecindarios circulares de un radio específico para cada átomo. Generalmente, se emplean valores de radio 1 (ECFP2), 2 (ECFP4) o 3 (ECFP6), y longitudes de 1.024 o 2.048 bits. Una desventaja de estas representaciones es que no resultan adecuadas para consultas de subestructuras, ya que un mismo fragmento puede ser analizado desde diferentes radios, pero sí son adecuadas para comparar de forma eficiente estructuras moleculares completas.

- De fragmentos: los *fingerprints* de fragmentos representan al compuesto químico como una colección de fragmentos atómicos o grupos químicos. Estos *fingerprints* identifican estructuras similares basándose en la presencia de fragmentos comunes, por lo que brindan una representación detallada de la estructura molecular y resultan idóneos para el análisis de similitud estructural.
- Farmacofóricos: los *fingerprints* farmacofóricos se distinguen de las categorías restantes de *fingerprints*, ya que son computados a partir de la identificación de características relevantes para la interacción de un compuesto con un receptor biológico o blanco terapéutico y no solamente en el análisis de la estructura molecular. Capturan información que vincula la disposición espacial y las propiedades químicas de enlaces, grupos funcionales y sitios de unión asociados con la actividad biológica.

#### 3.3.1.4. *Embeddings* moleculares

Saliendo de las representaciones moleculares tradicionales, sobre las cuales hemos expuesto a lo largo de la presente sección, presentamos un tipo de representación molecular que experimenta su auge en la comunidad científica en la actualidad. Se trata de los *embeddings* moleculares,

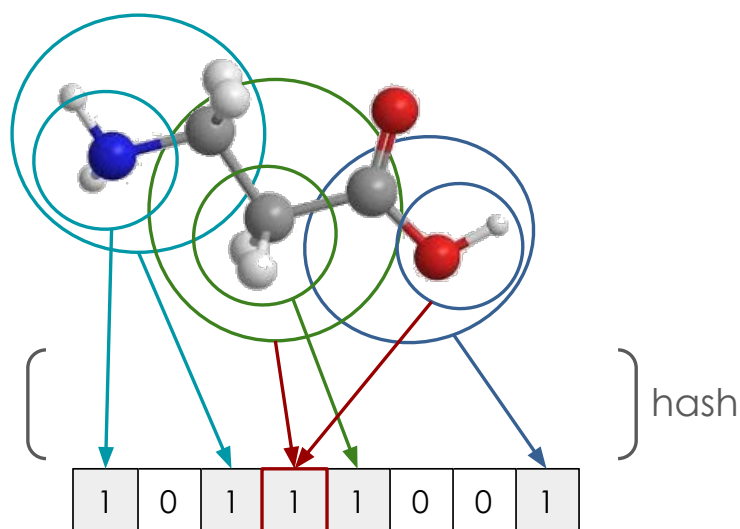


Figura 3.6: Representación esquemática de la estrategia de cómputo de un *fingerprint* circular de radio 1, similar a la adoptada por los *fingerprints de conectividad extendida* (ECFP). En esta estrategia, se estudian las subestructuras obtenidas a partir de analizar cada átomo en el grafo molecular y su vecindario en un radio establecido y se les aplica una función de cifrado *hash* para codificar su presencia en la secuencia. En este tipo de representaciones pueden ocurrir colisiones, con la consecuente pérdida de información.

representaciones de compuestos químicos que consisten en vectores numéricos obtenidos a partir de algoritmos de aprendizaje automático y profundo [103, 73, 421, 26]. Los *embeddings* moleculares capturan información esencial sobre la estructura y propiedades de un compuesto químico a partir de un proceso de entrenamiento automático, que detecta patrones intrínsecos en conjuntos masivos de datos moleculares. Los *embeddings* moleculares han ganado tracción en los últimos años debido al significativo avance tecnológico en aprendizaje profundo aplicado al procesamiento de lenguaje natural, disciplina de la que deriva el concepto de *embedding* [258]. Estas representaciones son ampliamente utilizadas en una variedad de aplicaciones de informática molecular, desempeñando un papel crucial en la identificación de compuestos químicos con propiedades deseables.

Los *embeddings* moleculares tienen longitudes variables, determinadas principalmente por las características de la arquitectura del modelo de aprendizaje empleado para su cómputo. Cada dimensión de dicho vector representa un atributo que describe a la molécula, pudiendo capturar información sobre aspectos de topología, conectividad, presencia de grupos funcionales, bioactividad y otras propiedades químicas. Si bien son altamente versátiles y flexibles, su interpretabilidad es

reducida, por cuanto no es posible *a priori* conocer qué aspectos de la molécula son codificados por cada una de las posiciones del vector. Además, la selección del conjunto de compuestos químicos empleados para el entrenamiento del modelo es determinante en la calidad de las representaciones obtenidas, siendo necesario utilizar conjuntos masivos y estructuralmente diversos para obtener *embeddings* ricos en información [103, 73].

Existen numerosos algoritmos y modelos basados en aprendizaje profundo para el cómputo de *embeddings* moleculares en la literatura [73]. La mayoría de ellos son obtenidos tomando como representación de entrada las fórmulas SMILES de los compuestos químicos empleados en el entrenamiento, si bien es posible basarse en otras representaciones moleculares para su obtención. Entre los ejemplos más notables de *embeddings* basados en aprendizaje profundo podemos destacar aquellos obtenidos a partir de arquitecturas de redes neuronales recurrentes (RNNs), *autoencoders* y mecanismos de *Self-Attention*. Un aspecto clave es la versatilidad de las arquitecturas empleadas, las cuales pueden ser combinadas o yuxtapuestas para obtener los múltiples beneficios de cada una de ellas.

Los *embeddings* basados en RNNs procesan la secuencia SMILES separada en *tokens* o unidades atómicas, las cuales pueden ser caracteres o *n*-gramas formados por subcadenas consecutivas de longitud *n*. Estas representaciones buscan identificar patrones en la estructura molecular, capturando información de dependencias de corto y largo rango entre *tokens* en la secuencia SMILES, por lo que la adecuada elección de la celda recurrente empleada en la construcción del modelo resulta un factor determinante en la calidad de las representaciones obtenidas. La dimensionalidad de estos *embeddings* queda determinada a partir de la cantidad de celdas recurrentes de la RNN y sus estados internos. Ejemplos de este tipo de *embedding* molecular fueron propuestos por Xu et al. [436] y Grisoni et al. [135].

Los *embeddings* basados en *autoencoders* son quizá los más intuitivos, derivados directamente de la propuesta original de Mikolov et al. [258] para la construcción de *embeddings* de palabras en el contexto de procesamiento de lenguaje natural. En estos modelos, la secuencia SMILES es también separada en unidades atómicas, cada una de las cuales es mapeada a un valor unívoco y constituye un atributo del vector de entrada luego suministrado al modelo. La dimensionalidad de los *embeddings* obtenidos está determinada por la del espacio latente aprendido por el autoencoder, la cual es fijada en el diseño de la arquitectura. Entre los ejemplos más notables de este tipo de representación encontramos los *embeddings* propuestos por Jaeger et al. [176] y por Öztürk et al. [290].

Una propuesta relativamente reciente para la construcción de *embeddings* moleculares es la de emplear arquitecturas basadas en mecanismos de atención (sección 2.5.3.4 del capítulo 2). Las arquitecturas de *Self-Attention* permiten analizar todas las partes de la secuencia SMILES en paralelo y establecer relaciones entre ellas a través del cómputo de pesos de atención, lo que representa un avance en términos de eficiencia computacional y expresividad con respecto a las arquitecturas basadas en RNNs. Se destacan en esta categoría las propuestas de Zheng et al. [449] y Shin et al. [349].

Los *embeddings* moleculares, además, pueden ser computados siguiendo una estrategia de aprendizaje supervisado, cuando se incorpora información del perfil de bioactividad de los compuestos empleados en el entrenamiento del modelo, o no supervisado, cuando el aprendizaje se basa únicamente en el análisis automático de la fórmula SMILES de cada compuesto. Los *embeddings* moleculares no supervisados admiten ser entrenados con conjuntos de datos químicos masivos y diversos, por lo que dan lugar a modelos con alta expresividad y capacidad de detección de patrones químicos. Los *embeddings* moleculares supervisados tienen la ventaja de ser expresivos no solo en términos de los aspectos estructurales del compuesto, sino que también en términos de sus propiedades farmacológicas. Sin embargo, su cómputo requiere de bases de datos químicos etiquetadas, las cuales suelen ser restringidas en cantidad de compuestos, lo cual resulta detrimental para la capacidad de aprendizaje y abstracción del modelo de *embeddings*. Además, carecen de flexibilidad, por cuanto es necesario entrenar un nuevo modelo de *embeddings* por cada propiedad a modelar, considerando que un mismo compuesto puede estar etiquetado con diferentes valores para distintos blancos terapéuticos o propiedades farmacocinéticas.

La elección del tipo de *embedding* molecular depende de varios factores, entre ellos los objetivos de investigación, la tarea específica para la cual dicha representación es requerida, la propiedad físico-química en estudio y los recursos computacionales de los que se dispone para el entrenamiento de los modelos. Un detalle no menor que se debe considerar sobre este tipo de representaciones es su invertibilidad, es decir, la capacidad de reconstrucción del compuesto químico a partir de dicha representación. La invertibilidad de los embeddings está absolutamente supeditada a la arquitectura y al buen desempeño del modelo empleado: por ejemplo, los modelos basados en *autoencoders* son capaces de reconstruir el compuesto original a partir de la representación latente; sin embargo, pueden arrojar reconstrucciones erróneas si no están entrenados correctamente.

### 3.3.2. Modelado QSAR

El modelado QSAR es una disciplina del campo de la informática molecular que se basa en el diseño de estrategias que permitan establecer relaciones cuantitativas entre la estructura molecular de los compuestos químicos y su bioactividad o sus propiedades farmacocinéticas y químicas. El modelado QSAR es una herramienta indispensable en el extenso proceso de descubrimiento de fármacos, con un largo recorrido histórico y establecido como una prominente área de investigación en sí misma. Tuvo sus inicios en la década de 1.960 de la mano del trabajo pionero en el área desarrollado por Hansch and Fujita [145], quienes postularon la idea de que la estructura molecular de un compuesto está relacionada con su actividad biológica. En dicho trabajo, establecieron relaciones matemáticas que permitían predecir el perfil de bioactividad de compuestos novedosos en función de ciertos descriptores moleculares. Desde aquel trabajo hasta la actualidad, y gracias al avance tecnológico en química medicinal y aprendizaje automático y profundo, el modelado QSAR ha evolucionado drásticamente hacia la obtención de modelos precisos de relaciones complejas y no lineales [316, 345].

El modelado QSAR no solo se emplea para predecir la actividad biológica de compuestos candidatos en el contexto del desarrollo de fármacos, sino que además se aplica en la predicción de toxicidad, ecotoxicidad y biodegradabilidad, a fin de evaluar el impacto de las sustancias en los organismos vivos y su medio ambiente [154]. Los modelos QSAR, además, se emplean para la optimización de propiedades físico-químicas de compuestos, como paso preliminar a la fase preclínica del desarrollo de fármacos. La importancia del modelado QSAR radica en que permite racionalizar y acelerar el proceso de diseño de fármacos, reduciendo significativamente los costos. Además, permite la evaluación eficiente de grandes quimiotecas, aumentando así la reserva de compuestos candidatos para etapas finales del proceso en las que la tasa de deserción es muy alta, tal y como ilustramos en la figura 3.1 [362].

El desarrollo de un modelo QSAR sigue, a grandes rasgos, los mismos pasos que el de cualquier modelo predictivo de aprendizaje automático, tal y como fue descrito en el capítulo 2, y se ilustra en la figura 3.7. En primera instancia se obtienen y preparan los datos de origen químico, los cuales pueden estar en cualquiera de los formatos de representación vistos anteriormente. A dichos datos se les realiza una etapa de preprocesamiento, que generalmente involucra pasos como la remoción de duplicados, la canonicalización de fórmulas SMILES, la identificación y exclusión de mezclas, metales y polímeros, la eliminación de sales y solventes de las fórmulas químicas, y la toma de decisiones con respecto a las formas isoméricas potencialmente presentes en el conjunto de datos [386]. En función de la representación empleada como entrada y de los algoritmos de aprendizaje que vayan a

utilizarse para el modelado, se puede llevar a cabo un paso de selección de características, junto con la correspondiente imputación de valores faltantes y la normalización de los atributos numéricos. Luego se procede al particionado de los datos, el cual se realiza considerando si la tarea predictiva es de clasificación o regresión. Si se trata de una tarea de clasificación, las particiones se computan de forma estratificada, manteniendo las proporciones de clase presentes a través de todas las particiones. La mayoría de las tareas de clasificación en modelado QSAR son binarias, considerando los compuestos etiquetados como *activos* o *inactivos*, según exhiben o no el perfil de bioactividad que se busca predecir, respectivamente.

Una vez preparados los datos, se procede a la selección y entrenamiento de los modelos. En la literatura es posible encontrar estrategias de modelado QSAR basadas en una gran variedad de técnicas de aprendizaje automático [68, 316, 130]; en particular, los modelos basados en aprendizaje profundo han ganado tracción durante los últimos años, constituyendo hoy una de las técnicas estándar para modelado QSAR [447, 118, 322, 119, 248]. Esto se debe en gran parte a que las redes neuronales profundas (DNNs) obtienen rendimientos similares o superiores a los algoritmos tradicionales y a la vez son capaces de lidiar con grandes conjuntos de datos y con representaciones moleculares de alta dimensionalidad sin la necesidad de un paso previo de selección de características [322]. Para la validación de los modelos desarrollados, se seleccionan métricas de rendimiento adecuadas a la tarea de modelado específica y a las características propias del conjunto de datos en estudio: si se trata de un problema de clasificación con alto desbalance de clases, como suele ocurrir con las bases de datos químicos [202], se priorizan métricas como la *precisión*, la *exactitud balanceada* (*BAcc*) o el *puntaje F1* por sobre la *exactitud* (*Acc*), por ejemplo (ver sección 2.7). Finalmente, se analiza la significancia estadística de los datos por medio de técnicas de análisis de la varianza.

### **Dominio de aplicabilidad**

Una pieza importante del proceso de modelado QSAR es la definición de su *dominio de aplicabilidad*. El dominio de aplicabilidad de un modelo QSAR se vincula con la determinación de las condiciones bajo las cuales un modelo QSAR es válido, y puede ser definido como el subespacio químico en el que resulta esperable que el modelo realice predicciones precisas [200, 412]. El espacio molecular de compuestos candidatos es potencialmente infinito y de enorme diversidad estructural y físico-química, por lo que delimitar ciertos criterios mínimos de similitud o afinidad entre compuestos antes de establecer relaciones entre su estructura química y su actividad biológica es indispensable para garantizar la utilidad de un modelo QSAR [412].



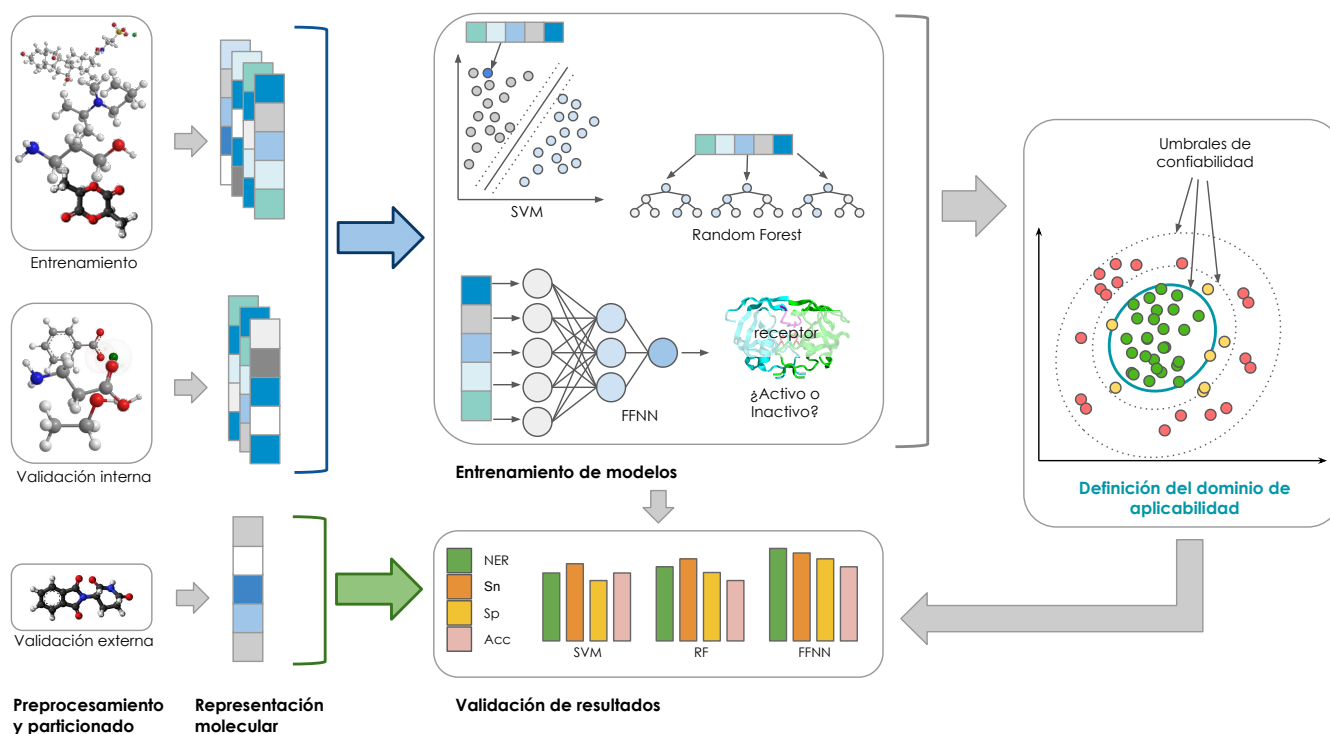


Figura 3.7: Esquemmatización del proceso de desarrollo de un modelo QSAR. Inicialmente, se computan las representaciones moleculares de los compuestos químicos y se realiza el particionado del conjunto de datos en particiones de entrenamiento, validación interna y validación externa, siguiendo alguna de las estrategias de particionado explicadas en el capítulo 2. Luego se realiza el entrenamiento del modelo, donde la predicción es un valor asociado a la bioactividad del compuesto con respecto a un receptor biológico particular, en el caso de un modelo de regresión, o un valor categórico de bioactividad (*activo o inactivo*, por ejemplo), en el caso de un modelo de clasificación. Empleando alguna estrategia se determina el dominio de aplicabilidad del modelo y, en función de las predicciones del modelo final y de su dominio de aplicabilidad, se analiza el desempeño del mismo.

La definición del *dominio de aplicabilidad* de un modelo QSAR involucra la determinación del rango de condiciones que deben cumplir los compuestos a ser evaluados por el modelo a fin de poder garantizar precisión en los resultados obtenidos. Estas condiciones pueden ser de distinta índole, incluyendo restricciones en la estructura molecular, el rango de valores de ciertos descriptores moleculares, la presencia o ausencia de ciertos grupos funcionales, etc. En principio, cada modelo QSAR tiene asociado un dominio de aplicabilidad, cuyos límites pueden ser variables y se definen en función del asesoramiento experto, del desarrollo del propio modelo o de la tarea predictiva.

Existen diferentes estrategias para la definición del dominio de aplicabilidad de un modelo QSAR,

la mayoría de las cuales se basa en métricas estadísticas y análisis de los datos químicos empleados para entrenar el modelo [200, 306, 412]. En principio, dichas estrategias determinan los límites del dominio de aplicabilidad de forma independiente al proceso de entrenamiento del modelo, y se basan en principios fundamentales como el principio de similitud estructural (sección 3.2) y en métricas de distancia (capítulo 2, sección 2.7) para detectar subestructuras moleculares o valores de características moleculares atípicos. A partir de tal análisis, se elaboran criterios para incluir o excluir compuestos nunca vistos por el modelo de su dominio de aplicabilidad [317, 228, 42]. Este tipo de estrategias se denominan *de detección de novedad (novelty detection)* [200] y, si bien han sido tradicionalmente empleadas en el área de modelado QSAR, presentan la limitación de ser fuertemente dependientes del conocimiento experto y de la correcta interpretación semántica de la representación molecular empleada, por lo que son propensas a fallas y restan flexibilidad al proceso de representación molecular.

Más allá de las estrategias de definición de dominio de aplicabilidad basadas en detección de novedad, existe otra categoría de estrategias basadas en el análisis del modelo entrenado, asociadas a la confianza en la predicción. Estas estrategias dependen del proceso de entrenamiento y se basan en la interpretación probabilística de los resultados del modelo. Estas estrategias, denominadas *de estimación de confianza (confidence estimation)* [200] son flexibles y versátiles, pudiendo adaptarse a diferentes tareas predictivas, representaciones moleculares de entrada, y no son tan fuertemente dependientes del conocimiento experto. Las estrategias basadas en estimación de confianza definen umbrales mínimos de confiabilidad sobre las probabilidades de salida del modelo, y determinan la pertenencia de un compuesto nunca visto al dominio de aplicabilidad en función de la probabilidad de salida arrojada por el modelo para dicho compuesto. Desde ya, existen también estrategias de definición del dominio de aplicabilidad híbridas, que integran componentes de *novelty detection* y *confidence estimation*.

### Desafíos del modelado QSAR

Además de los múltiples desafíos inherentes al desarrollo de modelos de aprendizaje automático y profundo, discutidos en el capítulo anterior, el modelado QSAR enfrenta desafíos particulares del dominio de la informática molecular. En primer lugar, es necesario contar con bases de datos químicas grandes y diversas, donde el perfil de bioactividad a predecir se vea caracterizado por una multiplicidad de patrones estructurales. La insuficiencia de datos puede derivar en modelos imprecisos o con dificultades para generalizar ante compuestos nunca vistos. Si bien existen enormes quimiotecas que constan de millones de compuestos no etiquetados [360, 196, 116], las bases de

datos etiquetadas suelen ser escasas en cantidad de compuestos y a menudo contar con información redundante, contradictoria o heterogénea. Por otra parte, el perfil de bioactividad de un compuesto se compone de múltiples interacciones farmacocinéticas y farmacodinámicas complejas, por lo que a menudo es necesario desarrollar múltiples modelos o modelos multi-tarea, capaces de correlacionar las diversas aristas del perfil farmacológico de los compuestos [352, 248].

La definición del dominio de aplicabilidad de un modelo QSAR tiene a su vez desafíos particulares, vinculados a la interpretación semántica de las características moleculares y de la incidencia de ciertos fragmentos o subestructuras en el perfil de bioactividad de un compuesto. Además, el dominio de aplicabilidad depende de los datos empleados en el entrenamiento del modelo y de la estrategia de modelado, por lo que puede restringirse significativamente de no ser lo suficientemente diverso o representativo el conjunto de datos empleado en el entrenamiento de modelo [412].

Uno de los aspectos más críticos del modelado QSAR es la interpretabilidad de los modelos obtenidos, considerando que se trata de modelos ideados para brindar soporte a expertos en química medicinal y farmacia en la toma de decisiones de cara al desarrollo de nuevos medicamentos [251]. Más allá de la interpretabilidad de los modelos en términos algorítmicos o de su proceso matemático o probabilístico de inferencia y aprendizaje [16, 243], podemos referirnos a los modelos QSAR con respecto a su interpretabilidad físico-química. En este sentido, se ven involucradas no solamente la capacidad de entender cómo las características o atributos en términos de las que se describe al compuesto inciden en la predicción del modelo, sino también la capacidad de análisis de la relación entre las semánticas físico-químicas de dichas características moleculares y el blanco farmacológico. En la historia del modelado QSAR existen modelos desarrollados primordialmente gracias al conocimiento experto, basados en principios fundamentales de la propiedad estudiada. Esta noción de interpretabilidad se sitúa en un plano de abstracción mayor a los comúnmente abordados por las disciplinas de la inteligencia artificial explicable, y resulta de vital importancia en el dominio del modelado QSAR.

### 3.3.3. Cribado virtual de fármacos

Considerando los desafíos inherentes a la exploración de espacios moleculares potencialmente infinitos, el *cribado virtual de fármacos*, conocido por su nombre en inglés (*virtual screening*), es una disciplina fundamental en informática molecular que abarca el conjunto de estrategias computacionales para la búsqueda eficiente de nuevos compuestos candidatos [332, 368, 226, 236, 323, 197]. El cribado virtual involucra el uso y desarrollo de técnicas computacionales para múltiples

tareas, no solo vinculadas al análisis del espacio de compuestos químicos, sino también de los blancos terapéuticos. Esta disciplina involucra múltiples tareas, entre las cuales podemos enumerar:

- Selección de blancos terapéuticos: esta tarea consiste en la identificación de proteínas, enzimas o receptores que desempeñen un rol importante en el proceso biológico en estudio, buscando en particular identificar objetivos susceptibles de modulación por compuestos químicos. Los receptores biológicos ligan con compuestos químicos por medio de dinámicas de acoplamiento en ciertos sectores activos de la proteína, también denominados coloquialmente *bolsillos* (*pockets*). Por lo tanto, la tarea de detectar blancos viables deviene en la identificación de estructuras proteínicas con sitios activos alcanzables, lo que resulta arduo considerando la extrema complejidad de las mismas. Una cuidadosa selección de blancos terapéuticos permite concentrar el posterior diseño de fármacos en áreas del blanco con un alto potencial terapéutico.
- Modelado estructural de blancos terapéuticos y ligandos: vinculada a la selección de blancos terapéuticos, esta tarea radica en la creación de modelos tridimensionales de proteínas y moléculas para permitir un análisis minucioso de sus interacciones a nivel atómico. Este enfoque permite determinar los mecanismos de acoplamiento entre ligandos (compuestos químicos) y los sitios activos de las proteínas (blancos farmacológicos), diseñar conformaciones 3D de los compuestos que permitan acoplamientos estables, y permitir además la optimización de fármacos existentes y compuestos candidatos. En esta tarea se comprenden las estrategias de *docking molecular*, que simulan la interacción de acoplamiento [359].
- Cribado basado en estructura: el objetivo de este conjunto de estrategias es el análisis de la estructura molecular del compuesto químico y su similitud con compuestos conocidos que exhiban propiedades físico-químicas deseables. Este enfoque comprende un amplio espectro de técnicas de análisis de similitud estructural y físico-química, incluyendo el análisis automático de bases de datos con información estructural de compuestos, el análisis de representaciones moleculares en alta dimensionalidad por medio de métricas de distancia o de algoritmos de aprendizaje automático, y el análisis de proyecciones en baja dimensionalidad de conjuntos de compuestos químicos por medio de estrategias de analítica visual.
- Diseño *de novo*: el diseño *de novo* de fármacos engloba un conjunto de estrategias computacionales orientadas a la creación de nuevos compuestos químicos con propiedades físico-químicas específicas. Esta subdisciplina se sustenta en algoritmos y modelos computacionales del aprendizaje profundo, principalmente, y combina elementos del modelado predictivo de propiedades, la simulación de interacciones y el análisis de similitud estructural entre moléculas.

Desarrollaremos en profundidad contenidos asociados al diseño *de novo* en la sección 3.3.5 del presente capítulo.

- **Reposicionamiento de fármacos:** las estrategias de reposicionamiento de fármacos se comprenden dentro del espectro de tareas de cribado virtual de fármacos, si bien no pertenecen al proceso tradicional de desarrollo de nuevos medicamentos en etapas. Estas estrategias buscan evaluar compuestos existentes, previamente validados y aprobados como medicamentos, para nuevas indicaciones terapéuticas. Las estrategias de reposicionamiento se basan en el principio de que un compuesto químico es capaz de interactuar con más de un receptor biológico o blanco terapéutico en los organismos vivos, por lo que, en lugar de crear un nuevo compuesto desde cero para un determinado perfil de bioactividad deseado, se aprovechan los estudios previos sobre compuestos ya aprobados y se los analiza para nuevos tratamientos en los que se hipotetiza que pueden tener efectividad. Las estrategias de reposicionamiento de fármacos son altamente dependientes del análisis de determinantes de similitud estructural y de las herramientas de modelado predictivo de propiedades farmacodinámicas, y resultan cruciales en contextos en los que se necesitan nuevas terapias farmacológicas en poco tiempo. Un ejemplo notable de estrategias de reposicionamiento de fármacos es el caso de las numerosas terapias farmacológicas descubiertas en tiempo récord para el tratamiento preventivo y paliativo del COVID-19, donde se descubrió que medicamentos para la malaria, como la *hidroxicloroquina*, o antiparasitarios, como la *ivermectina*, podían presentar efectos potencialmente beneficiosos ante dicha enfermedad.

Las técnicas computacionales de cribado virtual de fármacos permiten no solamente explorar enormes espacios químicos de forma eficaz, diversificando la reserva de compuestos a analizar en etapas posteriores del proceso de desarrollo de fármacos, sino que además son aliadas indispensables en la evaluación de las propiedades farmacodinámicas de los compuestos candidatos antes de ser sintetizados en el laboratorio. El cribado virtual permite ahorrar tiempo, esfuerzo y recursos, maximizando las probabilidades de éxito y viabilidad química de los compuestos candidatos en su fase preclínica. Además, colabora en la identificación de nuevos blancos farmacológicos, ampliando las posibilidades de diseño de terapias.

Los desafíos enfrentados por las estrategias de cribado virtual se relacionan con las abstracciones computacionales que realizan los modelos empleados en las diversas tareas. En primer lugar, las interacciones farmacodinámicas son complejas y su simulación resulta costosa, por lo que se priorizan aquellos compuestos candidatos identificados en etapas previas del desarrollo, por ejemplo, por

medio de modelos QSAR. Sin embargo, los modelos *in silico* a menudo se basan en supuestos que simplifican los procesos, lo que puede derivar en predicciones inexactas y simulaciones de acoplamiento infructuosas [383]. Asimismo, los procesos biológicos consisten en múltiples blancos terapéuticos que interactúan de formas diversas y en simultáneo, por lo que se suma el desafío de hallar compuestos que interactúen con sitios activos de múltiples estructuras proteínicas a la vez.

### 3.3.4. Analítica visual en informática molecular

La *analítica visual* (AV) es una disciplina de la informática que estudia el uso y diseño de representaciones gráficas y visualizaciones para el análisis de datos a fin de extraer información relevante de los mismos [78, 445]. Esta disciplina está orientada al desarrollo de estrategias de análisis a partir de visualizaciones, que permitan identificar patrones, relaciones y tendencias *a priori* ocultas en la información. La analítica visual se aplica en las más diversas áreas de investigación, resultando crucial en la actualidad, donde se generan enormes volúmenes de información heterogénea y no estructurada a cada instante. En el contexto particular de la informática molecular, las estrategias de analítica visual se centran en el diseño de herramientas que integren representaciones gráficas y visualizaciones de datos moleculares y biológicos, constituyendo un pilar fundamental dado que los espacios moleculares de fármacos potenciales son vastos, diversos, con intrincadas estructuras moleculares e interacciones complejas. La analítica visual en informática molecular simplifica el proceso de análisis de datos complejos y asiste a expertos en la toma de decisiones en el desarrollo de fármacos.

Existen diferentes estrategias de visualización de información química, de las cuales se valen las herramientas de analítica visual para abonar al proceso de desarrollo de un nuevo medicamento. A grandes rasgos, podemos categorizarlas de la siguiente manera:

- Representación molecular [19]: la representación gráfica de estructuras moleculares constituye una pieza clave de la disciplina, incluyendo la visualización de figuras bidimensionales y tridimensionales de las estructuras de compuestos químicos. Por medio de la inspección visual de dichas estructuras es posible analizar conformaciones 3D viables para síntesis, analizar la similitud estructural entre compuestos y diseñar nuevas estrategias para modelar y simular propiedades farmacodinámicas de los mismos. Los primeros antecedentes de representación molecular visual se remiten a los modelos coloquialmente conocidos como modelos de *barras y esferas*, en los cuales se basan las representaciones gráficas actuales (figura 3.8).

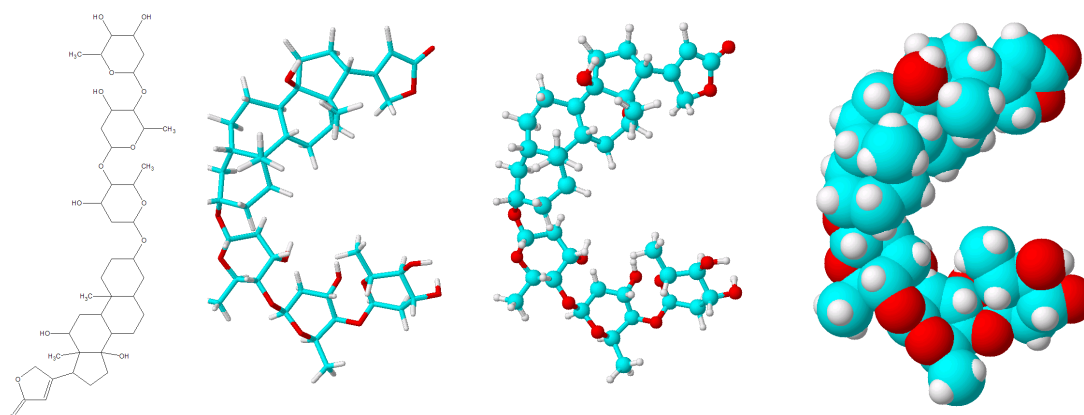


Figura 3.8: Representaciones gráficas en 2D y 3D de la estructura molecular de la *digoxina* ( $C_{41}H_{64}O_{14}$ ), fármaco empleado para el tratamiento de la insuficiencia cardíaca y la arritmia. De izquierda a derecha: representación *wireframe* (2D), de *barras* (2D o 3D), de *barras y esferas* (2D o 3D), y *spacefill* (3D).

- Visualización de grandes volúmenes de datos químicos [142, 323]: teniendo en cuenta los avances tecnológicos en informática y química medicinal, todos los días se generan grandes cantidades de datos. Dada la vastedad del espacio químico explorable, resulta indispensable contar con estrategias de visualización que permitan explorar grandes conjuntos de compuestos de forma simple e informativa. La visualización de grandes volúmenes de datos químicos enfrenta desafíos particulares, vinculados principalmente a la alta dimensionalidad de las representaciones moleculares empleadas y a la carencia de datos etiquetados para muchas propiedades y perfiles de bioactividad. Estas estrategias generalmente son combinadas con técnicas de reducción dimensional, que permitan estudiar las relaciones entre compuestos en espacios bidimensionales y tridimensionales, y se aparejan con algoritmos de *clustering* o agrupamiento en el caso de conjuntos de datos no etiquetados, de forma de encontrar fácilmente patrones estructurales significativos en subespacios químicos. En el capítulo 7 se muestran algunos ejemplos de este tipo de visualizaciones, todos ellos provenientes de un trabajo desarrollado por nuestro grupo de investigación en el marco de la presente tesis [323].
- Visualización de proteínas [402]: las estructuras proteínicas son complejas e intrincadas, por lo que el desarrollo de estrategias para su visualización es fundamental para facilitar la identificación de sitios activos con los que puedan ligar los compuestos candidatos. La figura 3.9 muestra un ejemplo de visualización de una estructura proteínica en 3D.

Las estrategias de analítica visual aplicadas a informática molecular enfrentan retos significativos

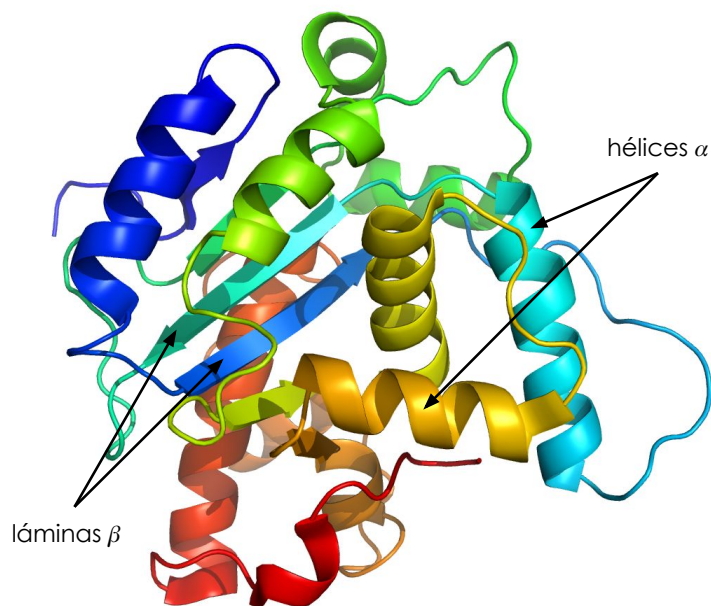


Figura 3.9: Representación gráfica en 3D de la estructura proteínica de la subunidad 2P51, enzima presente en la levadura *Schizosaccharomyces pombe*. Las estructuras helicoidales representan hélices  $\alpha$ , mientras que las flechas planas representan láminas  $\beta$ , ambas subestructuras conformadas por cadenas de aminoácidos. Se emplean diferentes colores para facilitar la identificación de cada una de dichas subestructuras [402].

en términos de la complejidad de los datos a representar y de su escalabilidad, siendo necesario proporcionar herramientas que sean capaces de representar con alto nivel de detalle grupos enormes de compuestos. A su vez, en la mayoría de los casos tienen el requisito adicional de ser interactivas, ya que son empleadas como soporte por expertos en química medicinal, por lo que resulta crucial que sean interoperables y de alto rendimiento. Otro aspecto importante tiene que ver con la integración de múltiples fuentes complementarias de información, que permitan detectar patrones significativos e interpretables en los datos analizados.

### 3.3.5. Diseño *de Novo*

El diseño *de novo* de fármacos, también denominado *diseño racional*, es una disciplina emergente de la informática molecular que engloba el conjunto de técnicas computacionales aplicadas a la creación de compuestos químicos novedosos con propiedades farmacocinéticas y farmacodinámicas específicas [149, 382, 271]. Estas estrategias pueden tanto partir de compuestos existentes,



analizando sus estructuras moleculares y propiedades físico-químicas para identificar determinantes de bioactividad, como crear compuestos químicos desde cero empleando conocimiento experto y modelos computacionales avanzados. El resultado de dichos procesos es la obtención de estructuras moleculares desarrolladas a medida de un objetivo terapéutico. Este enfoque innovador resulta esencial en la búsqueda de alternativas a tratamientos ineficientes o invasivos, para el desarrollo de medicamentos destinados a enfermedades resistentes, complejas o emergentes, y para mitigar los efectos secundarios en tratamientos farmacológicos preexistentes.

El diseño *de novo* ha sido materia de estudio teórico durante varios años en el dominio de la química medicinal y ha ganado tracción en la última década, gracias a los vertiginosos avances en aprendizaje automático y profundo que han habilitado una creciente comprensión de las interacciones moleculares. Existen diferentes estrategias para el diseño *de novo*, las cuales pueden ser en principio analizadas desde la información química en la que se basan, desde la unidad estructural fundamental en la que se basan para construir nuevas estructuras moleculares, o desde el enfoque computacional que emplean [149].

Desde el punto de vista de la información química empleada, podemos distinguir el diseño *basado en estructura* y el diseño *basado en propiedades* [149]. En el enfoque de diseño basado en estructura se emplea la estructura de la proteína objetivo y sus sitios de unión activos, a fin de crear compuestos químicos afines y que sean buenos ligandos de la misma. En el diseño basado en propiedades, la información estructural empleada es la de compuestos conocidos que son ligandos activos de un blanco farmacológico específico, a fin de identificar subestructuras y grupos funcionales determinantes en el perfil de bioactividad y orientar el diseño de un nuevo compuesto al cumplimiento de propiedades físico-químicas determinadas.

Desde el punto de vista de la unidad estructural empleada para la construcción de nuevas moléculas, encontramos los enfoques *basados en átomos* y los enfoques *basados en fragmentos* [149]. Los enfoques pioneros en diseño *de novo* fueron estrictamente basados en átomos, donde la construcción de la estructura molecular es de granularidad fina, seleccionando átomos y enlaces como unidades fundamentales de construcción. Este enfoque es extremadamente flexible y extensible, siendo teóricamente posible modelar la infinitud del espacio químico. Sin embargo, tiene la gran desventaja de ser propenso a la generación de compuestos sintéticamente inviables e inestables, además de la complejidad inherente a la generación y posterior exploración de un vasto espacio molecular. El enfoque basado en fragmentos, por su parte, toma como unidad fundamental de construcción un conjunto amplio de fragmentos moleculares, el cual puede ser personalizado de acuerdo al objetivo farmacológico, reduciendo el posterior espacio molecular y simplificando el proceso de diseño.

Por último, desde el punto de vista del enfoque computacional empleado, las estrategias de diseño *de novo* se basan en diferentes modelos y arquitecturas de aprendizaje automático y profundo, siendo hoy en día las más prominentes aquellas basadas en modelos generativos (ver capítulo 2, sección 2.5.3.6) [382]. El diseño del nuevo compuesto puede darse como resultado de un proceso automático de generación de la estructura molecular en algún formato invertible (SMILES o grafos, por ejemplo) [140, 2], o puede facilitarse por medio de la generación de un espacio multidimensional a partir de cuya exploración se extraigan los nuevos compuestos candidatos [407]. Entre los ejemplos más notables de modelos generativos para el diseño *de novo* de compuestos químicos encontramos *autoencoders* variacionales, redes adversarias generativas, modelos de aprendizaje por refuerzo y modelos de lenguaje [301, 271, 407]. Estos modelos son evaluados en términos de métricas específicas, tales como *unicidad* (*uniqueness*) o la capacidad de generar compuestos nuevos sin repeticiones; *validez* (*validity*), o la capacidad de generar moléculas viables desde un punto de vista químico; y *novedad* (*novelty*), o la capacidad de generar compuestos novedosos que no formen parte del conjunto de compuestos conocidos, empleados para el entrenamiento del modelo [51, 299].

Un aspecto no menor en el diseño *de novo* es la sintetizabilidad de los compuestos generados. Resulta insuficiente que el compuesto formulado presente altas probabilidades de exhibir el perfil de bioactividad deseado, sino que es indispensable que los compuestos generados sean viables desde el punto de vista de la síntesis química [149]. En la literatura, la factibilidad química de los compuestos generados *de novo* continúa siendo un problema de investigación abierto, y las estrategias de diseño *de novo* que consideran este aspecto suelen abordarlo desde dos enfoques. Por un lado, empleando esquemas de reacciones químicas basados en reglas, donde se ensamblan porciones del compuesto de acuerdo a reacciones obtenidas por compuestos conocidos de estructuras similares. Por otro lado, mediante un análisis automático adicional que proponga rutas de síntesis generalizadas y la selección de posibles reactivos de bases de datos de compuestos disponibles.

## 3.4. Síntesis y conclusiones

La informática molecular representa la conjunción entre las prolíficas y fundamentales áreas de informática, biología, química y farmacia, todas ellas propulsadas por el constante avance científico-tecnológico y el desarrollo de nuevos paradigmas de investigación signados por el aprendizaje profundo [257]. A lo largo del presente capítulo hemos presentado algunos de los pilares fundamentales del desarrollo de fármacos, que dan soporte al desarrollo de las estrategias computacionales para quimioinformática de la actualidad: técnicas de representación molecular, modelado predictivo

de propiedades físico-químicas y bioactividad, cribado virtual de fármacos y diseño *de novo* de compuestos químicos.

Sin duda, el modelado QSAR constituye uno de los ejes centrales de la investigación en química medicinal, siendo la piedra angular detrás de la búsqueda y optimización de fármacos candidatos en etapas preliminares a la síntesis química. En el capítulo 4 presentamos un enfoque integral de modelado QSAR basado en redes neuronales profundas (DNNs) que permite el entrenamiento a partir de representaciones moleculares de alta dimensionalidad, brindando además un método de estimación del dominio de aplicabilidad basado en estimación de confianza y una estrategia de interpretabilidad que permite analizar la contribución particular de cada atributo de entrada en el proceso de inferencia del modelo.

# Capítulo 4

## Enfoque integral de modelado QSAR empleando estrategias de aprendizaje profundo

El modelado QSAR es una de las áreas de investigación más importantes del a informática molecular, siendo un determinante en la identificación temprana de compuestos candidatos en el proceso de desarrollo de fármacos. En este capítulo exploramos una propuesta para el modelado de tres propiedades relevantes a la química medicinal, empleando representaciones moleculares de alta dimensionalidad y modelos basados en redes neuronales profundas (DNNs). Nuestro enfoque de modelado permite además la definición del dominio de aplicabilidad mediante la estimación de confianza en la predicción e incorpora una estrategia de interpretabilidad *a posteriori*.

---

### 4.1. Introducción

La integración de las ciencias computacionales en la industria farmacéutica y biomédica ha generado múltiples aplicaciones y avances tecnológicos durante las últimas décadas [88, 246]. En la búsqueda por optimizar el largo y costoso proceso de descubrimiento y desarrollo de nuevos fármacos, las herramientas computacionales han permitido acelerar la identificación y priorización de compuestos candidatos, reduciendo costos y mejorando la tasa de éxito [66, 232]. Una de las tareas claves en el proceso de descubrimiento de fármacos *in silico* es diseñar modelos para predecir la actividad biológica y las propiedades físico-químicas de los compuestos candidatos a fármacos.

Como vimos en el capítulo 3, estos modelos, denominados de Relación Cuantitativa Estructura-Actividad (QSAR, por sus siglas en inglés), predicen la relación entre las características o descriptores moleculares que codifican la estructura molecular de los compuestos y la propiedad o actividad biológica objeto de estudio.

El desarrollo de modelos QSAR generalmente implica tratar con representaciones de datos de alta dimensionalidad. Los compuestos candidatos evaluados en el proceso de desarrollo de fármacos suelen representarse o describirse mediante un gran número de características moleculares, que codifican sus propiedades estructurales y físico-químicas [102, 300], y cada una de dichas características moleculares puede adoptar un valor en un rango amplio de posibles valores. Por lo tanto, comúnmente es necesario emplear técnicas de selección de características. Generalmente, el proceso de selección de características se aplica de forma previa al desarrollo de un modelo QSAR dado que las técnicas tradicionales de aprendizaje automático suelen experimentar dificultades en el aprendizaje en escenarios de alta dimensionalidad [126]. Sin embargo, considerando la gran variedad de posibles descriptores moleculares y características que se pueden calcular a partir de las estructuras moleculares de los compuestos químicos, el proceso de selección de características representa un problema desafiante en términos combinatoriales. La selección de características se realiza, o bien de forma manual, lo cual requiere labor y conocimiento expertos y permite la consideración de un espacio de descriptores moleculares acotado, con el consecuente potencial de pérdida de información valiosa para el modelado [222], o bien de forma automática, con lo cual no se puede garantizar que la selección no sufra sesgos introducidos por la técnica computacional elegida para dicha tarea.

Una vez entrenado, el propósito de un modelo QSAR es predecir propiedades de compuestos nuevos, es decir, nunca vistos por el modelo en el transcurso del proceso de entrenamiento. Por lo tanto, un aspecto importante del proceso de modelado es estimar la confiabilidad de las predicciones sobre compuestos nuevos. Como vimos en el capítulo 3 de la presente tesis, el **dominio de aplicabilidad** de un modelo QSAR es la región o conjunto de regiones del espacio molecular en el cual se espera que las predicciones realizadas por el modelo QSAR sean precisas. El análisis del dominio de aplicabilidad es un paso importante en el proceso de construcción de un modelo QSAR confiable y la estimación del dominio de aplicabilidad de un modelo QSAR constituye un tópico de investigación abierto, que implica una serie de desafíos debido a la extensión del espacio químico [318, 200, 189].

Otro aspecto a considerar en el proceso del desarrollo de un modelo QSAR es su interpretabilidad. Tal y como hemos discutido anteriormente, los modelos QSAR son empleados por expertos del área de farmacia y bioquímica para la búsqueda de fármacos candidatos, por lo que alcanzar modelos

interpretables resulta crítico en el proceso de selección de modelos predictivos [183]. En principio podemos concebir dos niveles diferentes de aproximación al concepto de la interpretabilidad de modelos predictivos en el dominio químico. Por un lado, podemos definirla en términos de la comprensión del proceso matemático y algorítmico que el modelo sigue para realizar inferencia o predicción. Esta tarea puede ser abordada por los denominados *métodos de inteligencia artificial explicable*, que permiten construir explicaciones sobre el funcionamiento de un modelo QSAR, o la forma en que este alcanza una predicción para una entrada específica, mediante diversas estrategias que van desde identificar cómo contribuyen las distintas variables del modelo en la obtención de una predicción hasta el uso de métodos de razonamiento contrafactual o modelos subrogados [281, 420]. Por otro lado, podemos entender la interpretabilidad de un modelo en términos de la capacidad de los expertos para evaluar la incidencia de las representaciones de entrada de los datos empleados para su entrenamiento en sus resultados, a partir de la comprensión de la semántica asociada a dichos datos en relación con la actividad biológica o fenómeno fisicoquímico que el modelo está intentando predecir. Esta última definición de interpretabilidad es especialmente valiosa en el dominio de la química medicinal, puesto que más allá de la comprensión del proceso de entrenamiento y las relaciones matemáticas establecidas por el modelo QSAR, resulta fundamental para los expertos vislumbrar potenciales hipótesis sobre los principios fundamentales que gobiernan los fenómenos estudiados y sus relaciones con las características estructurales de las moléculas.

En otras palabras, existe un nivel de interpretabilidad más abstracto en términos del dominio químico que se sustenta en los insumos generados por las estrategias de inteligencia artificial explicable, como puede ser la identificación de las características o descriptores moleculares más relevantes, pero que demanda además la integración de conocimientos de química medicinal que pueden exceder incluso a la información extraíble de los datos utilizados para generar el modelo QSAR. En este sentido, la selección de características moleculares tiene la ventaja de simplificar la interpretabilidad del modelo, ya que permite reducir la complejidad del trabajo de los expertos al reducir el conjunto de características a analizar, el cual quedará constituido por solo aquellas que resultan *a priori* más relevantes para entrenar el modelo QSAR [76]. Sin embargo, el aprendizaje profundo y los modelos predictivos basados en redes neuronales profundas (*DNN*, por sus siglas en inglés), si bien cuestionados en términos de su baja interpretabilidad con respecto al proceso matemático de aprendizaje [16], sí presentan características que posibilitan la construcción de explicaciones en el dominio químico sin el riesgo de eliminar características moleculares prematuramente, las cuales podrían albergar información valiosa para el entrenamiento y comprensión del modelo QSAR.

## 4.2. Enfoque integral para modelado QSAR basado en aprendizaje profundo

Por lo expuesto en la sección anterior, resulta de interés considerar estrategias y técnicas de modelado que permitan el entrenamiento en espacios de alta dimensionalidad, a la vez que proporcionen formas intuitivas de estimar el dominio de aplicabilidad del modelo generado y de interpretar los descriptores moleculares de mayor incidencia en el proceso de entrenamiento. Las redes neuronales artificiales se convirtieron durante la última década en una de las técnicas de aprendizaje automático más prominentes [104], aunque su adopción en modelado QSAR ha sido criticada por la dificultad en la interpretación de los resultados [32]. Con el reciente auge del aprendizaje profundo y los avances en las más diversas arquitecturas de DNNs, los modelos basados en redes neuronales son hoy en día menos propensos a sobreajuste y son efectivos para resolver problemas de análisis de datos a gran escala y de alta dimensionalidad [33], por lo que se perfilan como estrategia viable para el modelado QSAR, por medio de la cual es posible abordar todos los desafíos planteados en un enfoque integral.

En este escenario, desarrollamos una propuesta integral para modelado QSAR que consistió en la construcción de modelos basados en DNNs para la predicción de tres propiedades relevantes en las ciencias biomédicas y en un análisis comparativo de estos modelos con el estado del arte [322]. Como parte de nuestro trabajo, exploramos una estrategia de estimación del dominio de aplicabilidad de los modelos propuestos por medio de la utilización de las probabilidades de salida del modelo QSAR basado en DNNs, basando la estimación en la confiabilidad de la predicción. Además, propusimos un método de análisis visual *a posteriori* de las características y descriptores moleculares empleados para representar a los compuestos candidatos durante el entrenamiento de los modelos, el cual permite identificar aquellas características moleculares que desempeñan un papel importante en la definición del perfil de actividad biológica del compuesto químico.

Las contribuciones presentadas en este capítulo se resumen a continuación:

- Aplicamos avances recientes en DNNs al desarrollo de modelos QSAR y obtuvimos mejoras significativas en el rendimiento en comparación con otras técnicas utilizadas anteriormente en estos conjuntos de datos, sin necesidad de llevar a cabo una fase de selección de características.
- Demostramos la efectividad de usar las probabilidades de salida de una red neuronal como medio de estimación del dominio de aplicabilidad del modelo QSAR, lo que constituye un enfoque

intuitivo y representa un avance sobre otros métodos que únicamente indican la inclusión o exclusión de un compuesto determinado en el dominio de aplicabilidad del modelo.

- Propusimos un método de interpretabilidad *a posteriori* basado en un análisis agregativo de los parámetros entrenables de la red neuronal, el cual permite determinar cuáles son los descriptores moleculares y características más influyentes en el proceso de aprendizaje. Este método no ha sido aplicado previamente en el contexto de modelado QSAR. Presentamos los resultados de esta estrategia por medio de una visualización basada en mapas de calor.

### 4.3. Trabajo relacionado con la propuesta

Durante las últimas décadas, el proceso de diseño racional de fármacos se ha basado en técnicas de aprendizaje automático para las tareas de modelado [169, 372, 157]. Algoritmos tradicionales de aprendizaje automático, tales como *Support Vector Machine (SVM)*, *Decision Trees*, *Naïve Bayes* y *k-Nearest Neighbors (kNN)* han sido ampliamente utilizados debido a su relativo buen rendimiento y simplicidad [380, 22, 242]. Además, se ha observado una creciente tendencia en la comunidad a los enfoques de modelado QSAR basados en consenso, que consisten en ensamblar diferentes clasificadores base para evaluar sus predicciones de forma combinada y, en consecuencia, aumentar las capacidades de predicción del modelo [213, 106, 218, 10, 321]. Este tipo de modelos se encuentran típicamente entre las técnicas de mayor rendimiento para la predicción de varias propiedades químicas en el modelado QSAR, aunque a expensas de una interpretabilidad de los resultados limitada y de comúnmente requerir de un paso previo de selección de características moleculares para lograr tal rendimiento [144, 224].

La propuesta integral presentada en este capítulo fue evaluada en tres casos de estudio. Por un lado, se desarrollaron modelos predictivos para la interacción entre compuestos químicos y los *citocromos P450* en sus isoformas *2C9* y *3A4*, que son una familia de enzimas relacionadas con el metabolismo de fármacos [415]. Por otro lado, se evaluó la propuesta en la predicción de Biodegradabilidad Simple (*RB*), propiedad definida como la capacidad de degradación de una sustancia en contacto con microorganismos inoculados por un período de tiempo específico [137], de gran relevancia para las ciencias biomédicas por su relación con la toxicidad por sobre-exposición [61, 315].

Respecto a los antecedentes científicos relacionados con los casos de estudio abordados por nuestro trabajo, la mayoría de los modelos QSAR desarrollados previamente para predecir la interacción



entre compuestos químicos y el *citocromo P450* involucran selección de características y utilizan algoritmos de aprendizaje automático tradicionales. Varios de ellos proponen modelos basados en consenso, tales como Cheng et al. [67], Shah et al. [344] y Nembri et al. [279], exhibiendo este último trabajo el mejor desempeño al momento del desarrollo de nuestra propuesta. En todos estos trabajos se emplearon representaciones moleculares basadas en descriptores moleculares tradicionales y *fingerprints* de conectividad extendida (*ECFP*) [311], ambas representaciones presentadas en el capítulo 3. Con respecto al caso de estudio de Biodegradabilidad Simple, la mayoría de los enfoques de modelado propuestos se basan en consensos de clasificadores simples o en redes neuronales [200, 241, 107, 63, 28, 123], empleando además procesos de selección de características. En particular, el trabajo de Mansouri et al. [241] obtuvo el mejor desempeño en el caso de estudio analizado al momento del desarrollo de nuestra propuesta.

Con respecto a las tecnologías de aprendizaje automático empleadas en el área, las técnicas de aprendizaje profundo han sido crecientemente adoptadas por la comunidad científica para el desarrollo de modelos QSAR y para otras tareas en el proceso de descubrimiento de fármacos y hoy en día se han establecido entre las estrategias predominantes en el área [104]. Si bien el desarrollo de modelos QSAR basado en redes neuronales artificiales no es necesariamente novedoso [424, 157], tal y como hemos discutido en el capítulo 2, el surgimiento de nuevas estrategias para entrenar este tipo de modelos, tales como la aplicación de técnicas para evitar el sobreajuste y el desvanecimiento/explosión de gradientes durante el entrenamiento, ha cambiado radicalmente el panorama en términos del rendimiento predictivo esperable y alcanzable por los modelos QSAR actuales. Aunque la aplicación de técnicas de aprendizaje profundo en el modelado QSAR es un fenómeno relativamente reciente con respecto a la trayectoria científica en el área, se han publicado numerosos trabajos en los que se han desarrollado con éxito modelos basados en aprendizaje profundo para descubrimiento de fármacos [231, 219, 203], en los que se concluyó que dichos modelos superan estadísticamente a aquellos modelos basados en métodos tradicionales de aprendizaje automático en diversos casos de estudio [231, 219, 203]. Por caso, Ma et al. [231] propusieron modelos basados en DNN logrando un mayor rendimiento predictivo que modelos basados en *Random Forest* en variados conjuntos de datos químicos. Por su parte, Lenselink et al. [219] compararon cinco estrategias de modelado diferentes sobre un conjunto de referencia de bioactividad extraído de la base de datos pública *ChEMBL* [116] y encontraron que los modelos basados en DNN superaron a los métodos tradicionales. También demostraron que un ensamble de DNN es capaz de mejorar aún más el rendimiento obtenido por modelos basados en DNN simples. Koutsoukas et al. [203] mostraron que los modelos basados en DNN superan estadísticamente a los modelos basados en métodos tradicionales en diversos conjuntos de datos.

Con respecto a la determinación del dominio de aplicabilidad de modelos QSAR, la mayoría de los artículos de investigación en el área abordan la temática utilizando un método independiente al empleado para el modelado, adoptando diferentes estrategias y medidas estadísticas para determinar sus límites [200]. La mayoría de ellos se concentran en definir diferentes criterios de similitud entre moléculas para detectar valores atípicos, que luego son adoptados como criterio para excluir compuestos del dominio de aplicabilidad del modelo [317, 228, 42]. En particular, Klingspohn et al. [200] definió una taxonomía de los métodos para estimación de dominio de aplicabilidad e identificó dos categorías principales de técnicas: aquellas basadas en identificación de valores atípicos (detección de novedad o *novelty detection*) y aquellas basadas en inferencia a partir del modelo entrenado (estimación de confianza o *confidence estimation*). Se concluyó que aquellas técnicas para estimación de dominio de aplicabilidad basadas en la estimación de confianza funcionan mejor que las técnicas de detección de novedad y, por lo tanto, resultan adecuadas para definir el dominio de aplicabilidad de un modelo QSAR.

Con relación a la interpretabilidad de los modelos QSAR, el estudio de la temática—particularmente en el caso de modelos basados en redes neuronales—constituye un tópico de interés dentro de la comunidad de aprendizaje automático [350, 398, 194] y continúa siendo un tema de investigación y debate en el área de descubrimiento de fármacos. El propósito de un modelo QSAR es brindar soporte a expertos en el proceso de selección de compuestos candidatos durante el diseño y descubrimiento de fármacos, por lo que resulta deseable que sus resultados sean interpretables [298]. Los enfoques basados en consenso, a pesar de tener un buen desempeño predictivo, tienden a carecer de interpretabilidad, ya que su resultado de salida es una combinación de diferentes clasificadores base. En la bibliografía se han explorado algunos enfoques basados en la interpretabilidad *a posteriori* [261, 24, 310], los cuales toman el modelo entrenado y apuntan a brindar interpretabilidad a sus predicciones en términos de las entradas del modelo. En lugar de basarse en una comprensión algorítmica a bajo nivel del modelo, el cual es el enfoque más habitualmente adoptado para el análisis de interpretabilidad, las técnicas *a posteriori* tienen como objetivo caracterizar el comportamiento del modelo predictivo sin intentar explicar su representación y operaciones internas, sino proporcionando una descripción funcional de cómo impactan las entradas al modelo en el entrenamiento y/o en la predicción final. En esta línea, Tsang et al. [388] propusieron analizar las interacciones estadísticas entre las entradas de una red neuronal de múltiples capas durante el proceso de entrenamiento por medio de la interpretación directa de sus pesos aprendidos. Su método demostró ser efectivo tanto en conjuntos de datos de aplicaciones sintéticas como del mundo real, si bien su estudio no abordó conjuntos de datos químicos.

Cabe destacar que al momento de la compilación y redacción de la presente tesis han sido propuestos nuevos trabajos de investigación sobre modelado QSAR para los casos de estudio abordados en nuestra propuesta [125, 291, 114], así como también en torno a la temática de determinación del dominio de aplicabilidad de modelos QSAR [374, 413, 306] e interpretabilidad de modelos QSAR [55, 30].

## 4.4. Nuestra propuesta

Sobre la base del estado del arte y a las limitaciones observadas en los trabajos de referencia, desarrollamos una propuesta integral que involucra el desarrollo de modelos QSAR basados en DNNs junto con la estimación de sus dominios de aplicabilidad y un método para brindar interpretabilidad desde un punto de vista del dominio químico a cada uno de dichos modelos [322]. Los casos de estudio que abordamos fueron Biodegradabilidad Simple (*RB*) y la interacción entre compuestos candidatos y el *citocromo P450* para las isoformas *CYP2C9* y *CYP3A4*. A fines de validar nuestros resultados, comparamos nuestros modelos QSAR con los modelos previamente publicados que constituyeran el estado del arte para dichos casos de estudio [279, 241]. Además, presentamos una técnica de definición del dominio de aplicabilidad basada en estimación de confianza derivada de los modelos entrenados. Por último, aplicamos una técnica de interpretabilidad *a posteriori* basada en una agregación simple de los parámetros entrenables de las DNNs, que permitió analizar las contribuciones particulares de cada una de las entradas de los modelos. Según nuestro entender tras una exhaustiva revisión bibliográfica, nuestra propuesta fue pionera en emplear este método para brindar interpretabilidad a modelos QSAR. El código fuente los conjuntos de datos empleados en el desarrollo de nuestro trabajo son públicamente accesibles<sup>1</sup>.

La figura 4.1 resume el diseño experimental de nuestro trabajo: primero, procesamos los tres conjuntos de datos asociados a los casos de estudio (a). En segundo lugar, enriquecimos los conjuntos de datos originales agregando nuevos descriptores moleculares (b) y dividimos los conjuntos de datos originales y enriquecidos en particiones fijas para entrenar y validar nuestros modelos, siguiendo los modos de particionado empleados en los trabajos de referencia (c). Luego, desarrollamos nuestros modelos basados en DNNs siguiendo un proceso iterativo de búsqueda de hiperparámetros, seleccionamos los mejores modelos y entrenamos réplicas de los mismos (d). Luego evaluamos su desempeño utilizando diferentes métricas y contrastamos estos resultados con los trabajos de referencia (e). Finalmente, desarrollamos una estrategia de definición del dominio de aplicabilidad de

---

<sup>1</sup>Recursos: <https://github.com/VirginiaSabando/DNN-QSAR-2019.git>

los modelos basado en la estimación de confianza y aplicamos un método de análisis de características *a posteriori* que permite determinar las características más influyentes en la predicción final (f).

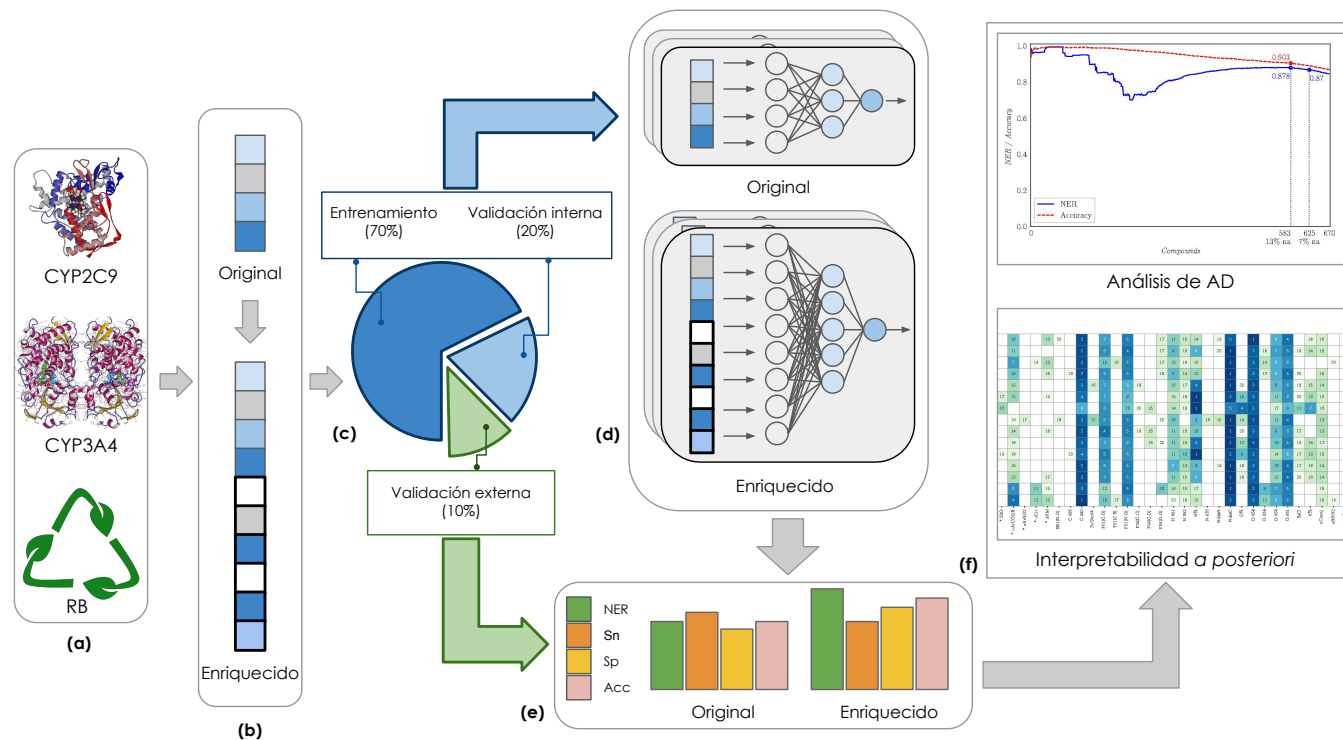


Figura 4.1: Representación del diseño experimental de nuestra propuesta: (a) preprocesamiento de datos asociados a los tres casos de estudio, (b) enriquecimiento del conjunto de datos, (c) partición del conjunto de datos, (d) desarrollo y entrenamiento de modelos, (e) evaluación de modelos, (f) análisis del dominio de aplicabilidad y análisis de características *a posteriori*.

## 4.5. Casos de estudio, metodología y diseño experimental

En esta sección del presente capítulo proporcionamos una descripción general de todas las técnicas utilizadas para el desarrollo de nuestro enfoque integral de modelado, así como la preparación de los conjuntos de datos y la selección de los hiperparámetros del modelo.

Para los tres casos de estudio, partimos de los conjuntos de datos originalmente utilizados en los trabajos de referencia [279, 241], los cuales caracterizan a los compuestos químicos en términos de un conjunto pequeño de descriptores moleculares. Luego, enriquecimos los conjuntos de datos incluyendo nuevos descriptores moleculares pertenecientes a las mismas familias de descriptores 0–2D

ya incluidos en los conjuntos de datos originales. Trabajamos con las mismas particiones de datos reportadas por los trabajos de referencia: Entrenamiento (*Train*), Validación interna (*Validation*) y Validación externa (*Held-out*). En pos de la exhaustividad experimental, también entrenamos nuestros modelos utilizando validación cruzada de cinco pliegos (*five-fold Cross Validation*).

Todos nuestros modelos se basan en redes neuronales multicapa de propagación hacia adelante *feed-forward*. Desarrollamos modelos para los conjuntos de datos originales, tal y como habían sido utilizados en los trabajos de referencia, y para los conjuntos de datos enriquecidos. La arquitectura de cada modelo varía según el número de descriptores moleculares de entrada al modelo: los conjuntos de datos enriquecidos tienen entradas al modelo de mayor cardinalidad y, por ende, requieren arquitecturas neuronales más complejas, por lo que sus modelos poseen más nodos que los desarrollados para las versiones originales de los conjuntos de datos. Nuestros modelos basados en DNN se obtuvieron siguiendo un proceso de dos fases (figura 4.1-d). La primera consistió en una fase exploratoria, donde consideramos diferentes arquitecturas y estrategias de optimización, desarrollamos prototipos y ajustamos sus hiperparámetros. La segunda fase consistió en seleccionar el mejor prototipo de la primera fase evaluando el desempeño de los modelos entrenados en la primera fase en el conjunto de Validación interna. Debido a la estocasticidad inherente de las redes neuronales, repetimos el proceso de entrenamiento quince veces variando la semilla aleatoria utilizada para la inicialización de los parámetros entrenables de la red neuronal. Como resultado de este proceso de dos fases, obtuvimos quince modelos para cada caso de estudio. Finalmente, reportamos el desempeño promedio de estos quince modelos (*Average model*) en la partición de Validación externa, seleccionando como mejor modelo (*Best model*) a aquel que obtuviese el mejor desempeño.

Basándonos en los hiperparámetros del modelo *Best*, construimos un modelo para las versiones original y enriquecida de cada conjunto de datos, a saber, *Best\_O* y *Best\_E*, usando menos nodos por capa para la versión original. Esto nos permitió comparar el desempeño de nuestra estrategia sobre un mismo conjunto de compuestos con y sin la aplicación de un proceso de selección de características, y a la vez estimar el potencial de nuestro enfoque en conjuntos de datos de alta dimensionalidad.

### 4.5.1. Conjuntos de datos

Los conjuntos de datos *CYP2C9* y *CYP3A4* comprenden un total de 11.940 y 12.118 compuestos respectivamente, siendo la proporción entre compuestos activos/inactivos en sus particiones de Entrenamiento y Validación interna de 49/100 para *CYP2C9* y 66/100 para *CYP3A4*. En cuanto a las particiones de Validación externa, las proporciones son 56/100 en *CYP2C9* y 98/100 en *CYP3A4*.

CYP2C9 y CYP3A4 comparten los mismos compuestos en sus particiones de Entrenamiento, así como en sus conjuntos de Validación interna. En los conjuntos de datos originales, CYP2C9 incluye diez descriptores moleculares, mientras que CYP3A4 incluye ocho descriptores moleculares. También incluyen un *fingerprint ECFP* de 1024 bits para cada compuesto [311]. Para las versiones enriquecidas de ambos conjuntos de datos, agregamos un total de 2.701 descriptores moleculares al conjunto de datos CYP2C9 y 2.699 descriptores moleculares al conjunto de datos CYP3A4, lo que lleva a los conjuntos de datos a un total de 2.711 y 2.707 descriptores moleculares, respectivamente, además del *fingerprint ECFP* de 1024 bits. Realizamos el cálculo de descriptores moleculares y ECFP de radio 2 utilizando Dragon 7 [358]. Hubo algunos compuestos en los conjuntos de datos proporcionados por Nembri et al. [279] cuyos códigos SMILES no se formaron correctamente y, por lo tanto, no pudimos calcular sus descriptores moleculares. Producto de este filtrado, se eliminó un compuesto de ambas particiones de Entrenamiento, seis compuestos de la partición de Validación externa de CYP2C9 y un compuesto de la partición de Validación externa de CYP3A4.

El conjunto de datos *RB* comprende 1.725 compuestos, donde la relación entre compuestos activos/inactivos es 51/100 para la partición de Entrenamiento, 49/100 para la partición de Validación interna y 40/100 para la partición de Validación externa. En su versión original, el conjunto de datos *RB* cuenta con un total de 41 descriptores moleculares calculados, mientras que su versión enriquecida cuenta con 1.480 descriptores moleculares. Calculamos estos descriptores utilizando Dragon 7 [358].

#### 4.5.2. Construcción de los modelos QSAR

Las capas de entrada de nuestros modelos QSAR comprenden un nodo por cada descriptor molecular y bit del *fingerprint ECFP*. Utilizamos unidades lineales rectificadas (ReLU) como función de activación de los nodos de las capas ocultas de las DNNs [275], mientras que la capa de salida implementa una función *softmax* con un nodo por clase. Empleamos tamaño de *minibatch*  $mb = 200$  y el optimizador Adam [198] durante el entrenamiento. Para la inicialización de los parámetros entrenables de las DNN, empleamos las inicializaciones Xavier/Glorot [121] y He Normal [150]. También aplicamos *normalización por lotes* (*Batch normalization*) [173] empleando un parámetro de escalado  $\gamma = 0,9$  en todas las capas de nuestros modelos para acelerar el entrenamiento y evitar explosión de gradientes. Entre las técnicas de regularización empleadas para evitar sobreajuste, empleamos *dropout* [357], *early stopping* y regularización *L2* [160] variando el coeficiente de penalización  $\lambda$  en cada modelo.

El modelo QSAR obtenido para el conjunto de datos enriquecido CYP2C9 es una arquitectura de red neuronal multicapa de propagación hacia adelante que consta de una capa de entrada de 3.735 nodos (para 2.711 descriptores moleculares más 1.024 bits de ECFP) y tres capas ocultas de 50, 20 y 5 nodos. Utilizamos *learning rate*  $\alpha = 0,00001$ , inicialización Xavier/Glorot y coeficiente de penalización  $\lambda = 0,0001$  en la regularización L2. Para lidiar con el desequilibrio de clases, optimizamos una función de costo ponderada, que penaliza las instancias mal predichas de la clase minoritaria, aplicando un factor proporcional al desequilibrio de clases observado en el conjunto de entrenamiento.

En el caso del conjunto de datos enriquecido CYP3A4, la arquitectura del modelo es similar a la utilizada para CYP2C9, con la diferencia de que la capa de entrada consta de 3.731 nodos (para 2.707 descriptores moleculares más 1.024 bits de ECFP). Utilizamos inicialización He Normal, y adoptamos los mismos criterios de regularización y *learning rate* que en el caso de CYP2C9. Para mitigar el desequilibrio de clases, en este caso de estudio aplicamos una técnica de muestreo estratificado, donde se extrajo un número igual de compuestos pertenecientes a cada clase para construir cada uno de los *minibatches* durante el entrenamiento. Se tomaron muestras al azar de los compuestos con reemplazo del conjunto de entrenamiento antes de alimentarlos al modelo durante la fase de entrenamiento.

Por último, para el conjunto enriquecido RB la capa de entrada consta de 1.480 nodos para descriptores moleculares, y la DNN está formada por tres capas ocultas de 20, 10 y 5 nodos, respectivamente. Utilizamos un valor de *learning rate*  $\alpha = 0,0001$ , inicialización Xavier/Glorot y coeficiente de penalización  $\lambda = 0,001$  en la regularización L2. Al igual que para CYP3A4, utilizamos una técnica de muestreo estratificado para contrarrestar el desequilibrio entre las clases activo/inactivo.

### 4.5.3. Dominio de aplicabilidad

El enfoque integral que desarrollamos utiliza las probabilidades de la capa de salida de nuestros modelos para estimar su dominio de aplicabilidad. Cada compuesto es clasificado por el modelo QSAR como *activo* o *inactivo*. Esa clasificación es realizada en función de la probabilidad de pertenencia a cada una de esas clases, calculada por el propio modelo: una probabilidad de salida entre 0 y 0,5 corresponde a un resultado de clasificación *inactivo*, mientras que una probabilidad de salida entre 0,5 y 1 corresponde a un resultado de clasificación **activo**. Es importante notar que los valores extremos (cerca de 0 para la clase *inactiva* y cerca de 1 para la clase *activa*) corresponden a predicciones **más confiables** del modelo QSAR, en las que el modelo fue capaz de realizar la inferencia con menor

error, mientras que aquellas probabilidades de salida cercanas a 0,5 (para ambas clases) corresponden a predicciones **menos confiables**.

Para definir el dominio de aplicabilidad de los modelos QSAR clasificamos los compuestos en el conjunto de datos y así obtenemos la probabilidad de salida asociada a cada compuesto, la cual es arrojada por la capa de salida de las DNNs. Luego, elaboramos un *ranking* entre todos los compuestos, ordenándolos en función de sus probabilidades de salida **en orden decreciente de confiabilidad**: de predicciones más confiables (probabilidades cercanas a los valores extremos 0—*inactivos*—y 1—*activos*) a predicciones menos confiables (probabilidades cercanas a 0,5). Evaluamos el desempeño de los modelos de clasificación en función de cuatro métricas, todas ellas descritas en el capítulo 2: *Exactitud* (Acc, Ec. 2.20), *Sensibilidad* (Sn, Ec. 2.21), *Especificidad* (Sp, Ec. 2.22) y *Tasa de No Error o Exactitud Balanceada* (NER/BAcc, Ec. 2.24).

Dado que nuestra propuesta consiste en un enfoque integral, que incorpore la definición del dominio de aplicabilidad en la evaluación del rendimiento del modelo QSAR, dichas métricas no fueron computadas sobre la totalidad de los compuestos en el conjunto de datos. En cambio, nos basamos en la clásica métrica *Mean Average Precision (MAP)* [240] (Ec. 4.1) y evaluamos el desempeño del modelo teniendo en cuenta el *ranking* de compuestos elaborado anteriormente: se computa el valor de cada métrica a intervalos seleccionados de dicho ranking, donde cada intervalo comprende los  $k$  compuestos clasificados **con mayor confiabilidad** por el modelo (valores de probabilidad de salida más cercanos a los extremos 0 y 1), independientemente de si la predicción corresponde a la clase *activo* o *inactivo*. La cantidad de compuestos  $k$  es un número entero comprendido en el intervalo abierto  $[1, n]$ , siendo  $n$  el número total de compuestos del conjunto de datos. Como resultado de este procedimiento, se obtienen  $n$  mediciones de rendimiento del modelo, una por cada intervalo de compuestos, donde la primera medición corresponde al intervalo  $k = 1$ , que contiene únicamente el primer compuesto predicho con más confiabilidad por el modelo, siguiendo con el intervalo  $k = 2$  que contiene los dos primeros compuestos predichos con mayor confiabilidad, y así sucesivamente hasta el intervalo  $k = n$  que contiene todos los compuestos del conjunto de datos.

$$MAP = \frac{1}{n} \sum_{k=1}^n f(k) \quad (4.1)$$

donde  $f(k)$  es una de las cuatro métricas de rendimiento detalladas anteriormente, evaluada en el  $k$ -ésimo intervalo de los  $n$  intervalos definidos por los  $k$  compuestos de mayor probabilidad de salida.

Finalmente, a los efectos de evaluar el desempeño del modelo dentro del alcance de su dominio de aplicabilidad, se promedian los valores de la métrica computados sobre los  $n$  intervalos. De esta



forma, el valor  $k$  puede variarse para comprender más o menos compuestos en la evaluación, actuando así como un umbral de confiabilidad que puede ser modificado de acuerdo el caso de estudio. El valor  $k$  define el alcance del dominio de aplicabilidad del modelo: a medida que  $k$  aumenta, la medición de rendimiento del modelo comprende más compuestos cuyas predicciones son menos confiables, por lo que la definición del dominio de aplicabilidad del modelo es más laxa y, como consecuencia, resulta esperable que el rendimiento del modelo sea inferior.

Presentamos un resumen gráfico de la metodología adoptada para definir el dominio de aplicabilidad de los modelos en la figura 4.2.

#### 4.5.4. Análisis de características moleculares *a posteriori*

Dado que los modelos QSAR brindan soporte a expertos en el desarrollo de fármacos, es deseable que cualquier modelo QSAR propuesto sea interpretable [298, 247]. Para los especialistas del dominio resulta útil conocer cuáles son las características que hacen que una familia particular de compuestos exhiba cierto perfil de actividad respecto a una propiedad de interés, ya que esto permite reducir el espacio de búsqueda y orientar la exploración durante el proceso de descubrimiento de fármacos.

Dentro de nuestro enfoque integral, proponemos una técnica de análisis de características *a posteriori* como una forma de proporcionar interpretabilidad a nuestros modelos neuronales, de forma tal que es posible determinar cuáles fueron los descriptores moleculares más relevantes para el proceso de entrenamiento del modelo.

Una vez completado el entrenamiento del modelo, calculamos una puntuación para cada descriptor molecular y bit del *fingerprint ECFP* desde los nodos de entrada hacia los nodos de salida. Para un cierto descriptor molecular, al cual le corresponde un nodo de entrada determinado, esta puntuación se calcula agregando los pesos del modelo neuronal que están conectados a dicha entrada. Para cualquier capa de la DNN, la puntuación de un nodo  $j$  se calcula como:

$$S(n_j) = \frac{1}{p} \sum_{i=1}^p |w_{j,i}| S(n_i) \quad (4.2)$$

que es el promedio de los  $p$  productos entre los pesos  $w_{j,i}$  que conectan el nodo  $j$  con los  $p$  nodos de la siguiente capa y sus respectivas puntuaciones. Dado que esta es una definición recursiva, al establecer la puntuación del nodo correspondiente a la clase **activo** de la capa de salida en 1, podemos calcular todos los puntajes de los nodos de la DNN comenzando desde los nodos de salida y retrocediendo en

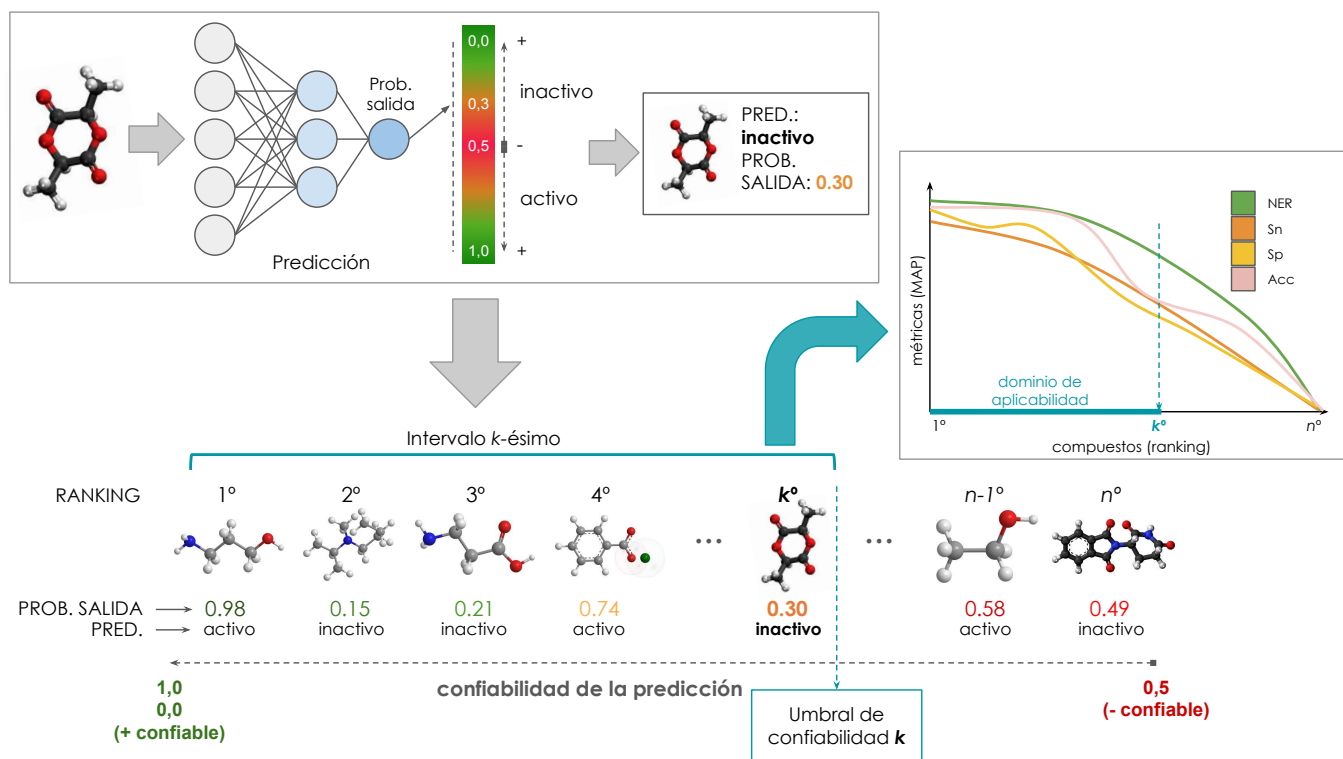


Figura 4.2: Estrategia de definición del dominio de aplicabilidad: Primero, clasificamos los compuestos en el conjunto de datos y así obtenemos la probabilidad de salida asociada a cada compuesto. Las probabilidades de salida cercanas a 0 (predicción *inactivo*) y a 1 (predicción *activo*) son predicciones **confiables** del modelo QSAR, mientras que aquellas probabilidades de salida cercanas a 0,5 (*activo* o *inactivo*) corresponden a predicciones **no confiables**. Luego, elaboramos un *ranking* entre todos los compuestos, ordenándolos en función de sus probabilidades de salida **en orden decreciente de confiabilidad**. Finalmente, evaluamos el desempeño del modelo teniendo en cuenta el *ranking* de compuestos elaborado anteriormente: se computa el valor de cada métrica a intervalos seleccionados de dicho ranking, donde cada intervalo comprende los  $k$  compuestos clasificados **con mayor confiabilidad** por el modelo.

la arquitectura de la DNN hasta calcular los puntajes correspondientes a los nodos de entrada, cada uno de ellos vinculado a un descriptor molecular.

En todos los casos consideramos los valores absolutos de los pesos, como una forma de analizar el impacto cuantitativo de los descriptores en la salida, independientemente de si contribuyen de manera directa o inversa al cómputo de la probabilidad de salida. Cabe destacar que los valores de descriptores moleculares de entrada son normalizados previamente, puesto que de otra forma descriptores con mayor variabilidad en su valor podrían producir un mayor impacto sin que ello

implique una verdadera incidencia mayor en el proceso de entrenamiento. Intuitivamente, pequeñas variaciones en los valores de aquellos descriptores moleculares con puntuaciones más altas tienen un impacto significativo en la salida de la DNN, por lo que son más influyentes o relevantes en el proceso de entrenamiento y, por ende, para la propiedad predicha por el modelo QSAR.

## 4.6. Resultados obtenidos

En esta sección del presente capítulo describimos la metodología empleada para evaluar nuestra propuesta integral, y brindamos detalles sobre los resultados experimentales obtenidos. A fin de validar nuestra propuesta, realizamos una evaluación de cada uno de nuestros modelos en comparación con los modelos del estado del arte previos a nuestro desarrollo para los tres casos de estudio, denominados en sus respectivos trabajos de referencia *Consenso 1* y *Consenso 2* [279, 241].

Teniendo en cuenta que entrenamos quince réplicas diferentes de cada modelo QSAR, cada una con una semilla aleatoria diferente para la inicialización de los parámetros entrenables de la red, reportamos tanto el desempeño promedio de los modelos, es decir, tomando en cuenta todos los ensayos, como el desempeño del mejor modelo en términos de *NER*, es decir, el modelo obtenido a partir de la mejor inicialización aleatoria de parámetros entrenables de la red neuronal. En cada caso, reportamos tanto el *NER* como el porcentaje de compuestos fuera del dominio de aplicabilidad, denominados *compuestos no asignados* (*%na*) siguiendo la nomenclatura de Nembri et al. [279] y de Mansouri et al. [241]. A efectos de una comparación justa con el estado del arte, reportamos tanto el rendimiento de nuestros modelos en términos de *NER* fijando el *%na* al porcentaje del mejor método reportado en los trabajos de referencia, como también el *%na* de nuestros modelos al alcanzar un desempeño en términos de *NER* igual al reportado en los métodos de referencia [279, 241].

En las tablas 4.1, 4.2 y 4.3 presentamos los resultados de nuestros modelos en sus versiones *Original* y *Enriquecida* de CYP2C9, CYP3A4 y RB, respectivamente. También incluimos los resultados de los modelos del estado del arte *Consenso 1* y *Consenso 2*, reportados por Nembri et al. [279] y Mansouri et al. [241].

CYP2C9		Validación interna				Validación externa			
		NER	Sn	Sp	Acc	NER	Sn	Sp	Acc
Original	Consenso 1	0,89	0,89	0,88	-	0,83	0,85	0,82	-
	Average_O	0,91	0,90	0,92	0,91	0,83	0,79	<b>0,88</b>	0,86
	Best_O	0,92	0,92	0,92	0,92	0,85	0,82	0,87	0,86
Enriquecido	Average_E	0,92	0,93	0,91	0,92	0,85	0,87	0,83	0,84
	<b>Best_E</b>	<b>0,93</b>	<b>0,94</b>	<b>0,93</b>	<b>0,93</b>	<b>0,87</b>	<b>0,89</b>	0,86	<b>0,87</b>

Tabla 4.1: Resultados en las particiones de Validación interna y externa de CYP2C9. Consistentemente con los resultados informados por Nembri et al. [279], el porcentaje de *compuestos no asignados* (%na) se fijó en 40 % para la Validación interna y 45 % para la Validación externa. Los valores marcados en **negrita** corresponden a los mejores resultados obtenidos en cada métrica.

CYP3A4		Validación interna				Validación externa			
		NER	Sn	Sp	Acc	NER	Sn	Sp	Acc
Original	Consenso 1	0,88	<b>0,92</b>	0,83	-	0,80	<b>0,89</b>	0,70	-
	Average_O	0,89	0,83	<b>0,94</b>	0,91	0,82	0,76	<b>0,88</b>	0,83
	Best_O	0,91	0,91	0,91	0,91	0,83	0,84	0,82	0,83
Enriquecido	Average_E	0,92	0,89	0,94	0,92	0,84	0,84	0,85	0,84
	<b>Best_E</b>	<b>0,93</b>	0,91	<b>0,94</b>	<b>0,93</b>	<b>0,85</b>	0,86	0,84	<b>0,85</b>

Tabla 4.2: Resultados en las particiones de Validación interna y externa de CYP3A4. Consistentemente con los resultados informados por Nembri et al. [279], el porcentaje de *compuestos no asignados* (%na) se fijó en 36 % para la Validación interna y 42 % para la Validación externa. Los valores marcados en **negrita** corresponden a los mejores resultados obtenidos en cada métrica.

Ready Biodegradability (RB)		Validación interna				Validación externa			
		NER	Sn	Sp	Acc	NER	Sn	Sp	Acc
Original	Consenso 2	0,91	0,88	<b>0,94</b>	-	0,87	0,81	<b>0,94</b>	-
	Average_O	0,92	0,94	0,90	0,91	0,88	<b>0,85</b>	0,91	0,90
	Best_O	0,91	0,91	0,91	0,91	0,88	0,85	0,92	0,90
Enriquecido	Average_E	<b>0,94</b>	0,93	0,90	<b>0,94</b>	0,88	0,83	0,93	0,90
	<b>Best_E</b>	<b>0,94</b>	<b>0,95</b>	0,92	0,93	<b>0,89</b>	<b>0,85</b>	0,93	<b>0,91</b>

Tabla 4.3: Resultados en las particiones de Validación interna y externa de RB. Consistentemente con los resultados informados por Mansouri et al. [241], el porcentaje de *compuestos no asignados* ( $\%na$ ) se fijó en 15% para la Validación interna y 13% para la Validación externa. Los valores marcados en **negrita** corresponden a los mejores resultados obtenidos en cada métrica.

En las tablas 4.1, 4.2 y 4.3 se puede observar que tanto en la Validación interna como en la Validación externa, nuestros modelos QSAR enriquecidos (*Average\_E* y *Best\_E*) exhiben similar o mejor desempeño en términos de *NER* que los modelos entrenados con el conjunto de descriptores moleculares Originales. En todos los casos, *Best\_E* supera a los modelos del estado del arte al mantener  $\%na$  al mismo valor que el reportado en los correspondientes trabajos de referencia [279, 241].

Como se muestra en las tablas 4.1 y 4.2, los modelos que desarrollamos para los casos de estudio CYP2C9 y CYP3A4 superan los resultados informados por Nembri et al. [279]. En la partición de Validación interna de ambos casos de estudio, el desempeño promedio en términos de *Sn* y *Sp* tanto en la versión original como en la versión enriquecida de los conjuntos de datos es más alto que el logrado por el modelo *Consenso 1* [279]. A su vez, dichos valores son equilibrados, lo que indica que nuestros modelos han superado con éxito el desequilibrio de clases inherente a los conjuntos de datos, prediciendo correctamente los compuestos activos e inactivos con una precisión similar.

Al evaluar nuestros modelos en la partición de Validación externa de CYP2C9, nuestros modelos también mejoran el rendimiento de los resultados de referencia. Sin embargo, se observa un leve desequilibrio entre *Sn* y *Sp*, lo que es consistente con los resultados informados por Nembri et al. [279]. En el caso de CYP3A4, los resultados obtenidos por nuestro modelo *Best\_E* en dicha partición son balanceados y significativamente mejores a *Consenso 1* en términos de *Sp*, lo que indica que el modelo no ha sido sujeto a sobreajuste para clasificar correctamente a los compuestos activos, que constituyen la clase minoritaria. La mejor réplica en la versión enriquecida del conjunto de datos

CYP2C9, es decir, *Best\_E*, obtuvo una mejora de 0,04 en términos de *NER* con respecto al mismo resultado para el modelo *Consenso 1*, mientras que para CYP3A4 se obtuvo una mejora de 0,05.

Con respecto al caso de estudio RB, en la tabla 4.3 podemos ver que el rendimiento de nuestros modelos supera los resultados del estado del arte, obtenidos por el modelo *Consenso 2* [241]. El *NER* promedio de nuestros modelos en la partición de Validación interna es más alto que el del *Consenso 2* en las versiones Original y Enriquecida del conjunto de datos y, al mismo tiempo, muestra valores promedio de *Sn* y *Sp* más equilibrados. Similares resultados se observaron en la partición de Validación externa, aunque el rendimiento de los modelos en términos de *Sn* y *Sp* exhibe un mayor desbalance. Este fenómeno se observa también para el modelo *Consenso 2*. No obstante, nuestro mejor modelo entrenado con la versión enriquecida del conjunto de datos RB, es decir, *Best\_E*, alcanzó el rendimiento más alto en términos de *NER* en ambas particiones de Validación.

A partir de los resultados observados en las particiones de Validación externa, concluimos que nuestros modelos poseen capacidad de generalización y que logran lidiar con el desbalance de clase de forma efectiva. En los tres casos de estudio puede observarse que los modelos entrenados a partir de las versiones enriquecidas exhiben mayor rendimiento predictivo y valores de *Sn* y *Sp* más equilibrados que aquellos modelos entrenados con las versiones originales, que comprenden solo aquellos descriptores elegidos mediante un proceso de selección de características. Este hallazgo sugiere que las versiones enriquecidas de los conjuntos de datos podrían contener información relevante para el proceso de entrenamiento, expresada a través de descriptores moleculares que no estaban presentes en las versiones originales de dichos conjuntos de datos.

Nuestros resultados experimentales demuestran que los modelos QSAR basados en DNN son aptos para escenarios de alta dimensionalidad y que posibilitan la omisión de un proceso de selección de características, el cual a su vez podría conducir a la pérdida de información valiosa y, por lo tanto, a una disminución en el rendimiento predictivo del modelo QSAR resultante [222]. Además, nuestro trabajo demuestra que los modelos QSAR basados en DNN son capaces de superar los modelos basados en consenso de modelos base, constituyendo una técnica sólida para modelado QSAR.

#### 4.6.1. Resultados de la técnica de dominio de aplicabilidad integrada

Como parte de nuestro enfoque integral, propusimos un modelo de dominio de aplicabilidad integrado basado en las probabilidades de salida de los modelos QSAR, dadas por la capa de salida de las DNNs. Este enfoque se aplicó en cada modelo QSAR y se evaluó en los conjuntos de Validación

interna y externa por medio de la medición de la correlación entre clasificaciones incorrectas y la confianza de la predicción de los modelos.

En las figuras 4.3 a 4.5 se presenta una evaluación completa del rendimiento de nuestros modelos en las particiones de Validación externa de los tres conjuntos de datos. Dichas figuras muestran el desempeño de los modelos entrenados sobre los conjuntos de datos enriquecidos en términos de las cuatro métricas de rendimiento antes presentadas para distintos umbrales de definición del dominio de aplicabilidad. Siguiendo la metodología detallada en la sección 4.5.3 del presente capítulo, la cual se resume de forma gráfica en la figura 4.2, el eje horizontal representa el número  $k$  de compuestos comprendidos en el intervalo, ordenados en orden decreciente de confiabilidad a partir de su probabilidad de salida, de modo que los compuestos que se encuentran más a la izquierda en el eje horizontal son aquellos para los cuales la DNN arrojó una predicción con mayor grado de confiabilidad.

La definición de un valor  $k$  en el eje horizontal de las figuras actúa como umbral para evaluar el rendimiento de los modelos, siendo los compuestos restantes aquellos fuera del dominio de aplicabilidad del modelo o *no asignados* ( $\%na$ ). Si bien es posible establecer cualquier umbral  $k$ , en particular marcamos dos umbrales: uno de ellos corresponde al porcentaje de compuestos no asignados  $\%na$  y el otro al rendimiento en términos de  $NER$  coincidentes con los reportados por Nembri et al. [279] y Mansouri et al. [241]. En el eje vertical se representan las diferentes métricas de desempeño, evaluadas sobre el conjunto según la Ecuación 4.1.

Según se puede observar en las figuras 4.3 a 4.5, a medida que aumenta el número de compuestos considerados dentro del dominio de aplicabilidad, es decir, hacia la derecha en el eje horizontal, los valores de las métricas tienden a decaer de forma continua, lo que implica que el desempeño del modelo está, de hecho, correlacionado con nuestra definición de dominio de aplicabilidad. Sin embargo, cabe señalar que estas curvas no son suaves en todos los casos de estudio. Para los compuestos de mayor confiabilidad, es decir, más a la izquierda en el eje horizontal, se observan algunos picos en las curvas, los cuales son causados por compuestos que son clasificados incorrectamente por el modelo con muy alta probabilidad de salida. Esto ocurre, por ejemplo, en la figura 4.3 *a* y *c*, en las curvas asociadas a  $NER$  y  $Acc$ .

En el caso de estudio CYP3A4, ilustrado por la figura 4.4, nuevamente se evidencia la efectividad de nuestro enfoque dominio de aplicabilidad integrado. A partir del análisis del rendimiento promedio en las subfiguras (a) y (b), la curva asociada a  $S_n$  exhibe variaciones en su pendiente, mientras que la curva para  $S_p$  es suave. Dichas fluctuaciones en la pendiente de  $S_n$  para  $Average_E$  pueden explicarse

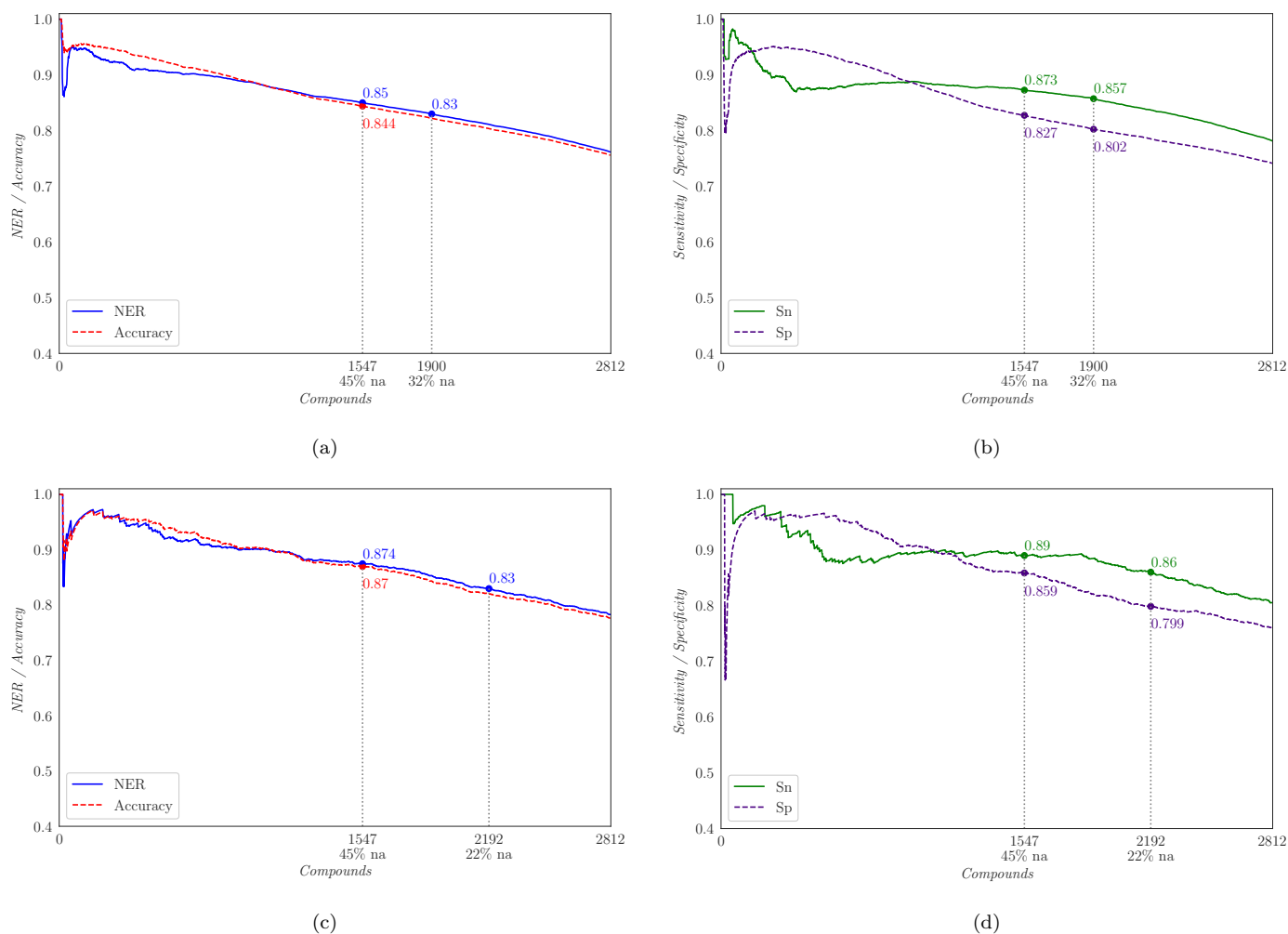


Figura 4.3: Desempeño de los modelos QSAR en la partición de Validación externa del conjunto de datos CYP2C9 enriquecido. (a) *NER* y *Accuracy* (*Acc*) para *Average\_E* (b) *Sn* y *Sp* para *Average\_E*. (c) *NER* y *Accuracy* (*Acc*) para *Best\_E* (d) *Sn* y *Sp* para *Best\_E*.

por algunas réplicas del modelo cuyo rendimiento es ligeramente más bajo que el resto en términos de *Sn*.

Con respecto al caso de estudio RB (figura 4.5), se observan picos abruptos en *Sn* tanto en las subfiguras para los modelos *Average\_E* como en *Best\_E*. Estas fluctuaciones indican que el modelo no siempre es capaz de predecir correctamente los compuestos activos con alta probabilidad de salida. En el caso de *Best\_E*, considerando que su desempeño en la partición de Validación interna es superior, este fenómeno podría estar relacionado con problemas de generalización. Más aún, y como se discutió anteriormente, un fenómeno similar se observa a través de los resultados desbalanceados en términos de *Sn* y *Sp* para el modelo *Consenso 2* [241], según se puede observar en la tabla 4.3.



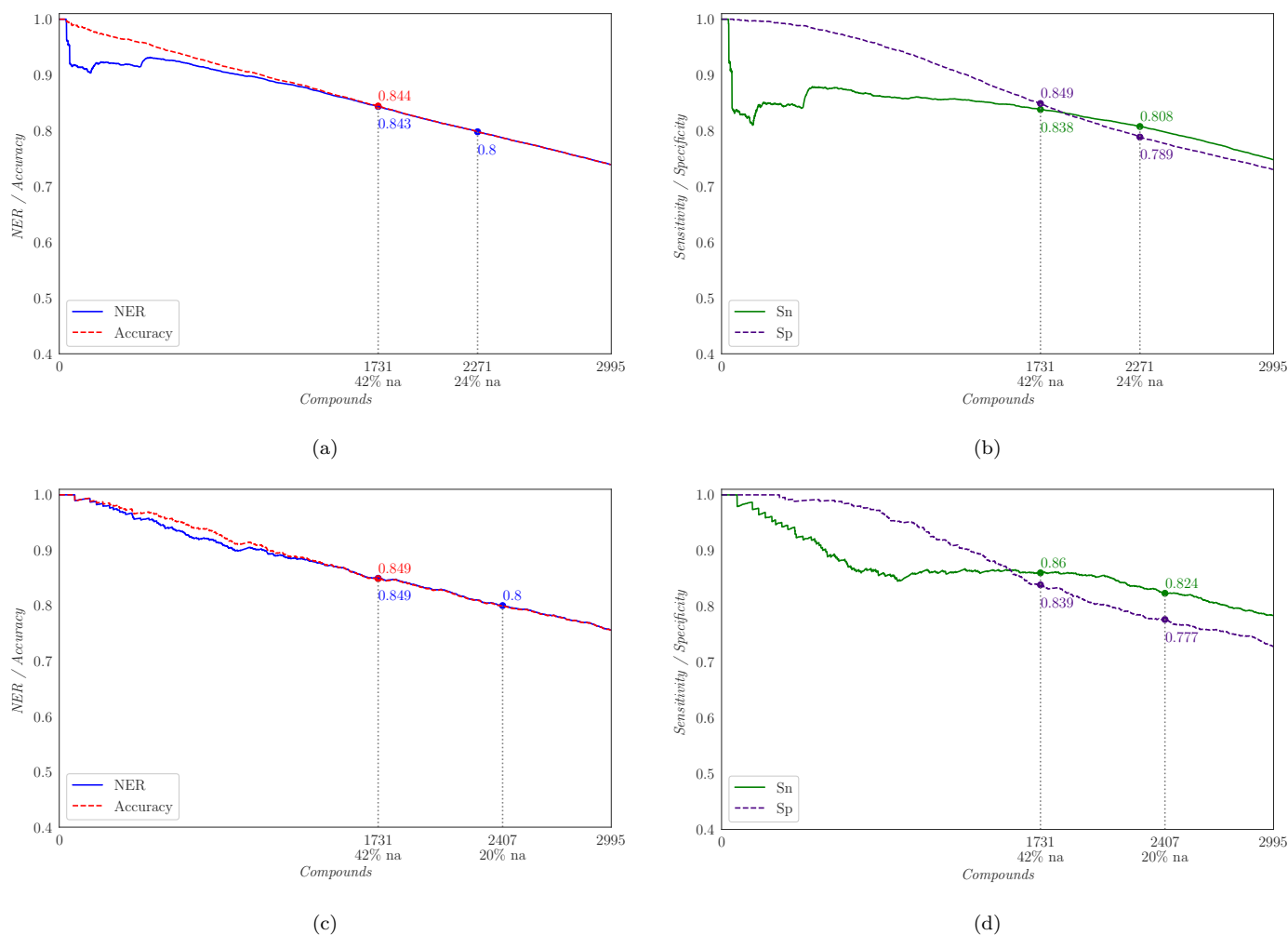


Figura 4.4: Desempeño de los modelos QSAR en la partición de Validación externa del conjunto de datos CYP3A4 enriquecido. (a) *NER* y *Accuracy* (*Acc*) para *Average\_E* (b) *Sn* y *Sp* para *Average\_E*. (c) *NER* y *Accuracy* (*Acc*) para *Best\_E* (d) *Sn* y *Sp* para *Best\_E*.

Este problema de generalización podría deberse a que los compuestos de la partición de Validación externa sean significativamente diferentes a los compuestos de las particiones de Entrenamiento y Validación interna. Para evaluar esta hipótesis, analizamos la similitud entre los compuestos de las dos particiones de Validación (interna y externa) con respecto a la partición de Entrenamiento para los tres conjuntos de datos utilizando dos métricas de distancia: euclídea estandarizada y coseno. Cada distancia fue a su vez computada por dos métodos de enlace; por un lado, empleando *enlace simple* (*single linkage*), también conocido como *vecino más cercano*, que es la distancia más corta entre un par de instancias pertenecientes a dos agrupamientos diferentes (en nuestro caso, dos particiones de datos a comparar); por otro lado, empleando *enlace promedio* (*average linkage*), en el que las

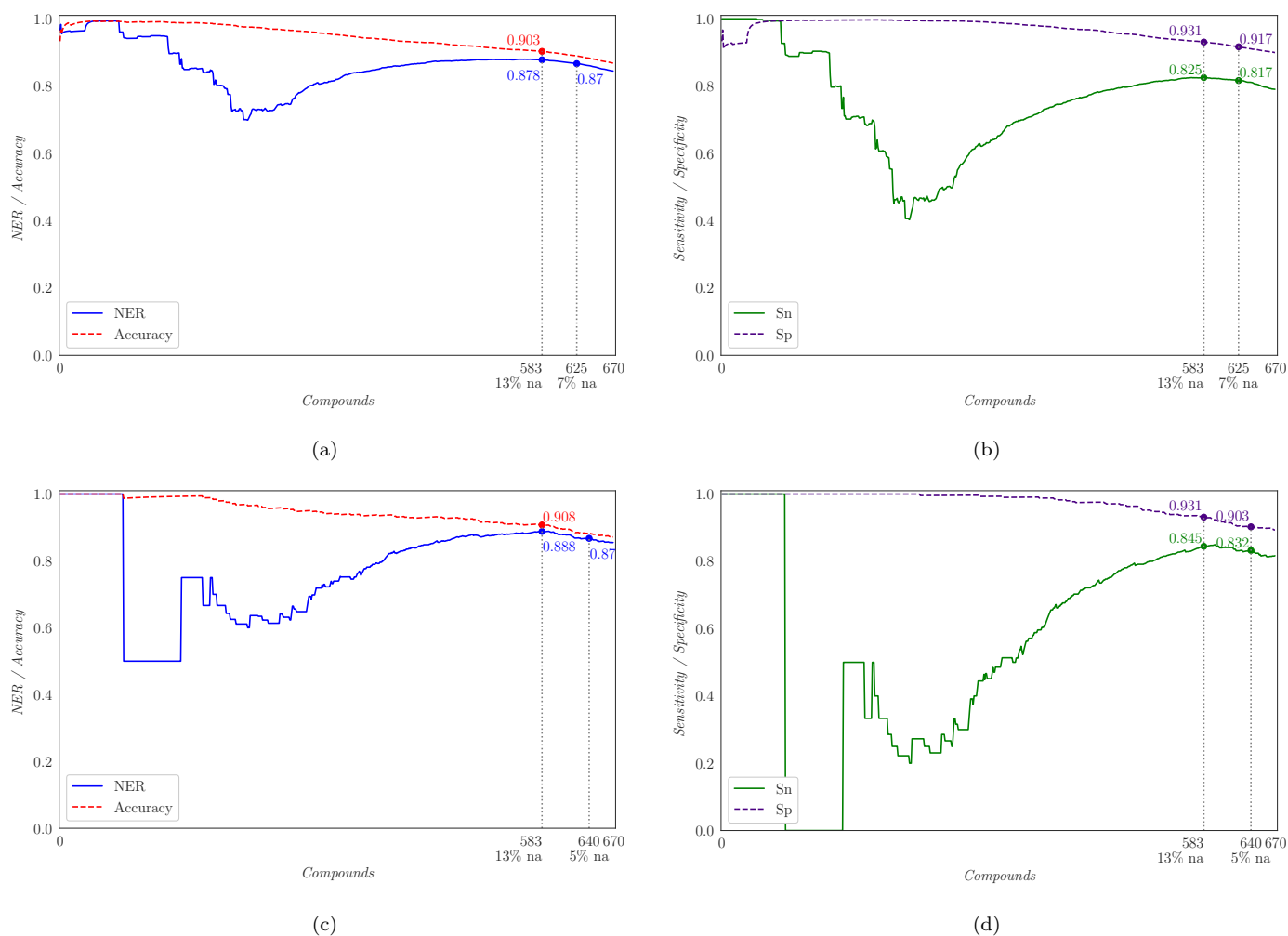


Figura 4.5: Desempeño de los modelos QSAR en la partición de Validación externa del conjunto de datos RB enriquecido. (a) *NER* y *Accuracy* (*Acc*) para *Average\_E* (b) *Sn* y *Sp* para *Average\_E*. (c) *NER* y *Accuracy* (*Acc*) para *Best\_E* (d) *Sn* y *Sp* para *Best\_E*.

distancias entre cada par de compuestos en los dos agrupamientos (particiones de datos comparadas) se suma y luego se divide por el total de compuestos—es decir, la media aritmética de todas las distancias. Ambas métricas de distancia se explican en mayor detalle en el capítulo 2.

Luego de computar las distancias entre cada partición de validación y la partición de entrenamiento, procedimos a medir la diferencia relativa entre las mismas. Los resultados de dicho análisis se presentan en las tablas 4.4 y 4.5. De forma sintética, valores de diferencia mayores indican una mayor distancia entre las particiones de Validación externa y Entrenamiento que la que hay entre las particiones de Validación interna y Entrenamiento. Los resultados hallados muestran que, de hecho, existe una diferencia significativa entre los compuestos en la partición de Validación externa del

conjunto de datos RB y los del conjunto de Validación interna, al medir sus respectivas distancias a la partición de Entrenamiento. Estas distancias promedio son, a su vez, significativamente mayores que las observadas para los conjuntos de datos CYP2C9 y CYP3A4. La diferencia entre estos conjuntos de compuestos podría explicar los problemas de generalización de *Best\_E* en el caso de estudio RB. Asimismo, este problema de generalización debido a las diferencias en la distribución de datos se exagera a medida que aumenta la dimensionalidad de los datos, por lo que los modelos creados a partir de la versión enriquecida pueden verse razonablemente desfavorecidos en comparación con los modelos de la versión original del conjunto de datos.

Distancia euclídea estandarizada		
Conjunto de datos	Enlace simple	Enlace promedio
RB Original	20,36 %	1,62 %
RB Enriquecido	28,54 %	5,91 %
CYP3A4 Original	6,38 %	0,12 %
CYP3A4 Enriquecido	6,42 %	0,10 %
CYP2C9 Original	0,47 %	0,10 %
CYP2C9 Enriquecido	0,06 %	0,43 %

Tabla 4.4: Resumen del análisis de similitud entre las particiones de Validación con respecto a sus respectivas particiones de Entrenamiento empleando distancia euclídea estandarizada. El valor reportado representa la diferencia relativa entre las distancias computadas entre la partición de Entrenamiento y las particiones de Validación interna y externa. Las distancias fueron computadas por enlace simple (menor distancia entre dos compuestos de cada partición) y enlace promedio (promedio de distancias entre todos los pares de compuestos de ambas particiones).

Por último, a partir de las figuras 4.3 a 4.5 se observa que nuestros modelos alcanzaron el rendimiento de los modelos reportados por Nembri et al. [279] y Mansouri et al. [241] en términos de *NER* para los tres casos de estudio, pero con un porcentaje de *compuestos no asignados* (*%na*) mucho menor. Para CYP2C9, se informa un valor de *NER* de 0,83 con 45% *na* para *Consenso 1*, mientras que *Best\_E* alcanza el mismo valor de *NER* descartando solo el 22% de los compuestos en la Validación externa. En el caso de CYP3A4, se reporta un valor de *NER* de 0,80 con 42% *na* para *Consenso 1*, mientras que *Best\_E* logra el mismo valor de *NER* con solo 20% *na*. Por último, para el caso de estudio RB, Mansouri et al. [241] reporta un *NER* de 0,87 con 13% *na* para *Consenso 2*, mientras que *Best\_E* alcanza el mismo valor de *NER* con solo 5% *na*. Un análisis similar podría realizarse tomando en consideración el *%na* informado por Nembri et al. [279] y Mansouri et al.

Conjunto de datos	Distancia coseno	
	Enlace simple	Enlace promedio
RB Original	23,58 %	0,00 %
RB Enriquecido	29,81 %	0,17 %
CYP3A4 Original	12,17 %	0,03 %
CYP3A4 Enriquecido	12,09 %	0,03 %
CYP2C9 Original	0,33 %	0,03 %
CYP2C9 Enriquecido	1,69 %	0,02 %

Tabla 4.5: Resumen del análisis de similitud entre las particiones de Validación con respecto a sus respectivas particiones de Entrenamiento empleando similitud por coseno. El valor reportado representa la diferencia relativa entre las distancias computadas entre la partición de Entrenamiento y las particiones de Validación interna y externa. Las distancias fueron computadas por enlace simple (menor distancia entre dos compuestos de cada partición) y enlace promedio (promedio de distancias entre todos los pares de compuestos de ambas particiones).

[241] para los tres conjuntos de datos, ya que nuestros modelos exhibieron desempeños en términos de *NER* sistemáticamente mejores para la misma cantidad de compuestos descartados que los modelos *Consenso 1* y *Consenso 2* en ambas particiones de Validación.

#### 4.6.2. Resultados del análisis de características moleculares *a posteriori*

Como parte de nuestra propuesta integral, realizamos un análisis de características *a posteriori* para obtener información sobre qué descriptores moleculares resultaron los más influyentes en el proceso de aprendizaje de nuestros modelos. Presentamos los resultados de este análisis en la figura 4.6, que consiste en tres mapas de calor [117]. En dichos mapas, cada fila corresponde a una de las quince réplicas de nuestros modelos enriquecidos, cada una de ellas empleando diferentes inicializaciones aleatorias de los parámetros entrenables de la DNN. Las réplicas se encuentran ordenadas por rendimiento en términos de *NER*, donde la fila superior representa la mejor prueba *Best\_E*. Cada columna de los mapas representa descriptores moleculares, e incluimos únicamente los veinte descriptores más relevantes a la salida de cada réplica, computando su relevancia por medio de la Ecuación 4.2. Aquellos descriptores marcados con un asterisco (\*) forman parte tanto de la versión original como de la versión enriquecida del conjunto de datos. Aquellos descriptores cuyo nombre comienza con *EFCP* representan bits de los *fingerprint EFCP* asociados a los compuestos,

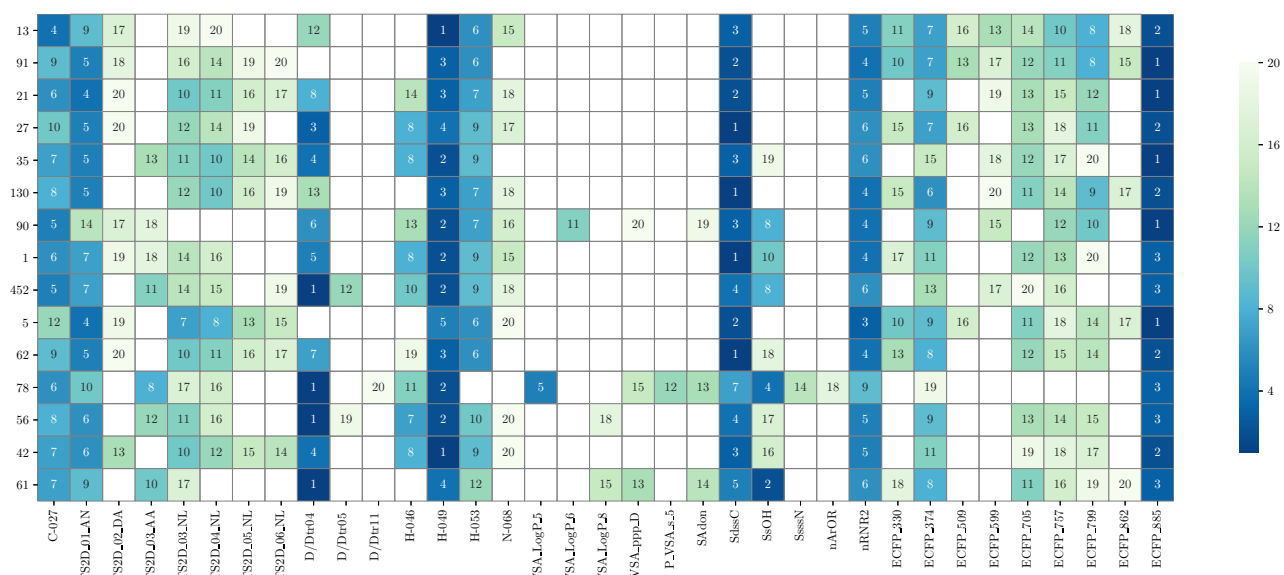
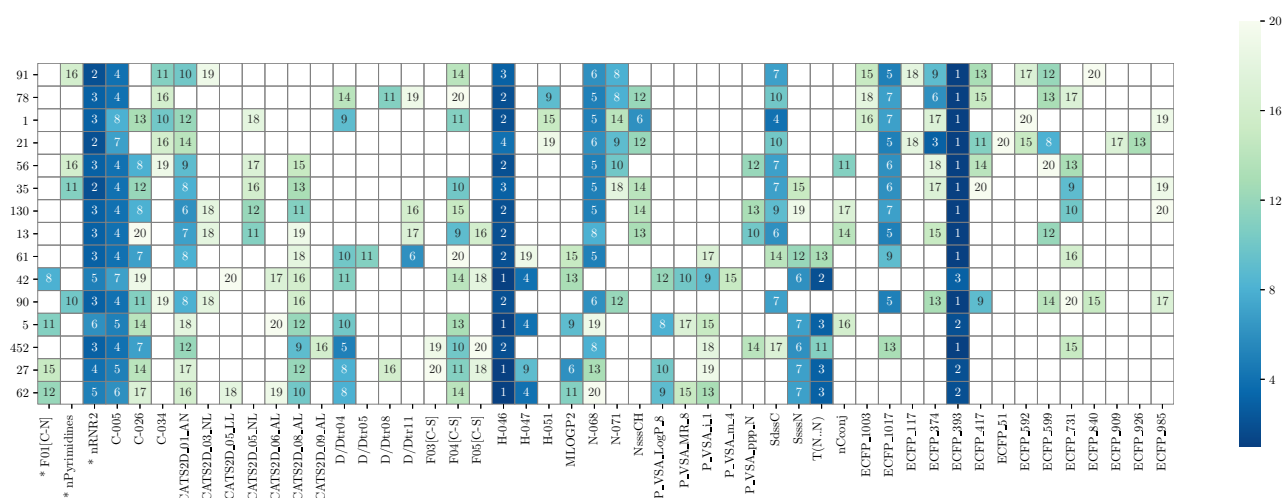
seguidos de un número que representa la posición del bit en el vector de 1.024 bits que nuestro análisis identificó como relevante. El orden de relevancia de un determinado descriptor para una cierta réplica se codifica de dos maneras: por medio del número dentro de la celda del mapa de calor correspondiente, y por medio del color de dicha celda, donde los colores más oscuros corresponden a las celdas que representan descriptores de mayor relevancia para una réplica en específico, mientras que los colores más claros codifican descriptores menos relevantes. Las celdas en blanco—es decir, sin colorear y sin número—indican descriptores moleculares que no fueron identificados dentro de los veinte más relevantes para la réplica en cuestión.

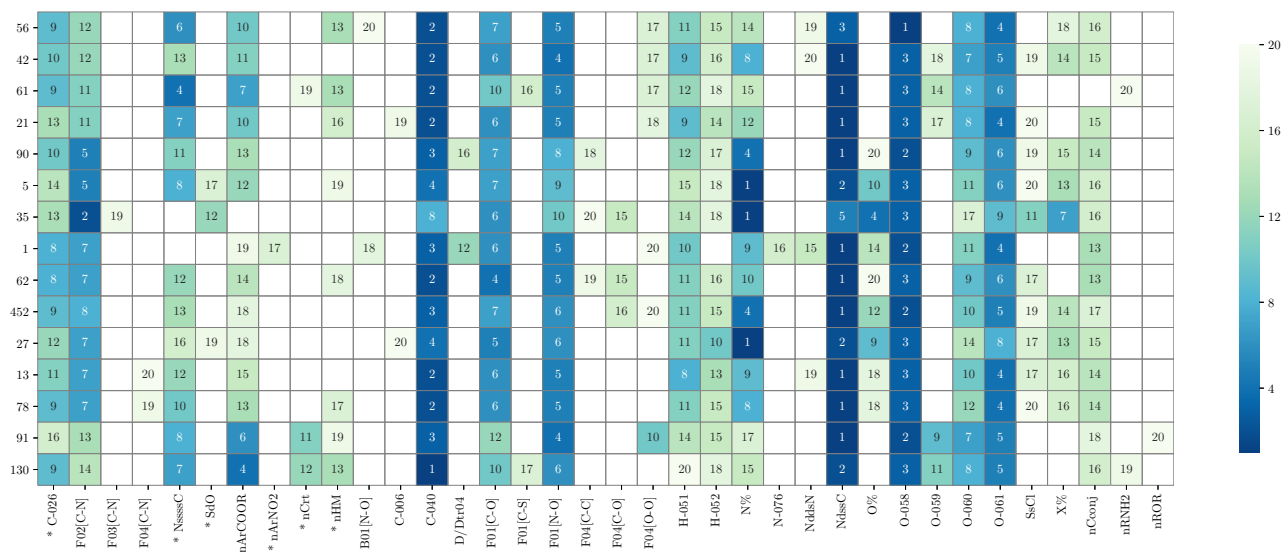
A partir de los mapas de calor en la figura 4.6, es posible identificar descriptores moleculares que fueron relevantes en los procesos de entrenamiento de todas las réplicas para un mismo caso de estudio. Por ejemplo, para CYP2C9, el bit *ECFP\_393* resultó la característica molecular más relevante para once de las quince réplicas. Los descriptores moleculares *H-046* y *nRNR2* también se seleccionaron con frecuencia en las distintas réplicas. En el caso de CYP3A4, el bit *ECFP\_885* se consideró entre los tres descriptores más relevantes en todas las réplicas, junto con los descriptores *SdssC* y *H-049* también se señalaron como relevantes para la mayoría de los ensayos, según nuestra medida. Para el conjunto de datos RB, el descriptor molecular *NdssC* fue identificado como el más relevante para la mayoría de los modelos, ya que se consideró el descriptor más relevante para once de los quince ensayos, seguido por los descriptores *O-058* y *C-040*.

Otro aspecto interesante que se puede observar en estos mapas de calor es que las réplicas que alcanzaron rendimientos similares tendieron a priorizar los mismos descriptores moleculares durante el proceso de entrenamiento, incluso arrojando un orden de relevancia similar. Finalmente, entre los descriptores moleculares identificados como relevantes para cada modelo por nuestra técnica, hay algunos descriptores que también se encuentran en las versiones originales de los conjuntos de datos. El mayor número de descriptores compartidos entre estos dos conjuntos se observa para el caso de estudio RB, donde 10 de los 36 descriptores señalados como los más influyentes también están presentes en el conjunto de datos original. Cabe destacar que los modelos de referencia *Consenso 1* desarrollados para los casos de estudio CYP2C9 y CYP3A4 por Nembri et al. [279] emplean un *fingerprint ECFP* como entrada para uno de los modelos base de su ensamble, por lo que en nuestro análisis consideramos que todos los bits ECFP están presentes en la versión original de estos dos conjuntos de datos.

Nuestro enfoque integral superó a los modelos de referencia reportados por Nembri et al. [279] y Mansouri et al. [241], permitiendo al mismo tiempo la identificación de descriptores moleculares relevantes no incluidos en los conjuntos de datos originales, lo que nos permite inferir que dichos

descriptores moleculares potencialmente codifican información relevante para el modelado de los casos de estudio. Mediante nuestro método de análisis de características *a posteriori* fue posible identificar descriptores moleculares relevantes para nuestros modelos y encontrar relaciones interesantes entre ellos, lo que contribuye a la interpretabilidad de los resultados de la predicción de nuestros modelos QSAR. Más allá de que la lectura de estos resultados resulta directa en el caso de los descriptores moleculares, la utilidad de este enfoque en el caso de los bits asociados a los *fingerprints ECFP* —





(c) RB

Figura 4.6: Análisis de características *a posteriori* sobre las quince réplicas entrenadas para los tres conjuntos de datos enriquecidos. Las filas representan diferentes réplicas de nuestro modelo ordenadas por rendimiento, identificadas por la semilla empleada para la inicialización de sus parámetros entrenables, y las columnas representan los veinte descriptores moleculares más relevantes. Los colores de celda más oscuros corresponden a las celdas que representan descriptores de mayor relevancia para una réplica en específico, mientras que los colores más claros codifican descriptores menos relevantes. El número dentro de cada celda identifica el orden de relevancia de dicho descriptor para la réplica correspondiente.

donde un mismo bit puede estar asociado a más de un patrón estructural de las moléculas analizadas— radica en la posibilidad de realizar un análisis de los resultados asociados al conjunto de datos, verificando para los compuestos estudiados qué patrones estructurales coinciden entre compuestos con similares valores en los bits identificados como características relevantes.

## 4.7. Síntesis y conclusiones

El modelado QSAR se ha convertido en una etapa clave en el complejo proceso de descubrimiento de fármacos a lo largo de los años. Si bien los modelos basados en DNN logran un rendimiento predictivo más alto que otras técnicas establecidas, tienen sus propios desafíos, como la dificultad en

la interpretabilidad de sus resultados y su propensión al sobreajuste. En este capítulo, se presentó el desarrollo de un enfoque integral de modelado QSAR basado en redes neuronales profundas (DNNs) [322], por medio del cual logramos resultados superiores al estado del arte para tres casos de estudio de relevancia en el dominio biomédico sin la necesidad de realizar un proceso de selección de características moleculares. Además, planteamos una estrategia eficaz para analizar el dominio de aplicabilidad de un modelo QSAR neuronal basado en las probabilidades de salida de la red, y proporcionamos una técnica basada en una agregación *a posteriori* de los parámetros entrenables de la DNN para identificar los descriptores moleculares y características más relevantes al proceso de modelado, lo que brinda un sentido de interpretabilidad a nuestros modelos.

Entre los casos de estudio abordados en nuestra propuesta, desarrollamos modelos para la predicción de Biodegradabilidad Simple (*RB*) y para predecir la interacción entre compuestos químicos y los *citocromos P450* en las isoformas CYP2C9 y CYP3A4. En particular, los *citocromos P450* están implicados en la oxidación de compuestos, siendo estas dos isoformas particularmente relevantes para el metabolismo de fármacos. Estudios científicos han demostrado que, si bien dichas isoformas son dianas biológicas diferentes, existe interacción y una fuerte correlación entre los perfiles de actividad biológica de ambas. En el proceso de desarrollo de fármacos es usual que distintas propiedades físico-químicas o biológicas estén correlacionadas o sean complementarias. Considerando la flexibilidad que caracteriza a las arquitecturas DNN, resulta interesante inspeccionar la posibilidad de modelar múltiples dianas biológicas por medio de un mismo modelo. En el próximo capítulo, exploramos el desarrollo de modelos basados en DNNs empleando aprendizaje multi-tarea como una estrategia de entrenamiento que permite modelar múltiples propiedades biológicas en simultáneo, aprovechando su complementariedad y mejorando el rendimiento predictivo de los modelos.



# Capítulo 5

## Modelado QSAR empleando aprendizaje multi-tarea (MTL)

El desarrollo de fármacos requiere de la identificación de compuestos candidatos que exhiban propiedades físico-químicas y perfiles de bioactividad específicos. Por lo general, el perfil de bioactividad de un medicamento involucra a distintos receptores o blancos farmacológicos que interactúan de formas complejas en los organismos vivos. En este capítulo, exploramos una propuesta basada en el modelado QSAR multi-tarea basado en DNNs, que permite el aprendizaje de patrones asociados a múltiples propiedades complementarias en simultáneo.

---

### 5.1. Introducción a la mutagenicidad de Ames

La mutagenicidad es la capacidad de ciertas sustancias de afectar la integridad del material genético de las células, por lo que a la hora de analizar la seguridad de fármacos candidatos y muestras industriales, químicas y ambientales, es requisito esencial analizar su potencial mutagénico. El continuo descubrimiento y desarrollo de nuevos compuestos químicos ha llevado a fortalecer las medidas regulatorias para asegurar el uso seguro de sustancias nuevas y existentes en términos de su mutagenicidad.

En este contexto, la prueba de Ames constituye la primera práctica estándar y el ensayo más utilizado para evaluar el riesgo mutagénico de nuevas sustancias y también con fines regulatorios previo al registro y aceptación de las mismas [12, 245]. La prueba de Ames es un modelo *in vitro* que emplea diferentes cepas de *Salmonella typhimurium* alteradas genéticamente en los genes involucrados

en la síntesis de *histidina* [270]. Dichas mutaciones genéticas causan que las bacterias requieran de un suministro externo de histidina para su replicación. La prueba de Ames evalúa la capacidad de las sustancias y compuestos químicos para revertir tales mutaciones, permitiendo el crecimiento bacteriano en entornos libres de histidina.

Las Directrices para el Ensayo de Productos Químicos emitidas por la Organización para la Cooperación Económica y Desarrollo (OECD) [284] indican que se debe utilizar al menos cinco cepas diferentes de *Salmonella typhimurium* para realizar una prueba de Ames, la cual determina que un compuesto o sustancia es mutagénica si el ensayo arroja resultado positivo al menos para una de dichas cepas. La razón de este requisito es que el uso de diferentes cepas contribuye a detectar distintos tipos de mutágenos. Por caso, para algunas de estas cepas, entre las cuales destacan *TA1535*, *TA1537*, *TA97*, *TA98* y *TA100*, se ha demostrado experimentalmente que en ocasiones no reaccionan ante ciertos mutágenos oxidantes, agentes de enlace e hidrazinas, los cuales sí son detectados por cepas de *Escherichia coli WP2* o *Salmonella typhimurium TA102* [12].

Si bien la prueba de Ames emplea resultados utilizando varias cepas distintas de *Salmonella typhimurium*, las cuales aportan información complementaria [12], la mayoría de los modelos QSAR desarrollados para predecir mutagenicidad no evalúan los resultados de los experimentos individuales realizados en cada una de dichas cepas. En su lugar, los modelos *in silico* para predicción de mutagenicidad son entrenados empleando etiquetas globales que categorizan el potencial mutagénico de los compuestos agrupando en un solo valor (*mutagénico* versus *no mutagénico*) los resultados de la multiplicidad de cepas comprendidas en el proceso experimental.

Recientemente, los modelos basados en redes neuronales profundas (*DNN*, por sus siglas en inglés) combinados con estrategias de Aprendizaje Multi-Tarea (*Multi-Task Learning* o *MTL*) han producido resultados interesantes en diferentes dominios [319], dadas sus capacidades para modelar múltiples objetivos en simultáneo. En este escenario, en este capítulo presentamos el desarrollo de un nuevo modelo QSAR basado en DNNs para predecir mutagenicidad que aprovecha los resultados experimentales de diferentes cepas de *Salmonella typhimurium* utilizadas en la prueba de Ames mediante un enfoque de aprendizaje multi-tarea. Nuestra estrategia de modelado superó en rendimiento a los modelos entrenados con el enfoque tradicional empleando etiquetas de clase globales y también a las estrategias de modelado basadas en consenso de modelos individuales entrenados a partir de distintas cepas.

## 5.2. Modelado QSAR de la prueba de mutagenicidad de Ames

El desarrollo de métodos *in silico* para predecir mutagenicidad por medio del modelado de la prueba de Ames es un campo activo de investigación en toxicología computacional [119, 163, 40, 155] y existen múltiples estudios de revisión de los modelos y herramientas de software más relevantes para predecir mutagenicidad sobre diferentes conjuntos de datos [59, 36, 162, 56, 377, 191]. Sin embargo, el impacto individual de las diferentes cepas utilizadas en la prueba de Ames en el diseño de métodos QSAR ha sido escasamente estudiado. Los modelos QSAR hallados en la literatura se entrenan empleando etiquetas globales (*mutagénico* versus *no mutagénico*) resultantes de la prueba de Ames, sin considerar los resultados intermedios de los experimentos realizados individualmente para cada cepa bacteriana.

Si bien las directrices de la OECD definen un conjunto mínimo de cepas que deben estar presentes en los experimentos *in vitro* de la prueba de Ames, en la práctica es común encontrar discrepancias en los conjuntos de cepas utilizados en estudios de toxicidad y en conjuntos de datos públicos [163, 192, 371]. Por ejemplo, Williams et al. [423] mostraron evidencia que apoya la hipótesis de que las cepas de *S. typhimurium* TA1535, TA1537, TA102 y la cepa *E. coli* WP2 *uvrA* podrían eliminarse del conjunto de cepas recomendado con poca o ninguna pérdida de sensibilidad para la detección de mutágenos bacterianos. Este estudio pone de manifiesto la falta de un consenso absoluto entre los expertos del dominio sobre el modo en que se debe realizar la prueba de Ames. Por lo tanto, resulta de interés analizar si es posible diseñar modelos *in silico* para predecir mutagenicidad teniendo en cuenta la contribución individual de cada una de las cepas involucradas en la prueba y su complementariedad. Además, surge como interrogante de investigación si tal análisis permitiría obtener resultados más precisos y modelos computacionales más interpretables para ensayos de toxicidad mutagénica que mediante el enfoque tradicional, que emplea etiquetas de mutagenicidad globales.

Tal y como hemos discutido en el capítulo 3, durante la última década las técnicas de aprendizaje profundo se han convertido en el estándar para una amplia variedad de tareas en el proceso de descubrimiento de fármacos [214, 17]. En particular, las redes neuronales profundas (DNNs) se encuentran hoy en día entre las técnicas más utilizadas para el modelado QSAR [118, 166, 435]. Dentro del amplio espectro de técnicas de modelado basadas en DNN, el Aprendizaje Multi-Tarea (MTL) [448] ha demostrado su potencial en múltiples dominios y es hoy en día explorado como una estrategia útil para modelar múltiples objetivos en simultáneo. La estrategia de entrenamiento MTL permite el desarrollo de modelos QSAR capaces de predecir el perfil de bioactividad de compuestos

candidatos con respecto a combinaciones arbitrarias de propiedades, tanto en tareas de regresión como de clasificación. Aunque la idea detrás de MTL no es exclusiva del aprendizaje profundo, los enfoques MTL aplicados a modelos basados en DNN permiten combinar la información de las diversas propiedades a predecir durante el proceso de entrenamiento del modelo, potenciando así su complementariedad a la vez que se explota la información de cada tarea predictiva individual.

En este capítulo presentamos el diseño y desarrollo de un modelo QSAR novedoso para predecir mutagenicidad, basándonos en los experimentos individuales con varias de las cepas más utilizadas en la prueba de Ames [248]. Nuestro modelo se basa en DNNs y sigue una estrategia de entrenamiento MTL, por medio de la cual cada cepa es tratada como una propiedad objetivo individual, a la vez que la información proporcionada por todas las cepas es aprendida en simultáneo por el modelo. Los resultados de nuestro modelo MTL son luego combinados mediante una estrategia de consenso para recrear la prueba de Ames. El objetivo de nuestra propuesta es modelar la prueba de mutagenicidad de Ames por medio de un enfoque MTL y analizar los potenciales beneficios de dicho enfoque con respecto a las estrategias de modelado tradicionales, donde únicamente se emplean etiquetas de clase globales. El principal desafío consiste en modelar efectivamente la contribución de las diferentes cepas a un resultado general de mutagenicidad de Ames, mientras que se enriquece el modelado del perfil de bioactividad de los compuestos para cada cepa individual por medio de un procedimiento de aprendizaje conjunto. Nuestra hipótesis motivadora consiste en que un enfoque MTL puede impulsar al modelo QSAR para aprovechar la complementariedad de las diferentes cepas a la vez explotando sus características predictivas específicas.

Como preguntas de investigación que motivaron nuestra propuesta, en primer lugar, analizamos si un modelo QSAR entrenado por medio de un enfoque MTL es capaz de superar el rendimiento de un modelo QSAR entrenado por medio de estrategias de aprendizaje tradicionales, es decir, empleando etiquetas de clase globales (enfoque de *tarea única*). Por medio de este análisis pretendemos dilucidar los beneficios de modelar la mutagenicidad de Ames utilizando información de cepas individuales en lugar de combinarlas en un único valor global, tal y como se realiza en la práctica estándar. En segundo lugar, examinamos el rendimiento del enfoque MTL en contraste con un modelo QSAR para mutagenicidad de Ames basado en consenso de cepas individuales después de un proceso de entrenamiento de tarea única. El propósito de dicho proceso experimental es determinar si el proceso de modelado para cada cepa individual se ve influenciado por el proceso de aprendizaje conjunto.

### 5.3. Trabajo relacionado con la propuesta

El desarrollo de modelos *in silico* para la predicción de mutagenicidad de Ames se ha convertido en un campo de investigación muy activo durante las últimas décadas [72]. Los modelos QSAR para predecir mutagenicidad de Ames se pueden clasificar en dos grandes grupos: modelos *basados en reglas* y modelos *estadísticos* [162]. Los modelos basados en reglas predicen cualitativamente criterios de valoración particulares, buscando coincidencias entre fragmentos moleculares específicos de compuestos nunca vistos por el modelo y ciertas alertas estructurales, es decir, estructuras similares con efectos adversos conocidos (por ejemplo, mutagenicidad). Estas reglas son elaboradas a partir de literatura científica y conocimiento experto, extraídas de grandes conjuntos de datos, o por medio de una combinación de ambos enfoques [155]. Los modelos basados en reglas generalmente son aplicables a problemas de predicción binarios, donde la tarea es determinar la presencia o ausencia de dichas alertas estructurales en un compuesto dado. Por otro lado, los modelos QSAR estadísticos predicen la toxicidad mediante el análisis de correlaciones estadísticas empleando representaciones vectoriales de compuestos, generalmente basadas en descriptores u otras características moleculares, y emplean técnicas de aprendizaje automático [153]. Aunque ambas metodologías tienen sus propias fortalezas y debilidades, los modelos estadísticos tienden a exhibir mejor rendimiento que los modelos basados en reglas, permitiendo además la predicción incluso en escenarios en los que se desconoce el mecanismo de acción biológico de la propiedad en estudio [155].

En la literatura se observan diversas estrategias de modelado QSAR desarrolladas bajo estos dos enfoques. Honma [162] presentó una revisión crítica sobre las herramientas QSAR más populares para predecir mutagenicidad, la mayoría de las cuales participaron en el proyecto *Ames/QSAR International Challenge Project*, donde doce proveedores internacionales de modelos QSAR desarrollaron y probaron diecisiete herramientas QSAR en tres fases realizadas entre 2014 y 2017. Los resultados finales de esta competencia fueron reportados por Honma et al. [163]: en términos de desempeño, la mayoría de las herramientas lograron un rendimiento en términos de *Sensibilidad* ( $S_n$ ) superior al 50% y alcanzaron valores de *Precisión* del 80%. En dicha competencia, solo una herramienta QSAR denominada *MUT\_Risk* [353] fue desarrollada teniendo en cuenta las contribuciones individuales de diferentes cepas de *S. typhimurium* para predecir toxicidad. Según Honma [162], *MUT\_Risk* utiliza diez modelos creados a partir de datos de cinco cepas individuales de *S. typhimurium* o *E. coli* (cepas 98, 100, 97+1.537, 1.535, 102+wp2), con y sin activación metabólica S9 de hígado de rata. En *MUT\_Risk*, para cada compuesto clasificado como mutagénico por cada uno de los cinco pares de modelos  $\pm S9$  (es decir, con y sin activación metabólica), se suma un punto al puntaje total arrojado por el modelo, donde  $lx$  *usuarix* establece un puntaje umbral para

evaluar toxicidad positiva. En particular, se realizaron experimentos trabajando con dos umbrales de toxicidad durante la competencia: *MUT\_Risk-0* clasifica un compuesto como mutagénico cuando la puntuación arrojada por el modelo es mayor que 0, mientras que *MUT\_Risk-1* clasifica a un compuesto como mutagénicos cuando la puntuación es superior a 1. La flexibilidad de dichos umbrales permitió al modelo alcanzar un desempeño predictivo equilibrado en términos de *Sensibilidad (Sn)* y *Especificidad (Sp)*. Sin embargo, el rendimiento de este enfoque en términos de *Exactitud Balanceada* y del *coeficiente de correlación de Matthews (MCC)* fue bajo en comparación con las otras herramientas QSAR participantes en la competencia [163].

Más allá de las herramientas QSAR presentadas en *Ames/QSAR International Challenge Project*, en los últimos años se han publicado varios trabajos de investigación sobre modelado QSAR de toxicidad empleando aprendizaje profundo [295]. Considerando la creciente disponibilidad de conjuntos de datos grandes y complejos, se han desarrollado diversos modelos utilizando un enfoque de aprendizaje MTL para predicción de toxicidad, los cuales resultan de particular interés para este capítulo. Tang et al. [367] presentaron una breve revisión que incluía diversos modelos QSAR basados en MTL entrenados para predecir diferentes tipos de toxicidad. Mayr et al. [253] desarrollaron un modelo de predicción de toxicidad basado en MTL utilizando un gran conjunto de datos del desafío *Tox21 2014* [168]. En este desafío, se probaron 12.707 compuestos químicos etiquetados para entre 1 y 12 tipos de toxicidad diferentes. En dicho trabajo, los autores compararon el desempeño de los modelos entrenados para una sola tarea de predicción y de modelos basados en DNNs multi-tarea, además de construir un modelo base para cada tarea basado en una *Support Vector Machine (SVM)* con kernel lineal. Los modelos multi-tarea lograron un mayor rendimiento que los modelos de tarea única y que los modelos basados en SVMs en 10 de las 12 tareas de predicción de toxicidad. Sin embargo, para el caso de un conjunto de datos particularmente desbalanceado, que incluía solo tres compuestos positivos, todos los modelos fallaron, exhibiendo las limitaciones de los modelos basados en aprendizaje profundo en escenarios de distribuciones de datos, exhibiendo fuertes desbalances de clase. Por otra parte, si bien esta propuesta de modelado fue desarrollada para diferentes tipos de toxicidad, cuyos determinantes químicos y estructurales son más fáciles de modelar *in silico* que aquellos de la mutagenicidad dada la amplia variedad de criterios de valoración químicos y datos disponibles [337], la misma pone de manifiesto la posibilidad de modelar exitosamente perfiles farmacológicos complejos por medio de una estrategia MTL.

Hughes et al. [170] propusieron un modelo de aprendizaje multi-tarea para predecir la reactividad de los compuestos químicos con *glutatión (GSH)*, *cianuros*, *ácido desoxirribonucleico (ADN)* y una serie de proteínas. La identificación de este tipo de reacciones químicas juega un papel central en

la detección de mecanismos subyacentes a diversos tipos de toxicidad inducida por fármacos. Según reportan en su trabajo, para construir su modelo predictivo lxs autorxs recolectaron 1.364 moléculas electrófilas reactivas con GSH, cianuros, ADN o proteínas y 1.439 moléculas no reactivas de la base de datos de metabolitos de *Accelrys (AMD)* [83], y emplearon más de 200 descriptores moleculares topológicos, es decir, computados a partir de la estructura molecular de los compuestos químicos teniendo en cuenta las propiedades químicas de sus átomos y sus tipos de enlace químico. Lxs autorxs plantearon la hipótesis de que modelar varios tipos de reactividad de forma conjunta por medio de un modelo MTL mejoraría el rendimiento predictivo de los modelos en los conjuntos de datos más pequeños y, de hecho, los modelos MTL superaron a los enfoques de modelado de tarea única para predecir sitios de reactividad de cianuros y proteínas. Lxs autorxs concluyeron que el alto rendimiento alcanzado por su modelo MTL para tales tareas posiblemente se debe a que dicho enfoque de aprendizaje es particularmente beneficioso para modelar tareas asociadas a conjuntos de datos pequeños y diversos.

Finalmente, Wu and Wei [428] estudiaron el rendimiento de modelos multi-tarea en un nuevo conjunto propuesto de *descriptores moleculares topológicos específicos del elemento (ESTD)* por sus siglas en inglés), los cuales fueron específicamente diseñados para análisis cuantitativo de toxicidad y predicción en moléculas pequeñas. Lxs autorxs experimentaron con modelos basados en DNNs de tarea única, entrenados por medio de un enfoque MTL, *Bosques Aleatorios (Random Forests)* y *árboles de decisión (Decision Trees)*, y validaron sus resultados por medio de cuatro conjuntos de datos de referencia comprendiendo mediciones cuantitativas de toxicidad. Según reporta su trabajo, el modelo MTL produjo los mejores resultados, que lxs autorxs atribuyeron a la inherente correlación entre los diferentes criterios de valoración cuantitativos de toxicidad.

Aunque ninguno de los artículos publicados aquí citados tuvo como objetivo predecir mutagenicidad de Ames usando un modelo entrenado por medio de una estrategia MTL, nos permitieron evaluar positivamente la capacidad de los modelos MTL basados en DNNs para modelar diversas tareas vinculadas a toxicidad y elaborar hipótesis de investigación en torno a su desempeño en la predicción de mutagenicidad. En este escenario, presentamos en este capítulo de la tesis una estrategia MTL aplicada a un modelo basado en DNNs para modelar la prueba de mutagenicidad de Ames mediante la integración de información de mutagenicidad de compuestos para diversas cepas de *S. typhimurium*.

## 5.4. Metodología y conjunto de datos empleado

En esta sección brindamos una descripción detallada de nuestro diseño experimental y de la etapa de preprocesamiento de los datos utilizados en la propuesta, además de una explicación de la arquitectura del modelo desarrollado, su proceso de entrenamiento y evaluación. Todos los datos y el código fuente utilizados y desarrollados en el marco de la propuesta presentada en este capítulo son públicamente accesibles<sup>1,2</sup>.

### 5.4.1. Sanitización y etiquetado del conjunto de datos

Para llevar a cabo nuestros experimentos, utilizamos el conjunto de datos *ISSSTY v1-a* [39], que contiene datos públicamente accesibles de mutagenicidad *in vitro* en diferentes cepas de *Salmonella typhimurium* (prueba de Ames) para 7.367 compuestos [174]. Dicho conjunto de datos fue recopilado y curado por el *Istituto Superiore di Sanita'* (ISS) y comprende información sobre el resultado experimental de la prueba de Ames en una amplia variedad de cepas de *S. typhimurium* con y sin activación metabólica. Además, incluye una etiqueta de mutagenicidad *general* para cada compuesto evaluado, la cual fue computada teniendo en cuenta el resultado de la evaluación de mutagenicidad de dichos compuestos en todas las cepas disponibles. En dicho conjunto de datos, un compuesto fue marcado como *mutagénico* o *positivo* cuando exhibió resultados positivos para **al menos una cepa**, independientemente de la cepa específica y de si estaba o no bajo activación metabólica. Un compuesto fue marcado como *no mutagénico* o *negativo* si verifica dos condiciones: (i) no exhibió resultados *positivos* o *equivocos* en ninguna de las cepas analizadas y (ii) el compuesto dio *negativo* para al menos una cepa entre las cepas *TA1535*, *TA100* y *TA97*, y al menos una cepa entre las cepas *TA1538*, *TA98* y *TA1537*, con y sin activación metabólica [38]. En dicho conjunto de datos, además, un compuesto fue etiquetado como *equivoco*, si para ninguna cepa arrojó resultados *positivos* y hay al menos un resultado *equivoco* para cualquiera de las cepas; o como *no concluyente*, si no se proporcionan suficientes datos experimentales para respaldar una de las etiquetas descriptas anteriormente.

Los 7.367 compuestos recuperados inicialmente del conjunto de datos *ISSSTY* fueron analizados y sanitizados por nuestro equipo de expertxs en química medicinal antes de su uso para la generación de modelos. La base de datos fue procesada por medio del software *LigPrep* [335], implementado en la Suite Maestro [336]. Siguiendo las prácticas estándar en el dominio y el criterio profesional de lxs

---

<sup>1</sup>Código fuente: [https://github.com/VirginiaSabando/MTL\\_DNN\\_Ames](https://github.com/VirginiaSabando/MTL_DNN_Ames)

<sup>2</sup>Conjunto de datos: <https://data.mendeley.com/datasets/ktc6gbfsbh>



expertxs a cargo del proceso, sintetizado en el capítulo 3, en primera instancia se excluyeron mezclas, polímeros y metales del conjunto de datos y se eliminaron contraiones y se ionizaron los distintos compuestos del conjunto a un valor de  $pH = 7,2$ . Se consideraron los estereoisómeros (enantiómeros R/S, diastereómeros, isómeros cis/trans), definiéndose cada compuesto según su forma isomérica, la cual se puede ver en el código SMILES en el conjunto de datos resultante. Como resultado de este proceso de sanitización, todas las cadenas SMILES fueron convertidas a su forma canónica mediante RDKit [132] y se eliminaron eventuales duplicidades, tras lo que finalmente obtuvimos 6.445 compuestos. Posteriormente, calculamos descriptores moleculares  $0D$ ,  $1D$  y  $2D$  para los compuestos resultantes usando el software *Mordred* [268]. Se eliminaron aquellos descriptores que presentaron más del 60 % de valores faltantes, lo que dio por resultado 1.360 descriptores moleculares por compuesto. Luego, todos los valores de descriptores faltantes fueron imputados por el valor medio del descriptor para el conjunto de datos, así como también fueron eliminados los descriptores que presentaron valores constantes [92, 224].

Luego del proceso de sanitización de los datos, procedimos a implementar nuestra estrategia de etiquetado, ilustrada paso a paso en la figura 5.1. Primero, analizamos las etiquetas presentes en el conjunto de datos resultante con el fin de prepararlo para la etapa de modelado. Estudiamos la etiqueta *general* provista por defecto en el conjunto de datos *ISSSTY* y descartamos aquellos compuestos que tuvieran una etiqueta *general no concluyente* en el conjunto de datos original, ya que tal etiqueta indicaría *a priori* que no hay suficiente información disponible sobre su potencial mutagénico en el conjunto de datos para la posterior tarea de modelado de mutagenicidad de Ames, tal y como ilustra la figura 5.1 (a). La etiqueta *general* provista por el conjunto de datos *ISSSTY* original no fue utilizada en ninguna de las etapas posteriores del diseño experimental.

Luego, integramos todas las etiquetas correspondientes a variaciones de una misma cepa (es decir, valores de actividad de la cepa con sus diferentes activaciones metabólicas) en una sola etiqueta para esa cepa—figura 5.1 (b). En particular, calculamos etiquetas para las cepas **TA98**, **TA100**, **TA102**, **TA1535** y **TA1537**, siguiendo el estándar *OECD-five*, que indica que se debe realizar la prueba Ames sobre las cepas *TA98*, *TA100*, *TA1535*, *TA1537* (o *TA97*) y *E. coli* (o *TA102*) [284]. Como resultado, obtuvimos un conjunto de datos que consta de 6.445 compuestos y cinco etiquetas por compuesto, cada una correspondiente a una cepa de *S. typhimurium* diferente. Siguiendo los criterios de etiquetado establecidos en la creación del conjunto de datos *ISSSTY* [38], estas etiquetas podrían ser *positivas*, *negativas*, *equivocas* o *no concluyentes*. Es importante notar que, en esta instancia de la etapa de preprocesamiento de los datos, aquellos compuestos que presentaran una etiqueta *no concluyente* en una determinada cepa necesariamente tendrían una etiqueta diferente a *no concluyente*

en al menos otra cepa, por haberse eliminado anteriormente los compuestos con etiquetas *generales no concluyentes* (la cual solo es posible de no haber etiquetas *positivas*, *negativas* o *equívocas* en ninguna cepa).

Posteriormente, modificamos todas las etiquetas de compuestos marcados como *no concluyentes* y *equívocos* en todas las cepas a una nueva etiqueta: *indefinido*. La razón detrás de este paso es que, si bien las etiquetas *no concluyente* y *equívoco* no brindan información significativa para predecir mutagenicidad de Ames, un compuesto podría tener diferentes etiquetas para las distintas cepas bajo estudio. De esta forma, aquellos compuestos etiquetados como *indefinido* para una o más cepas, pero que presenten etiquetas *positivo* o *negativo* en las cepas restantes, aún se tendrían en cuenta durante el entrenamiento del modelo MTL, por cuanto el entrenamiento del modelo solamente afectaría a las tareas correspondientes a cepas con dichas etiquetas. Después de este paso, se eliminaron algunos compuestos que tenían etiquetas no definidas para todas las cepas en el conjunto de datos, lo que dio como resultado un conjunto de datos final de 5.536 compuestos etiquetados. Este paso se ilustra en la figura 5.1 (c).

Finalmente, definimos etiquetas *globales* para cada compuesto, la cual tiene en cuenta las etiquetas computadas para todas las cepas objetivo. Estas etiquetas son necesarias para la predicción final de mutagenicidad de Ames por medio de una estrategia de consenso. Es posible definir la etiqueta *global* por medio de diferentes enfoques de etiquetado. En nuestro trabajo implementamos dos enfoques para computar estas etiquetas globales: un enfoque *conservador* y uno *laxo*.

El enfoque conservador de etiquetado consiste en los siguientes criterios:

- un compuesto se etiqueta como *positivo* o *mutagénico* si alguna de las cepas lo marca como positivo (mutagénico);
- un compuesto se etiqueta como *negativo* o *no mutagénico* si todas las cepas lo marcan como *negativo* (no mutagénico), y
- un compuesto se etiqueta como *indefinido*, si ninguna de las cepas marca al compuesto como *positivo* (mutagénico) y al menos una de ellas tiene una etiqueta *indefinida*.

Como se desprende de dichos criterios, este enfoque implica definir un compuesto como *no mutagénico* o *negativo* si y solo si el mismo está etiquetado como *no mutagénico* para todas las cepas consideradas en nuestro diseño experimental. Teniendo en cuenta que las cepas de *S. typhimurium* aportan información diferente pero complementaria y que no existe un consenso universal sobre qué

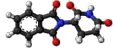

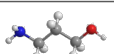
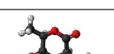
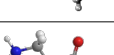
subconjuntos de cepas deben emplearse para la detección de mutagenicidad de Ames, el propósito de adoptar un enfoque conservador es asegurar que los compuestos químicos marcados como *no mutagénicos* hayan sido realmente probados en todas las cepas involucradas en el diseño experimental. De esta manera, se evitaría una imputación potencialmente incorrecta como *no mutagénico* a un compuesto que podría no haber sido probado experimentalmente para alguna cepa en particular. Por esta razón, a partir de este enfoque computamos la etiqueta global *Overall*, que fue la etiqueta global empleada finalmente en todas las etapas de modelado por consenso en nuestro diseño experimental.

Una de las consecuencias del enfoque de etiquetado conservador es que conduce a un desbalance de clases severo para la etiqueta global, la cual se puede apreciar en la tabla 5.1, al observar la proporción de compuestos positivos y negativos de la etiqueta *Overall* computada a partir de dicho enfoque. En este sentido, adicionalmente y por completitud llevamos a cabo una serie de experimentos adoptando un enfoque de etiquetado laxo, el cual representa un enfoque alternativo al conservador y no fue el adoptado en nuestro diseño experimental. Según el enfoque laxo, un compuesto recibe como etiqueta global el valor *no mutagénico* cuando dicho compuesto no ha sido marcado como mutagénico por ninguna cepa y al menos una cepa lo ha marcado como *no mutagénico*, aunque no todas las cepas bajo análisis tuvieran disponible su información de mutagenicidad. La figura 5.1 (c) ilustra los dos criterios de etiquetado aquí descritos. Es importante notar que, si bien la definición de una estrategia de etiquetado laxa para la clase negativa reduce el desequilibrio de clase, se corre el riesgo de potencialmente etiquetar incorrectamente un compuesto mutagénico como no mutagénico, teniendo en cuenta que, en el conjunto de datos original, para ciertas cepas no hay información experimental de la mutagenicidad de determinados compuestos, o dicha información es equívoca. En otras palabras, al seguir una estrategia de etiquetado de este tipo, la etiqueta global en el caso de compuestos *no mutagénicos* podría diferir de los resultados reales de la prueba de Ames para ciertos compuestos, dado que estamos imputando valores negativos a compuestos que no han sido probados exhaustivamente en todas las cepas.

El resultado del cómputo de las etiquetas globales se muestra en la figura 5.1 (d). La tabla 5.1 resume las distribuciones de los datos por cepa con respecto a etiquetas *positivas* y *negativas*, así como la cantidad de compuestos etiquetados para cada una de ellas.

#### 5.4.2. Arquitectura de los modelos propuestos

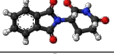
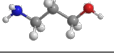
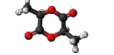
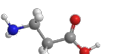
Con el objeto de responder a las preguntas de investigación planteadas, diseñamos un modelo QSAR que predice la mutagenicidad de Ames en función de datos experimentales de cepas

Compuesto	98	98_S9	100	100_S9	102	102_S9	1535	1535_S9	1537	1537_S9	General
	+	+	-	-	inc	inc	+	inc	-	-	+
	<del>inc</del>	<del>inc</del>	<del>inc</del>	<del>inc</del>	<del>-</del>	<del>-</del>	<del>inc</del>	<del>inc</del>	<del>-</del>	<del>-</del>	<del>inc</del>
	+/-	-	+	+/-	-	-	-	+/-	-	-	+
	-	-	-	-	inc	-	-	-	-	inc	-
	+/-	-	inc	+/-	inc	-	+/-	inc	-	-	+/-

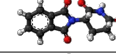
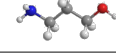
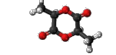
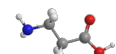
**Referencias**

+	Positivo / Mutagénico
-	Negativo / No mutagénico
+/-	Equívoco
inc	No concluyente
?	Indefinido
*	Don't care (+, -, ?)

**(a)**

Compuesto	98	100	102	1535	1537
	+	-	inc	+	-
	+/-	+	-	+/-	-
	-	-	-	-	-
	+/-	+/-	-	+/-	-

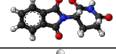
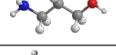
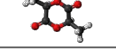
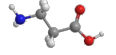
**(b)**

Compuesto	98	100	102	1535	1537
	+	-	?	+	-
	?	+	-	?	-
	-	-	-	-	-
	?	?	-	?	-

**(c)**

TA <sub>1</sub>	TA <sub>2</sub>	TA <sub>3</sub>	Resultado	Descripción
+	*	*	+	Mutagénico (Conservador / Laxo)
-	-	-	-	No Mutagénico (Conservador / Laxo)
-	?	-	?	Indefinido (Conservador)
-	-	-	-	No Mutagénico (Laxo)

**(d)**

Compuesto	98	100	102	1535	1537	Overall	Laxo
	+	-	?	+	-	+	+
	?	+	-	?	-	+	+
	-	-	-	-	-	-	-
	?	?	-	?	-	?	-

**(e)**

Figura 5.1: Resumen de la estrategia de etiquetado: primero descartamos los compuestos con etiqueta *general no concluyente* (a), luego agrupamos las etiquetas por cepa (b) y reemplazamos las etiquetas *equivoco* y *no concluyente* por la etiqueta *indefinido* (c). De esta manera, obtuvimos un conjunto de datos con cinco etiquetas por compuesto, una para cada cepa objetivo. Finalmente, siguiendo los enfoques de etiquetado *conservador* y *laxo* (d) computamos dos etiquetas globales a partir de las etiquetas por cepa para cada compuesto (e). Cabe destacar que la etiqueta empleada como referencia a lo largo de nuestro trabajo es la etiqueta *Overall*, computada siguiendo el enfoque *conservador* de etiquetado global.

Cepa	No. Compuestos	Positivos/Negativos	% Positivos
TA98	4.854	1.676 / 3.178	34,53 %
TA100	5.366	2.096 / 3.270	39,06 %
TA102	975	226 / 749	23,18 %
TA1535	2.657	436 / 2.221	16,41 %
TA1537	2.229	365 / 1.864	16,38 %
<i>Overall</i>	3.334	3.103 / 231	93,07 %

Tabla 5.1: Resumen del contenido del conjunto de datos resultante de nuestro proceso de etiquetado, empleado en nuestros experimentos, incluyendo la relación de desbalance de clase (porcentaje de instancias etiquetadas como positivas).

individuales de *S. typhimurium*. Dicho modelo QSAR se basa en redes neuronales profundas (DNNs) y fue desarrollado siguiendo una estrategia de aprendizaje multi-tarea (MTL). En el escenario experimental propuesto, el propósito del modelo MTL es aprender a predecir los resultados de la prueba de mutagenicidad para cada una de las cepas individuales en función del conjunto de datos descripto, al mismo tiempo detectando y aprendiendo los determinantes estructurales y químicos del conjunto de compuestos que hacen a la complementariedad o correlación entre dichas cepas bacterianas por medio del entrenamiento conjunto de parámetros entrenables de la DNN compartidos entre todas las cepas.

Para lograr dicho modelo, tratamos los resultados de las pruebas de mutagenicidad de cada cepa como un objetivo predictivo separado. Empleamos una arquitectura MTL basada en DNN que consta de dos segmentos de red conectados: (i) un *segmento compartido* que consta de una DNN cuyos pesos y funciones de activación son compartidas por todas las cepas objetivo, y (ii) un *segmento específico* para cada cepa objetivo. El segmento específico consta de cinco DNNs individuales, una para cada cepa objetivo: *TA98*, *TA100*, *TA102*, *TA1535* y *TA1537*. La salida del segmento compartido alimenta el segmento específico correspondiente a cada una de las tareas predictivas, como se puede ver en la figura 5.2 (a). Los pesos entrenables del segmento compartido son aprendidos mediante la optimización iterativa de las cinco tareas predictivas a la vez, por lo que las primeras capas de la arquitectura, pertenecientes al segmento compartido, se entrenan mediante la combinación de información provista por las cinco cepas objetivo. La salida del segmento específico a cada tarea, por su parte, constituye la salida de la arquitectura de red neuronal, por lo que nuestro modelo MTL tiene cinco salidas, una para cada tarea o cepa.

Los valores a predecir por el modelo MTL corresponden a las etiquetas de los compuestos químicos

del conjunto de datos para cada una de las cinco cepas consideradas, que pueden ser *positivo* (1), *negativo* (0) o *indefinido* (-1). Las etiquetas *indefinido* se enmascararon durante el proceso de entrenamiento; como resultado, un compuesto que presentara un valor de etiqueta *indefinido* para una determinada cepa no tendría impacto en el cálculo de la función de pérdida de la red neuronal durante el proceso de entrenamiento correspondiente a esa cepa. Los resultados del modelo MTL finalmente fueron combinados a través de una estrategia de consenso que calcula la predicción de la prueba de Ames. La arquitectura de este modelo, denominado  $MTL-DNN_{Cons}$ , es representada en la figura 5.2 (a). Las etiquetas utilizadas para evaluar el rendimiento de  $MTL-DNN_{Cons}$  fueron las etiquetas *Overall*, calculadas mediante los criterios de etiquetado descriptos anteriormente.

Finalmente, de forma tal de establecer una comparación justa con otros enfoques de modelado de la mutagenicidad de Ames, también proporcionamos resultados experimentales para otros tres modelos. Primero, usamos un modelo basado en DNNs de tarea única, como se muestra en la figura 5.2 (b), que consta de una sola salida y está entrenado para la tarea de predecir las etiquetas *Overall*, a saber,  $STL-DNN_{Overall}$ . En segundo lugar, desarrollamos un modelo basado en el consenso de cinco modelos DNN de tarea única, cada uno de ellos entrenado para una de las cepas individuales de *S. typhimurium*, a saber,  $STL-DNN_{Cons}$ , representado en la figura 5.2 (c). Por último, un modelo de referencia basado en *Random Forest*, a saber,  $RF_{Overall}$ , el cual también fue entrenado para predecir la tarea *Overall*, representado en la figura 5.2 (d). Todos nuestros modelos basados en DNN fueron creados y desarrollados empleando las herramientas provistas por las librerías Keras y Tensorflow [71], y el análisis de rendimiento de los distintos modelos se realizó por medio de herramientas provistas por la librería Scikit-Learn [293].

### 5.4.3. Diseño experimental

Nuestro flujo de trabajo experimental consta de varios pasos pensados para garantizar la reproducibilidad y la imparcialidad durante la generación y evaluación del modelo propuesto. En la figura 5.3 brindamos una descripción general de nuestro diseño experimental. El primer paso consistió en una búsqueda preliminar en la que probamos diferentes hiperparámetros para nuestros modelos basados en DNNs a fin de seleccionar aquellos que exhibieran el mejor rendimiento. Este proceso se realizó tanto para nuestro modelo propuesto  $MTL-DNN_{Cons}$  como para los modelos de referencia  $STL-DNN_{Overall}$ ,  $RF_{Overall}$  y  $STL-DNN_{Cons}$ . Considerando que dicha búsqueda preliminar ayuda a encontrar la mejor combinación de hiperparámetros para la tarea predictiva en cuestión y, por lo tanto, potencialmente implica realizar una gran cantidad de experimentos, llevamos a cabo dicha exploración dividiendo el conjunto de datos en particiones fijas: 70 % para entrenamiento

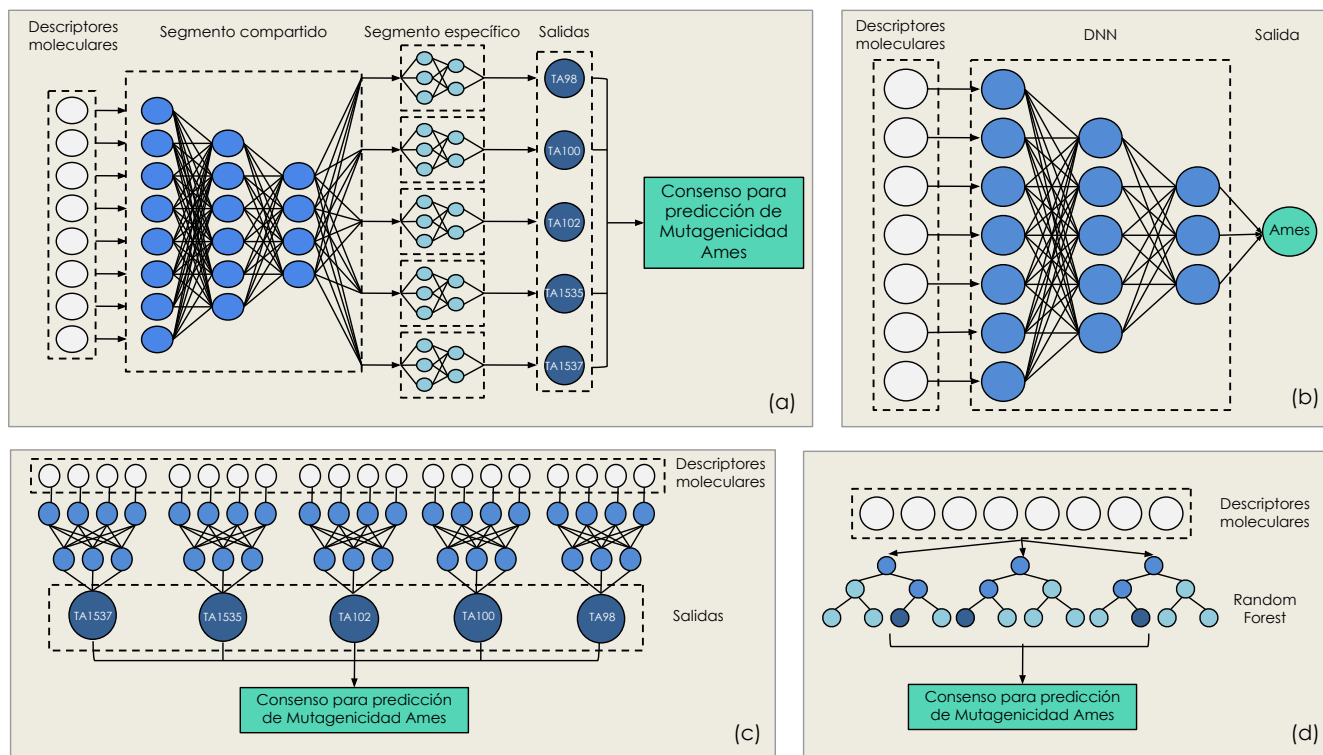


Figura 5.2: Resumen de las arquitecturas correspondientes a los modelos de mutagenicidad de Ames que comparamos en este trabajo. (a) Arquitectura de nuestro modelo  $MTL-DNN_{Cons}$ . Cada tarea predictiva corresponde a una cepa ( $TA98$ ,  $TA100$ ,  $TA102$ ,  $TA1535$  y  $TA1537$ ). La arquitectura consta de dos segmentos DNN conectados: un *segmento compartido* por todas las tareas y un *segmento específico* a cada tarea con secuencias de capas completamente conectadas individuales a cada cepa. Las salidas son agregadas por medio de una estrategia de consenso. (b) Arquitectura  $STL-DNN_{Overall}$  de tarea única para la predicción de la prueba de mutagenicidad de Ames. (c) Arquitectura del modelo  $STL-DNN_{Cons}$ , que consta de cinco modelos de tarea única para modelar la mutagenicidad de Ames medida en cada una de las cepas combinados por una estrategia de consenso. (d) Modelo  $RF_{Overall}$  de referencia.

(*Train*), 10% para validación interna (*internal validation*) y 20% para validación externa (*external validation*), siendo esta última partición preservada intacta hasta las etapas de evaluación final de los modelos. Empleamos diferentes semillas de inicialización aleatorias de los parámetros entrenables de las DNNs para cada ejecución en la búsqueda exploratoria, la cual se ilustra en la figura 5.3 (a).

Entre los hiperparámetros que variamos durante la búsqueda preliminar para nuestros modelos basados en DNN, detallados en el capítulo 2, se encuentran el coeficiente de regularización  $L2$   $\lambda \in \{0,001; 0,005; 0,01\}$ , el número de nodos neuronales por capa de la arquitectura de red del *segmento*

*compartido*  $n \in \{(100, 50, 10, 5), (100, 50, 20, 10), (200, 100, 50, 10), (200, 100, 20, 10), (200, 100, 10, 5)\}$  y el número de capas de la arquitectura neuronal de las redes del *segmento específico*  $s \in \{0, 1, 2\}$ , donde el número de nodos por capa coincide con el número de nodos neuronales en las últimas  $s$  capas del *segmento compartido*. También estudiamos el impacto del uso de funciones de costo ponderado durante el entrenamiento, de modo que la función de pérdida de las redes neuronales se ajustó con un peso calculado en función del desbalance de clase en el conjunto de datos. Dado que los escenarios de desequilibrio de clase pueden ser diferentes para cada cepa, se aplicaron diferentes pesos a cada DNN del *segmento específico*. Sin embargo, en ninguno de los experimentos desarrollados durante la búsqueda exploratoria de hiperparámetros esta técnica arrojó mejores resultados que aplicar una función de costo no ponderada. Otros hiperparámetros, como las funciones de activación de los nodos internos a las redes, tamaño de *minibatch* y el valor de *learning rate* de los modelos, se variaron durante los experimentos preliminares, pero no exhibieron un impacto significativo en el proceso de aprendizaje de los modelos y, por lo tanto, fueron fijados en el proceso de búsqueda. En el caso del modelo de referencia  $RF_{Overall}$ , variamos el número de estimadores  $n\_estimators \in \{100, 500, 700, 1.000\}$ , la profundidad máxima de los árboles de decisión  $max\_depth \in \{5, 10, 15\}$ , y también analizamos el impacto de incorporar información sobre el desbalance de clase por medio de la aplicación de pesos a la función de costo del modelo durante el entrenamiento.

Como resultado de la etapa de búsqueda exploratoria, seleccionamos la combinación de hiperparámetros que resultó la más favorable para cada arquitectura de las descritas anteriormente. Consecuentemente, obtuvimos un modelo *MTL-DNN*, dos modelos *STL-DNN* y un modelo basado en *Random Forest*. Los detalles completos sobre la parametrización de estos modelos se listan en las tablas 5.2 y 5.3. A partir de dichas parametrizaciones, llevamos a cabo una etapa de entrenamiento por medio de validación cruzada de cinco pliegos (*five-fold cross-validation*), fusionando las particiones de entrenamiento y validación interna en una sola partición para calcular los cinco pliegos. Usamos los mismos cinco pliegos para todos los modelos antes enumerados y realizamos diez repeticiones de cada experimento usando diferentes semillas de inicialización aleatoria, para asegurarnos de que el desempeño observado no estuviera ligado a la varianza inherente en las particiones de datos o a la inicialización de los pesos entrenables del modelo. Esta etapa se ilustra en la figura 5.3 (b).

La etapa de validación cruzada dio lugar a  $5 \times 10 = 50$  modelos entrenados para cada arquitectura de las ilustradas en la figura 5.2. Luego, calculamos el rendimiento promedio de cada modelo en los cinco pliegos de validación interna junto con sus intervalos de confianza del 95% para cada una de las diez semillas de inicialización aleatoria y seleccionamos los cinco modelos entrenados correspondientes a la inicialización aleatoria de mejor rendimiento para cada modelo. Finalmente, calculamos los



resultados de la predicción sobre la partición de validación externa del conjunto de datos en dichos modelos entrenados.

Parametrización	MTL-DNN <sub>C<sub>ons</sub></sub>		STL-DNN <sub>Overall</sub>	
No. capas segmento compartido	4	4	3	3
No. capas segmento específico	0	2	-	-
No. nodos por capa	100/50/10/5	200/100/50/10/50/10	100/50/15	200/50/10
No. salidas	5	5	1	1
Función de activación	ReLU	ReLU	ReLU	ReLU
<i>Batch norm</i> $\gamma$	0,9	0,9	0,9	0,9
Regularización L2 $\lambda$	0,005	0,005	0,001	0,0001
Tasa de <i>Dropout</i> por capa	0,25/0,15/0,1	0,25/0,15/0,1	0,25/0,15/0,1	0,25/0,15/0,1
<i>Learning rate</i> $\alpha$	0,0001	0,0001	0,0001	0,0001
Función de costo pesada	no	si	no	si

Tabla 5.2: Hiperparámetros de los modelos *MTL-DNN<sub>C<sub>ons</sub></sub>* y *STL-DNN<sub>Overall</sub>*.

Por tratarse de un modelo de tarea única, el modelo *STL-DNN<sub>Overall</sub>* entrega una predicción de mutagenicidad de Ames por compuesto químico. Sin embargo, el modelo *MTL-DNN* arroja cinco predicciones diferentes, una por cada una de las cepas objetivo consideradas en el proceso experimental. Por tanto, la etapa final para dicho modelo consistió en la evaluación del consenso de dichos resultados, dando lugar al modelo *MTL-DNN<sub>C<sub>ons</sub></sub>*. Teniendo en cuenta las cinco predicciones entregadas por el modelo *MTL-DNN* calculamos el valor de consenso de las predicciones para cada compuesto químico en la partición de validación externa, siguiendo los mismos criterios utilizados para calcular las etiquetas *Overall*, tal y como se muestra en la figura 5.3 (c). La evaluación final del rendimiento de los modelos se realizó comparando dichas predicciones con las etiquetas *Overall* computadas.

Además del diseño experimental aquí detallado, construimos y entrenamos un conjunto de cinco modelos basados en DNN de tarea única, uno para cada cepa objetivo. Dichos modelos se combinaron posteriormente mediante una estrategia de consenso, dando como resultado el modelo *STL-DNN<sub>C<sub>ons</sub></sub>*. Para el desarrollo de estos modelos, seguimos todos los pasos descritos anteriormente, incluyendo la búsqueda exploratoria de hiperparámetros y la etapa de validación cruzada, utilizando las etiquetas de clasificación correspondientes a cada cepa durante el entrenamiento. El propósito del modelo *STL-DNN<sub>C<sub>ons</sub></sub>* fue el de proporcionar un medio para evaluar el impacto en el rendimiento en la predicción de mutagenicidad de Ames al combinar la información sobre cepas individuales por medio de un

Parametrización	STL-DNN <sub>Cons</sub>				
	TA98	TA100	TA102	TA1535	TA1537
No. capas segmento compartido	3	3	3	3	3
No. capas segmento específico	-	-	-	-	-
No. nodos por capa	200/50/10	200/50/10	100/50/15	10/5/2	10/5/2
No. salidas	1	1	1	1	1
Función de activación	ReLU	ReLU	Tanh	Tanh	Tanh
<i>Batch norm</i> $\gamma$	0,9	0,9	0,9	0,9	0,9
Regularización L2 $\lambda$	0,0001	0,001	0,001	0,0001	0,0001
Tasa de <i>Dropout</i> por capa	0,25/0,15/0,1	0,25/0,15/0,1	0,25/0,15/0,1	0,25/0,15/0,1	0,25/0,15/0,1
<i>Learning rate</i> $\alpha$	0,0001	0,0001	0,0001	0,0001	0,0001
Función de costo pesada	no	no	no	no	no

Tabla 5.3: Hiperparámetros para cada modelo de cepa individual comprendidos por *STL-DNN<sub>Cons</sub>*.

enfoque de aprendizaje multi-tarea (MTL) en contraste con la agregación por consenso de múltiples modelos de aprendizaje de tarea única para las mismas cepas.

## 5.5. Resultados obtenidos

En esta sección, presentamos una discusión de los resultados obtenidos de nuestro diseño experimental. Analizamos diferentes escenarios de comparación de los modelos mostrados en la figura 5.2 para responder a las preguntas de investigación propuestas en este trabajo.

Evaluamos el rendimiento de nuestros modelos mediante ocho métricas: *Sensibilidad* ( $S_n$ ), *Especificidad* ( $S_p$ ), *Precisión*, *Exactitud* ( $Acc$ ), *Exactitud Balanceada* ( $BAcc$ ), *Puntaje F1*, *Puntaje H1* y *Coefficiente de Correlación de Matthews* o (*Matthews Correlation Coefficient* -  $MCC$ ). Tal y como se detalla en el capítulo 2 de la presente tesis,  $S_n$  y  $S_p$  miden la capacidad del modelo para detectar compuestos mutagénicos y no mutagénicos, respectivamente, mientras que *Precisión* indica la proporción de compuestos mutagénicos (positivos) que se predijeron correctamente.  $Acc$  mide el porcentaje de predicciones correctas, independientemente de su etiqueta de clase. Por su parte,  $BAcc$  es la media aritmética de  $S_n$  y  $S_p$ , *Puntaje F1* es la media armónica de  $S_n$  y *Precisión*, y *Puntaje H1* es la media armónica de  $S_n$  y  $S_p$ . Finalmente,  $MCC$  es una métrica de calidad para clasificadores binarios, según la cual un valor cercano a 1 indica que ambas clases son predichas correctamente.

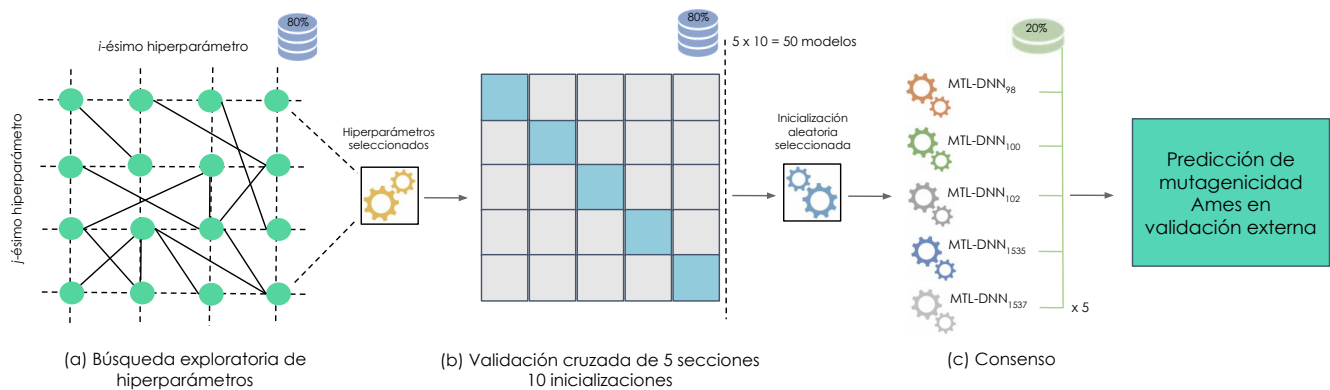


Figura 5.3: Resumen general del diseño experimental. Se utilizaron particiones fijas y el 20% de los datos se reservaron en una partición de validación externa. Realizamos una búsqueda exploratoria a fin de evaluar diferentes combinaciones de hiperparámetros para nuestros modelos (a). Como resultado, seleccionamos la mejor combinación de valores de hiperparámetros en función del rendimiento de los modelos en la partición fija de validación interna. Realizamos una etapa de entrenamiento por medio de validación cruzada de cinco pliegos (*five-fold cross-validation*) con diez semillas aleatorias para la inicialización de los parámetros entrenables de los modelos (b) y elegimos el modelo entrenado a partir de la semilla de mejor rendimiento. Como resultado, obtuvimos cinco modelos para cada arquitectura, correspondientes a cada uno de los pliegos del proceso de validación cruzada. Cada uno de los cinco modelos obtenidos del proceso de validación cruzada fueron evaluados en la partición de datos de validación externa. Obtuvimos las predicciones para cada cepa ( $MTL-DNN_{98}$ ,  $MTL-DNN_{100}$ ,  $MTL-DNN_{102}$ ,  $MTL-DNN_{1535}$ , y  $MTL-DNN_{1537}$ ) (c) y las combinamos a través de una estrategia de consenso para obtener el valor de la predicción final de la prueba de mutagenicidad de Ames.

En nuestro diseño experimental, nos enfocamos en *Puntaje F1* y *Puntaje H1* para llevar a cabo el proceso de selección de modelos.

Como se ilustra en la figura 5.3 (b), seleccionamos los cinco modelos de la etapa de validación cruzada correspondientes al mejor resultado obtenido a través de diez diferentes inicializaciones aleatorias de sus parámetros entrenables. Los rendimientos informados en esta sección se computaron promediando los resultados de tales cinco modelos entrenados en la partición de validación externa y sus intervalos de confianza del 95%. Para la evaluación del rendimiento no tuvimos en cuenta aquellos compuestos que tuvieran una etiqueta de clase *indefinido* en el modelo  $MTL-DNN_{Cons}$  o en el modelo  $STL-DNN_{Overall}$ . Si bien en este capítulo de la tesis nos concentramos en presentar y discutir los resultados en la partición de validación externa, las tablas con los resultados en cada sección de

validación interna se pueden encontrar en el Material Suplementario de nuestra publicación [248].

En primer lugar, nos enfocamos en evaluar si un modelo QSAR entrenado siguiendo una estrategia de aprendizaje MTL sería capaz de sacar mayor provecho de las contribuciones de las diferentes cepas individuales que un modelo QSAR de tarea única entrenado para predecir los resultados *Overall* de la prueba de Ames. En segunda instancia, nos propusimos analizar si una estrategia de aprendizaje MTL permitiría aprovechar la información compartida o complementaria entre las cepas para modelar la prueba de Ames, en comparación con el modelado de cada cepa individual por medio de arquitecturas de tarea única. Para responder a estas dos interrogantes, comparamos el rendimiento de nuestro modelo  $MTL-DNN_{Cons}$  (figura 5.2 (a)) con el rendimiento de los modelos  $STL-DNN_{Overall}$  y  $STL-DNN_{Cons}$  (figuras 5.2 (b) y (c), respectivamente). Además, también informamos el rendimiento del modelo  $RF_{Overall}$ , basado en *Random Forest*, como modelo de referencia para la etiqueta *Overall* (figura 5.2 (d)).

Los rendimientos informados en la tabla 5.4 se calcularon usando las etiquetas *Overall* como referencia y muestran que el modelo  $MTL-DNN_{Cons}$  superó los resultados obtenidos por los modelos  $STL-DNN_{Overall}$  y  $STL-DNN_{Cons}$ . En general, se puede observar que los tres modelos presentan valores altos de *Sn* y *Precisión*, lo cual indica que son capaces de detectar correctamente una alta proporción de los compuestos mutagénicos. Sin embargo, se observan diferencias significativas en la métrica *Sp*, es decir, la capacidad de los modelos para detectar compuestos no mutagénicos. En este sentido, el modelo de consenso  $MTL-DNN_{Cons}$  exhibió el mejor rendimiento en términos de *Sp*, alcanzando un valor  $Sp = 0,86$ . Por el contrario, el modelo  $STL-DNN_{Overall}$  obtuvo un valor medio de  $Sp = 0,43$ . Teniendo en cuenta que los compuestos mutagénicos constituyen la clase mayoritaria, este fenómeno podría indicar que el escenario de fuerte desequilibrio de clases (compuestos *positivos* versus *negativos*) que caracteriza al conjunto de datos utilizado representa un desafío adicional para el proceso de aprendizaje del modelo  $STL-DNN_{Overall}$ , que es entrenado con una etiqueta general que agrupa la información de todas las cepas.

El modelo  $STL-DNN_{Cons}$ , construido a partir del consenso entre modelos de tarea única entrenados para las cinco cepas de *S. typhimurium* consideradas en nuestra propuesta, también exhibe un rendimiento más bajo que  $MTL-DNN_{Cons}$  en términos de *Sp*, con un valor promedio de 0,72. A la luz de tal resultado, resulta aparente que el consenso de los modelos de cepas individuales está omitiendo información relevante para detectar compuestos no mutagénicos que sí logra capturarse de manera efectiva por medio de un enfoque de aprendizaje MTL. Es importante tener en cuenta que el modelo  $STL-DNN_{Cons}$  superó al modelo  $STL-DNN_{Overall}$ , lo que sugiere que tener en cuenta la información proporcionada por las cepas individuales podría tener un impacto positivo en el

rendimiento predictivo, en contraste con el modelado de un valor *Overall*. Dado que las métricas *BAcc* y *Puntaje H1* evalúan la capacidad predictiva del modelo en escenarios de desbalance de clase, los bajos valores de *Sp* obtenidos por  $STL-DNN_{Overall}$  también tienen un impacto en esas métricas. Finalmente, el modelo de referencia  $RF_{Overall}$  exhibió un desempeño inferior a los modelos basados en DNN, en especial en contraste con nuestro modelo propuesto  $MTL-DNN_{Cons}$ .

También analizamos los rendimientos de los modelos individuales por cepa que forman parte del modelo  $STL-DNN_{Cons}$ , ilustrado en la figura 5.2 (c), para determinar si el enfoque de aprendizaje MTL implicaría potencialmente algún beneficio con respecto al modelado de la mutagenicidad de Ames por medio de modelos individuales para cada cepa. Para ello, comparamos los resultados obtenidos por las arquitecturas  $MTL-DNN$  y  $STL-DNN$  en cada cepa, es decir, previamente al cómputo del consenso, utilizando las etiquetas de cepas individuales como referencia. Como se puede ver en la tabla 5.5, no se observan diferencias significativas en el rendimiento producido por las salidas de la arquitectura  $MTL-DNN$  y los modelos  $STL-DNN$  individuales. A pesar de ello, cuando evaluamos los resultados en el consenso de las cinco cepas, el modelo  $MTL-DNN_{Cons}$  exhibe un rendimiento predictivo mucho más alto que su contraparte de tarea única STL para la mayoría de las métricas (tabla 5.4). Esta observación vale especialmente para la métrica *Sp*, la cual evidencia una alta proporción de falsos positivos (compuestos no mutagénicos o *negativos* incorrectamente predichos como mutagénicos o *positivos*) en el caso del modelo  $STL-DNN_{Cons}$ . El criterio de etiquetado utilizado en el consenso, correspondiente con la etiqueta *Overall*, consiste en un enfoque **conservador** que solamente etiqueta un compuesto como no mutagénico o *negativo* si para todas las cepas se ha etiquetado de tal forma al compuesto en cuestión. Teniendo en cuenta esto, un modelo de cepa individual de bajo rendimiento en términos de *Sp*, es decir, con limitadas capacidades para predecir compuestos no mutagénicos, podría dar lugar a falsos positivos en la predicción del consenso para la prueba de Ames, por lo que es esperable que aquellas cepas para las cuales haya desequilibrio de clases o falta de datos afecten negativamente el rendimiento predictivo general. Este es el caso de la cepa *TA102*, que tiene el menor número de compuestos etiquetados del conjunto de datos, tal y como se muestra en la tabla 5.1. Sin embargo, un enfoque de aprendizaje MTL resulta robusto para escenarios con pocos datos o desequilibrio de clases, ya que el *segmento compartido* de la arquitectura del modelo se entrena conjuntamente a partir de datos provenientes de todas las cepas.

Hasta el alcance de nuestro conocimiento y tras una extensa revisión de la literatura, la publicación científica que respalda nuestra propuesta [248] constituye el primer trabajo publicado en la literatura del dominio que presenta un modelo QSAR empleando un enfoque de aprendizaje MTL para modelar la mutagenicidad de Ames. Los resultados obtenidos muestran que el modelado de la mutagenicidad

	Sp	Sn	Precisión	Acc	BAcc	Puntaje F1	Puntaje H1	MCC
<b><i>MTL-DNN</i><sub>C<sub>ons</sub></sub></b>	<b>0,86 ± 0,04</b>	0,99 ± 0,00	<b>0,99 ± 0,00</b>	<b>0,99 ± 0,00</b>	<b>0,93 ± 0,02</b>	<b>0,99 ± 0,00</b>	<b>0,92 ± 0,02</b>	<b>0,89 ± 0,03</b>
<i>STL-DNN</i> <sub>C<sub>ons</sub></sub>	0,72 ± 0,04	0,99 ± 0,00	0,98 ± 0,00	0,98 ± 0,00	0,86 ± 0,02	0,99 ± 0,00	0,84 ± 0,02	0,82 ± 0,03
<i>STL-DNN</i> <sub>Overall</sub>	0,43 ± 0,06	0,99 ± 0,00	0,96 ± 0,00	0,95 ± 0,00	0,71 ± 0,03	0,98 ± 0,00	0,60 ± 0,06	0,60 ± 0,04
<i>RF</i> <sub>Overall</sub>	0,60 ± 0,04	0,91 ± 0,01	0,97 ± 0,00	0,90 ± 0,01	0,76 ± 0,02	0,94 ± 0,01	0,73 ± 0,03	0,39 ± 0,02

Tabla 5.4: Resultados promedio en la partición de validación externa para los modelos *MTL-DNN*<sub>C<sub>ons</sub></sub>, *STL-DNN*<sub>Overall</sub>, *STL-DNN*<sub>C<sub>ons</sub></sub> y *RF*<sub>Overall</sub> junto con su intervalo de confianza del 95 %. Los resultados aquí listados fueron computados por medio de las predicciones en la partición de validación externa en los cinco modelos entrenados resultantes de la etapa de validación cruzada de nuestro diseño experimental. Como se puede observar de los mejores resultados, destacados en **negrita**, nuestro modelo *MTL-DNN*<sub>C<sub>ons</sub></sub> supera significativamente el rendimiento alcanzado por los modelos basados en estrategias de aprendizaje de tarea única.

		Sp	Sn	Precisión	Acc	BAcc	Puntaje F1	Puntaje H1	MCC
TA98	<i>STL-DNN</i> <sub>98</sub>	0,85 ± 0,01	0,82 ± 0,01	0,78 ± 0,01	0,84 ± 0,01	0,83 ± 0,01	0,80 ± 0,01	0,83 ± 0,01	0,66 ± 0,01
	<i>MTL-DNN</i> <sub>98</sub>	0,85 ± 0,01	0,81 ± 0,01	0,78 ± 0,01	0,84 ± 0,00	0,83 ± 0,00	0,80 ± 0,00	0,83 ± 0,00	0,66 ± 0,01
TA100	<i>STL-DNN</i> <sub>100</sub>	0,83 ± 0,02	0,76 ± 0,04	0,78 ± 0,01	0,80 ± 0,01	0,79 ± 0,01	0,77 ± 0,01	0,79 ± 0,01	0,59 ± 0,01
	<i>MTL-DNN</i> <sub>100</sub>	0,81 ± 0,02	0,77 ± 0,01	0,77 ± 0,01	0,79 ± 0,00	0,79 ± 0,00	0,77 ± 0,00	0,79 ± 0,00	0,58 ± 0,01
TA102	<i>STL-DNN</i> <sub>102</sub>	0,82 ± 0,03	0,53 ± 0,04	0,52 ± 0,02	0,75 ± 0,01	0,68 ± 0,01	0,52 ± 0,02	0,64 ± 0,03	0,35 ± 0,02
	<i>MTL-DNN</i> <sub>102</sub>	0,79 ± 0,03	0,55 ± 0,04	0,49 ± 0,04	0,73 ± 0,02	0,67 ± 0,02	0,52 ± 0,03	0,65 ± 0,03	0,33 ± 0,05
TA1535	<i>STL-DNN</i> <sub>1535</sub>	0,94 ± 0,02	0,57 ± 0,02	0,66 ± 0,06	0,88 ± 0,01	0,76 ± 0,01	0,61 ± 0,03	0,71 ± 0,01	0,54 ± 0,04
	<i>MTL-DNN</i> <sub>1535</sub>	0,93 ± 0,01	0,59 ± 0,02	0,61 ± 0,04	0,87 ± 0,01	0,76 ± 0,02	0,60 ± 0,03	0,72 ± 0,02	0,52 ± 0,04
TA1537	<i>STL-DNN</i> <sub>1537</sub>	0,94 ± 0,01	0,79 ± 0,01	0,71 ± 0,03	0,91 ± 0,01	0,86 ± 0,01	0,75 ± 0,01	0,85 ± 0,01	0,70 ± 0,02
	<i>MTL-DNN</i> <sub>1537</sub>	0,93 ± 0,01	0,78 ± 0,02	0,68 ± 0,02	0,90 ± 0,01	0,85 ± 0,01	0,73 ± 0,01	0,85 ± 0,01	0,67 ± 0,02

Tabla 5.5: Rendimiento obtenido por las arquitecturas *MTL-DNN* y *STL-DNN* en cada una de las cinco cepas consideradas en el proceso de modelado. En la mayoría de los casos, no hay diferencias significativas entre las salidas del modelo *MTL-DNN* y los cinco modelos individuales de la arquitectura *STL-DNN*.

de Ames aplicando una estrategia de consenso a partir de un modelo MTL supera las estrategias de modelado a partir de etiquetas globales (*Overall*), que es el enfoque de modelado comúnmente hallado en la literatura. Además, nuestro enfoque también supera la estrategia de consenso basada en modelos individuales de tarea única por cepas. De esta forma, nuestros resultados no solo confirman la viabilidad de aplicar aprendizaje multi-tarea para el modelado de la prueba de Ames, sino que ponen de manifiesto sus múltiples beneficios en escenarios de modelado complejos, con información escasa y desbalance de clases.

En términos de interpretabilidad, los modelos de aprendizaje profundo suelen ser considerados ininteligibles, incluso coloquialmente llamados *cajas negras*, dado que la interpretación de su

comportamiento y de su proceso de aprendizaje no es directa. En este sentido, nuestro modelo  $MTL-DNN_{Cons}$  no proporciona medios a  $lx$  expertx para interpretar la razón por la que un compuesto se predice como mutagénico o no. A pesar de ello, si un compuesto es detectado como mutagénico por nuestro modelo, es posible conocer los valores de predicción de cada una de las cinco cepas, lo que aporta información adicional que contribuye a la interpretabilidad de los resultados de mutagenicidad de Ames en contraste con los modelos tradicionales de tarea única.

### 5.5.1. Resultados obtenidos mediante el enfoque de etiquetado *laxo*

Los resultados obtenidos por medio del enfoque de etiquetado laxo se encuentran en la tabla 5.6. Al comparar las métricas obtenidas para los modelos  $MTL-DNN_{Cons}$  y  $STL-DNN_{Overall}$  para este enfoque de etiquetado global, en general, el modelo  $MTL-DNN_{Cons}$  tuvo un rendimiento superior. Los valores de  $Sp$  y  $Sn$  están equilibrados para el modelo  $MTL-DNN_{Cons}$ , lo que indica que, si bien el modelo  $STL-DNN_{Overall}$  predice mejor la clase *positiva*, posiblemente esto ocurra a costa de una menor capacidad de generalización. Estos resultados no difieren significativamente de los obtenidos con el enfoque de etiquetado conservador adoptado en nuestra propuesta. En este sentido, hemos priorizado el enfoque conservador tanto en este capítulo como en nuestro trabajo publicado [248], ya que es el que asegura que un compuesto etiquetado como no mutagénico ha sido probado en todas las cepas.

	Sp	Sn	Precisión	Acc	BAcc	Puntaje F1	Puntaje H1
$MTL-DNN_{Cons}$	<b>0,86 ± 0,04</b>	0,81 ± 0,01	<b>0,99 ± 0,00</b>	0,82 ± 0,01	<b>0,84 ± 0,02</b>	0,89 ± 0,00	<b>0,84 ± 0,02</b>
$STL-DNN_{Overall}$	0,43 ± 0,06	<b>0,98 ± 0,00</b>	0,96 ± 0,00	<b>0,94 ± 0,00</b>	0,70 ± 0,03	<b>0,97 ± 0,00</b>	0,60 ± 0,06

Tabla 5.6: Resultados en la partición de validación externa, obtenidos adoptando un enfoque de etiquetado global *laxo* en los modelos  $MTL-DNN_{Cons}$  y  $STL-DNN_{Overall}$ .

## 5.6. Síntesis y conclusiones

La prueba de Ames es uno de los métodos más utilizados para detectar mutagenicidad de compuestos químicos. Actualmente, el desarrollo de modelos *in silico* para la predicción de mutagenicidad es un campo de investigación altamente desarrollado y activo; sin embargo, la vasta mayoría de los modelos QSAR encontrados en la literatura utilizan etiquetas computadas a partir de información global de mutagenicidad, distinguiendo únicamente entre las categorías *mutagénico* y *no mutagénico* sin considerar los resultados intermedios obtenidos individualmente para

experimentos realizados en diversas cepas de *Salmonella typhimurium*. Estos modelos suelen presentar un rendimiento desequilibrado en términos de *Sensibilidad* ( $Sn$ ), es decir, la capacidad de predecir correctamente compuestos mutagénicos, y *Especificidad* ( $Sp$ ), es decir, la capacidad de predecir correctamente compuestos no mutagénicos. En el costoso y complejo proceso de descubrimiento de fármacos, así como para las regulaciones alimentarias y ambientales, resulta esencial desarrollar modelos QSAR capaces de predecir mutagenicidad con alta sensibilidad y especificidad.

En este capítulo, presentamos nuestra propuesta de un modelo novedoso para predecir mutagenicidad de Ames utilizando estrategias de aprendizaje profundo y un enfoque de aprendizaje multi-tarea (MTL), empleando información experimental de cinco cepas de *Salmonella typhimurium*: TA98, TA100, TA102, TA1535 y TA1537. Nuestro modelo permite predecir la mutagenicidad de un compuesto químico en cada cepa por separado, a la vez que aprende conjuntamente información compartida por todas las cepas. De esta manera, la predicción de la mutagenicidad de Ames se obtiene unificando los resultados correspondientes a cada una de las cepas objetivo por medio de un consenso. Según nuestra evaluación del estado del arte, la estrategia de aprendizaje MTL no ha sido aplicada previamente al modelado de pruebas de Ames, por lo que nuestro enfoque resulta un aporte novedoso.

Los resultados obtenidos por nuestro modelo MTL superan a aquellos obtenidos por los modelos de tarea única, es decir, entrenados para predecir la etiqueta *Overall* de la prueba de Ames y también a aquellos asociados al modelado de la mutagenicidad por medio de un consenso de modelos de cepas individuales. Nuestro enfoque MTL presenta un rendimiento equilibrado en términos de  $Sn$  y  $Sp$ , lo que significa que es capaz de detectar de forma precisa compuestos mutagénicos y no mutagénicos. Estos resultados respaldan nuestra hipótesis de que el aprendizaje multi-tarea es beneficioso para el modelado QSAR, dado que permite aprender en entornos complejos, con poca información y escenarios de fuerte desbalance de clases. Finalmente, el modelado multi-tarea destaca por sobre los enfoques de aprendizaje basados en tarea única por cuanto proporcionan una predicción para cada cepa, lo cual favorece la interpretabilidad de la predicción de mutagenicidad de un compuesto.

Independientemente del enfoque de aprendizaje utilizado en el proceso de modelado, los modelos QSAR se entrenan empleando representaciones moleculares que permitan capturar múltiples aspectos estructurales y físico-químicos de los compuestos. Tradicionalmente, y en sintonía con nuestras propuestas planteadas en los capítulos 4 y 5 de la presente tesis, las dos representaciones moleculares predominantes en la bibliografía han sido los descriptores moleculares y los *fingerprints de conectividad extendida* (ECFP), los cuales pueden ser calculados empleando herramientas de software especializadas y por medio de algoritmos ampliamente conocidos y estudiados. No obstante, durante



los últimos años la comunidad científica ha manifestado un creciente interés en la exploración y desarrollo de nuevas representaciones moleculares, a menudo empleando estrategias de aprendizaje profundo. En este contexto, resulta de profundo interés analizar distintos tipos de representación molecular y, particularmente, evaluar su desempeño en modelado QSAR, a fin de identificar potenciales representaciones capaces de mejorar el estado del arte en la predicción de propiedades relevantes para el desarrollo de fármacos. En el capítulo 6 presentamos un trabajo de revisión y análisis exhaustivos de una amplia variedad de representaciones moleculares y su aplicación en el desarrollo de modelos de regresión y clasificación para ocho propiedades de relevancia en quimioinformática.

# Capítulo 6

## Evaluación de representaciones moleculares

La representación molecular empleada para una tarea determinada en el proceso de desarrollo de fármacos resulta determinante en los resultados obtenidos. En los últimos años, los avances en aprendizaje profundo han propiciado el desarrollo de nuevas estrategias de representación molecular, denominados *embeddings moleculares*. En este capítulo desarrollamos una revisión minuciosa de distintos enfoques de representación molecular y los sometemos a un análisis comparativo en el contexto del modelado QSAR para ocho propiedades físico-químicas.

---

### 6.1. Representaciones moleculares en diseño de fármacos

Tal y como hemos expuesto en capítulos previos de la presente tesis, el modelado QSAR constituye una de las piedras angulares del proceso de descubrimiento de fármacos *in silico* [432, 433, 68]. Los modelos QSAR generalmente se entrenan utilizando representaciones moleculares tradicionales, tales como descriptores moleculares y *fingerprints ECFP*, las cuales se obtienen por medio de algoritmos ampliamente estudiados en la literatura [378]. Si bien el uso de representaciones moleculares tradicionales para modelado QSAR está bien establecido y, a menudo, los modelos QSAR entrenados a partir de dichas representaciones exhiben buenos resultados, durante los últimos años la comunidad científica ha manifestado un creciente interés en explorar nuevas representaciones moleculares enriquecidas, obtenidas a partir de algoritmos de aprendizaje automático, los cuales

permiten extraer aspectos de la estructura molecular y sus propiedades físico-químicas potencialmente ocultos o difícilmente identificables por medio de algoritmos tradicionales [73].

Con la consolidación del aprendizaje profundo en el área de diseño de fármacos, durante los últimos años se han propuesto varios algoritmos destinados al aprendizaje de representaciones moleculares novedosas, denominadas *embeddings* moleculares. Como ilustra la figura 6.1, los *embeddings* moleculares suelen ser vectores densos de números reales de dimensionalidad fija, mientras que las representaciones moleculares tradicionales pueden consistir en vectores de números reales o vectores de bits de distinta densidad (cantidad de unos y ceros). Estas representaciones son luego suministradas al modelo QSAR de clasificación o regresión para su entrenamiento. Los *embeddings* moleculares pueden ser obtenidos por medio de distintas estrategias de entrenamiento, permitiendo capturar información diversa sobre las propiedades físico-químicas, estructura molecular y perfiles de bioactividad de los compuestos candidatos.

Con respecto a su proceso de entrenamiento, los algoritmos para obtener *embeddings* moleculares pueden ser *supervisados*, lo que significa que tienen en cuenta las etiquetas de bioactividad del conjunto de moléculas empleado durante el entrenamiento, o *no supervisados*, lo que implica que las representaciones se construyen sin información sobre el perfil farmacológico de los compuestos empleados en el entrenamiento. Si bien los *embeddings* supervisados involucran información adicional en el proceso de aprendizaje de las representaciones, lo cual podría favorecerlos posteriormente en tareas predictivas, resultan en consecuencia menos flexibles que los *embeddings* no supervisados, ya que es necesario aprender nuevas representaciones para cada propiedad a predecir. Por otra parte, para el aprendizaje de *embeddings* supervisados es necesario tener conjuntos de datos etiquetados, que a menudo son pequeños y escasos, lo que puede afectar negativamente la calidad de los *embeddings* resultantes.

Por estas razones, los métodos de *embeddings* no supervisados resultan atractivos en la comunidad científica, puesto que pueden ser adaptados para aprovechar grandes volúmenes de información molecular no etiquetada, beneficiándose así de la diversidad química presente en grandes conjuntos de compuestos. En particular, las estrategias de aprendizaje automático desarrolladas para procesamiento de lenguaje natural (NLP, por sus siglas en inglés) resultan especialmente prometedoras, por cuanto permiten el aprendizaje directo a partir de la molécula en formato SMILES [418]. El formato SMILES, como se ha discutido en el capítulo 3, es el formato más ampliamente utilizado para almacenar y distribuir información molecular, además de ser el formato predominante en las grandes quimiotecas públicamente accesibles, que cuentan con millones de compuestos no etiquetados.

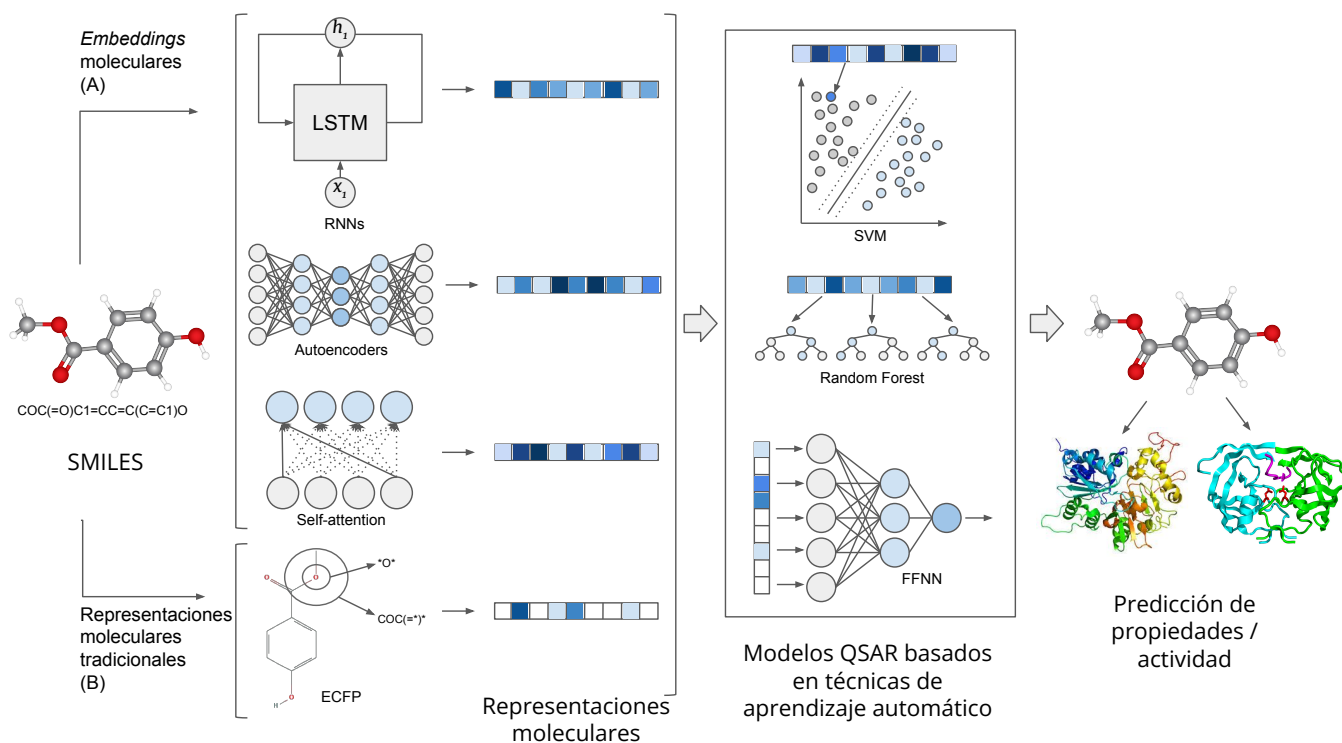


Figura 6.1: Los modelos QSAR son modelos de clasificación o regresión, dependiendo de la propiedad a predecir, y pueden entrenarse usando *embeddings* moleculares aprendidos por medio de técnicas de aprendizaje profundo (A), o empleando representaciones moleculares tradicionales (B). Los *embeddings* moleculares son vectores densos de números reales, mientras que las representaciones moleculares tradicionales pueden ser vectores de números reales o vectores de bits que varían en densidad (cantidad de unos y ceros).

Los algoritmos para el aprendizaje de *embeddings* moleculares emplean una amplia variedad de técnicas de aprendizaje profundo de última generación [103, 66], que van desde *autoencoders* basados en redes neuronales, hasta redes neuronales basadas en grafos (GNNs por sus siglas en inglés) [50, 431] y *Self-Attention* [397]. En particular, para el dominio del desarrollo de representaciones moleculares, recientemente se han realizado notables esfuerzos de investigación en el diseño de técnicas novedosas para el aprendizaje de *embeddings* moleculares empleando técnicas de NLP [73, 103, 84]. Los algoritmos de *embeddings* varían en complejidad, lo que permite elegir y adaptar las representaciones moleculares a tareas específicas. Por ejemplo, muchos algoritmos recientes basados en *Self-Attention* para obtener *embeddings* moleculares están especialmente diseñados con el objeto de identificar las subestructuras moleculares que tienen un impacto significativo en el perfil de bioactividad de un compuesto determinado [287, 449]. Hoy en día, numerosos estudios

QSAR emplean *embeddings* moleculares en lugar de representaciones tradicionales como formato de representación de los conjuntos de compuestos químicos de entrenamiento [73, 430, 182]. Sin embargo, se han publicado algunos estudios en los que los *embeddings* moleculares no exhiben diferencias significativas de desempeño en relación con las representaciones tradicionales en el modelado QSAR [182, 176, 124, 122, 440, 439, 69].

Como hemos discutido en capítulos anteriores, los beneficios teóricos de desarrollar nuevos métodos de representación molecular basados en aprendizaje profundo, sumados al creciente interés de la comunidad en este tópico, han desencadenado una proliferación de dichos métodos en la literatura científica. No obstante, la investigación empírica sobre cómo elegir el método de representación más adecuado para modelado QSAR y otras tareas en el complejo proceso de desarrollo de fármacos es aún incipiente. Esto se debe en parte a que realizar una comparación entre distintos *embeddings* no resulta una tarea sencilla, dado que se deben considerar múltiples aspectos inherentes al proceso de aprendizaje de dichas representaciones.

Una comparación justa y exhaustiva de los diferentes tipos de representación molecular existentes implica la realización de numerosos experimentos en varios conjuntos de datos y escenarios de entrenamiento. Si bien establecer una comparación justa entre diferentes representaciones moleculares no es sencillo, argumentamos que dicho análisis es necesario y que debe llevarse a cabo a través de un diseño experimental cuidadoso. En este capítulo, presentamos un amplio análisis comparativo de diversas técnicas de *embeddings* moleculares basadas en aprendizaje profundo, contrastadas a su vez con representaciones moleculares tradicionales, centrándonos en su idoneidad para el modelado QSAR [324]. Como parte de nuestro análisis, llevamos a cabo una revisión exhaustiva de la literatura sobre métodos de representación molecular y seleccionamos tres técnicas de *embeddings no supervisadas* y dos técnicas *supervisadas* destacadas. Reprodujimos estas cinco técnicas y comparamos su desempeño en distintos escenarios de modelado QSAR, empleando diversos conjuntos de datos vinculados a tareas de clasificación y regresión. También comparamos estos cinco *embeddings* con tres representaciones moleculares tradicionales, a saber, descriptores moleculares, *fingerprints ECFP* y claves *MACCS*. Nuestro trabajo abordó los siguientes objetivos de investigación:

- Determinar cuáles son los principales métodos de *embeddings* moleculares utilizados para el modelado QSAR en la literatura, y si los mismos superan o no a las representaciones moleculares tradicionales en tareas de clasificación/regresión.

- Comparar el desempeño predictivo de los *embeddings* supervisados con el de los *embeddings* no supervisados, a fin de evaluar el impacto en el modelado QSAR de incorporar información sobre la propiedad objetivo en el proceso de representación molecular.
- Estudiar el impacto en el desempeño predictivo de diferentes decisiones de diseño de las técnicas de *embeddings*, tales como la forma canónica de las fórmulas SMILES utilizadas o la dimensionalidad de los *embeddings* finales.

Finalmente, contrastamos el desempeño alcanzado por los *embeddings* moleculares estudiados con los resultados de referencia obtenidos utilizando representaciones moleculares tradicionales. Nuestro diseño experimental consistió en el desarrollo de más de 25.000 modelos entrenados, comprendiendo varias etapas de selección y réplicas de dichos modelos, seguidas de análisis de significancia estadística de los resultados.

## 6.2. Trabajo relacionado con la propuesta

Más allá del conjunto de técnicas de aprendizaje automático empleadas en una tarea de modelado particular, la confiabilidad de los modelos QSAR y su desempeño predictivo están relacionados con la elección de la representación molecular utilizada para su entrenamiento [73, 323]. Históricamente, la práctica estándar ha consistido en diseñar manualmente representaciones moleculares de alta calidad. Este proceso, conocido como *ingeniería de atributos*, ha dado lugar a representaciones moleculares de uso masivo, hoy en día consideradas representaciones tradicionales en el diseño de fármacos [378, 62]. Tal y como hemos expuesto en el capítulo 3 de la presente tesis doctoral, las representaciones moleculares tradicionales varían principalmente en el tipo de información que codifican y su uso depende de la tarea específica a desarrollar [134, 332]. Entre las representaciones moleculares tradicionales más utilizadas en modelado QSAR, podemos nombrar a los descriptores moleculares [378], los *fingerprints de conectividad extendida (ECFPs)* [311], y las claves *MACCS (Molecular ACCess System)* [99].

Si bien muchos de los modelos QSAR publicados en la literatura se basan en estas representaciones tradicionales [343, 441, 113, 322], la obtención de tales representaciones a través de un proceso de ingeniería de atributos es costoso y requiere una gran experiencia en un dominio específico, sobre todo cuando se trata de la selección de descriptores moleculares adecuados para el modelado predictivo de una propiedad determinada. Además, dado que cada una de dichas representaciones codifica información diferente sobre la molécula, no existe una única representación adecuada para cada

tarea. Por estas razones, durante los últimos años se ha observado una tendencia creciente en la comunidad científica hacia el uso de representaciones moleculares versátiles, que capturen diversos aspectos del espacio químico [73], los cuales potencialmente enriquezcan el proceso de aprendizaje del modelo QSAR. En este contexto, se han desarrollado muchos métodos novedosos para aprender *embeddings* moleculares, la mayoría de ellos basados en técnicas de aprendizaje profundo [73, 103].

Los métodos para aprender *embeddings* moleculares más frecuentemente hallados en la literatura emplean fórmulas SMILES [418] como representación base de los compuestos químicos, que es la representación lineal más utilizada para codificar información de la estructura molecular de los compuestos. Dado que las fórmulas de SMILES codifican directamente el grafo molecular por medio de una secuencia de caracteres ASCII, pueden fácilmente ser procesadas por medio de técnicas de aprendizaje profundo diseñadas para datos secuenciales, cadenas de caracteres, o estructuras que puedan ser representadas por medio de grafos [103, 182]. Dada su alta disponibilidad—la mayoría de las bases de datos moleculares se almacenan en formato SMILES por ser un formato flexible y eficiente en términos de sus requerimientos de almacenamiento—, se han desarrollado y entrenado muchos métodos no supervisados para aprender *embeddings* moleculares a partir de grandes bases de datos sin necesidad de contar con etiquetas de clase o información sobre el perfil de bioactividad de los compuestos [176, 229, 365, 290, 436]. Si bien existen antecedentes del uso de imágenes o representaciones gráficas estructuradas como representaciones base de los compuestos químicos para aprender *embeddings* moleculares en modelado QSAR [122, 210, 348], su uso no está tan extendido y tiene limitaciones en términos de la disponibilidad de datos. Por lo tanto, en la propuesta presentada en este capítulo nos enfocamos en los métodos de *embeddings* moleculares entrenados a partir de fórmulas SMILES.

La alta disponibilidad de datos moleculares en formato SMILES ha motivado numerosos enfoques para el aprendizaje de *embeddings* moleculares basados en *autoencoders*, que permiten realizar procedimientos de entrenamiento no supervisados [176, 124, 436, 289], tal y como se explica en el capítulo 2. En particular, Öztürk et al. [290] introdujeron *SMILESVec*, un método no supervisado que aprende representaciones de moléculas pequeñas utilizando el popular modelo *word2vec* de Mikolov et al. [258]. Lxs autorxs tomaron un conjunto de compuestos representados en formato SMILES y realizaron un paso de *tokenización* sobre dichas cadenas: convirtieron cada fórmula SMILES en una secuencia de *tokens* o elementos distinguibles en un alfabeto mediante la extracción de subcadenas de SMILES solapadas, que luego fueron utilizadas para entrenar su modelo. Finalmente, calcularon un *embedding* para cada molécula en el conjunto de datos, promediando los vectores aprendidos para cada uno de los *tokens* presentes en ella.

Por su parte, Jaeger et al. [176] presentaron *Mol2Vec*, que también se basa en el clásico modelo *word2vec* [258]. En este método, las fórmulas SMILES se someten a una etapa de preprocesamiento antes de ser suministradas como entrada al *autoencoder*, la cual consiste en un paso de tokenización de las cadenas SMILES utilizando el algoritmo para computar *fingerprints ECFP* [311]. Tras una fase de entrenamiento no supervisada, las representaciones finales se calculan sumando los vectores obtenidos de todos los *tokens* en la molécula.

Dado que la notación SMILES es de naturaleza secuencial, por cuanto es una traducción del recorrido del grafo molecular del compuesto a una secuencia de caracteres, también se encuentran en la literatura métodos para generar nuevas moléculas o aprender nuevos *embeddings* moleculares basados en redes neuronales recurrentes (RNNs) [436, 341, 301]. Muchos modelos generativos en la literatura se basan en *autoencoders* construidos a partir de RNNs, entrenados a partir de grandes conjuntos de datos no etiquetados [103], lo que constituye un enfoque interesante para el aprendizaje de representaciones moleculares. En esta dirección, Xu et al. [436] propusieron *Seq2Seq Fingerprint*, un método no supervisado basado en un *autoencoder* RNN multicapa construido usando *Gated Recurrent Units (GRU)* [70]. *Seq2Seq Fingerprint* aprende directamente de las fórmulas SMILES aprovechando la información de las dependencias y de la relación secuencial entre los caracteres de la cadena SMILES. Los *embeddings* moleculares se obtienen finalmente concatenando los estados internos de las capas ocultas del *autoencoder*.

Una tendencia relativamente reciente en el desarrollo de técnicas de *embeddings* moleculares es el uso del mecanismo de *Self-Attention* [397]. Tal y como desarrollamos en el capítulo 2, el mecanismo de *Self-Attention* resulta interesante para el aprendizaje de representaciones moleculares, puesto que permite capturar información valiosa sobre las subestructuras relevantes de la molécula y sus interdependencias, evitando los largos procesos de entrenamiento característicos de las RNN [397] y a la vez aportando un sentido de interpretabilidad a los resultados en términos químicos. En esta dirección, Oskooei et al. [287] desarrollaron *PaccMann*, un enfoque novedoso para predecir la sensibilidad a compuestos anticancerígenos por medio de redes neuronales multimodales basadas en el mecanismo de *attention*. Lxs autorxs propusieron una serie de *encoders* supervisados; el que mejor se desempeñó fue un *encoder* basado en *Self-Attention* entrenado usando una tokenización simple de cadenas SMILES y complementado con información de expresión génica. El modelo comprende una red neuronal multicapa poco profunda para la predicción de la propiedad de interés, la cual está directamente conectada al modelo neuronal con mecanismo de *Self-Attention* de forma tal que los *embeddings* moleculares son aprendidos de forma conjunta con el clasificador o modelo de regresión durante la fase de entrenamiento y, por ende, pueden ser categorizados como *embeddings* supervisados.



Por su parte, Zheng et al. [449] desarrollaron *SA-BiLSTM*, un método supervisado basado en una combinación de una red neuronal recurrente *Long-Short Term Memory (LSTM)* [159] y un mecanismo de *Self-Attention*. Al igual que en el caso de *PaccMann*, el modelo se conecta a una red neuronal multicapa poco profunda para modelado QSAR, por lo que los *embeddings* son aprendidos junto con la tarea de predicción. Un aspecto interesante de *SA-BiLSTM* es que no es entrenado directamente a partir de las cadenas SMILES de los compuestos químicos, sino que utiliza secuencias de *tokens* codificados por medio de *embeddings* obtenidos de un modelo *Mol2Vec* previamente entrenado [176].

Basándonos en una extensa revisión bibliográfica y con el objeto de abarcar una diversidad de técnicas, teniendo en cuenta diferentes estrategias de entrenamiento y arquitecturas de aprendizaje profundo, los cinco métodos aquí descritos fueron los elegidos para nuestro análisis comparativo de representaciones moleculares, junto con tres representaciones tradicionales. En las secciones subsiguientes se abordan los aspectos metodológicos de la propuesta, junto con los resultados observados y su discusión.

## 6.3. Metodología

Bajo la premisa de la importancia de un análisis comparativo detallado y cuidadoso de las representaciones moleculares disponibles para modelado QSAR, uno de los objetivos primordiales de nuestra propuesta fue garantizar la reproducibilidad de los experimentos. En esta sección proporcionamos una explicación detallada de nuestro diseño experimental y una descripción general de los conjuntos de datos empleados en el análisis comparativo y de su etapa de preprocesamiento. Además, detallamos las arquitecturas de red neuronal de los cinco métodos de *embeddings* moleculares reproducidos y la configuración empleada para su entrenamiento. Tanto los conjuntos de datos, como nuestro código fuente y los resultados de todos los pasos intermedios de la fase experimental son públicamente accesibles<sup>1</sup>.

### 6.3.1. Conjuntos de datos

Siguiendo los pasos necesarios para reproducir las técnicas no supervisadas, tal y como fueron detallados por los autorxs de las técnicas en cuestión [176, 290, 436], recopilamos y descargamos las fórmulas SMILES de aproximadamente 200 millones de compuestos adquiribles de la base de datos

---

<sup>1</sup>Recursos: [https://csunseduar-my.sharepoint.com/:f:/g/personal/virginia\\_sabando\\_cs\\_uns\\_edu\\_ar/EjUkG4X2A31EgJOAj0EjveYBMcoo08mKIpQoHquoQtdUhw](https://csunseduar-my.sharepoint.com/:f:/g/personal/virginia_sabando_cs_uns_edu_ar/EjUkG4X2A31EgJOAj0EjveYBMcoo08mKIpQoHquoQtdUhw)

ZINC [360]. Realizamos una etapa de preprocesamiento que consistió en filtrar los compuestos que no cumplieran con la *Regla de los Cinco de Lipinski* [227]: únicamente mantuvimos aquellos compuestos que tuvieran un peso molecular entre 12 y 600 Da., un recuento de átomos pesados entre 3 y 50, y un coeficiente de partición octanol/agua (cLogP) entre  $-5$  y  $7$ . También filtramos los compuestos que presentaron elementos atípicos en fármacos, tales como metales pesados, y eliminamos sales y solventes. Finalmente, obtuvimos fórmulas SMILES canónicas para cada uno de los compuestos restantes. Todo este proceso de sanitización se llevó a cabo utilizando herramientas provistas por la librería RDKit [211]. Después de la etapa de preprocesamiento, seleccionamos aleatoriamente un subconjunto de 40 millones de compuestos, el doble de los compuestos que se usaron en los trabajos de referencia [176, 290, 436], los cuales fueron luego utilizados para entrenar los métodos de *embeddings* no supervisados.

También seleccionamos ocho conjuntos de datos etiquetados diferentes, cinco correspondientes a tareas de clasificación y tres correspondientes a tareas de regresión. Estos conjuntos de datos se utilizaron para entrenar los dos métodos de *embeddings* supervisados, en particular, y para evaluar tanto métodos supervisados como no supervisados de *embedding* en tareas de predicción QSAR. Todos los conjuntos de datos de clasificación plantean tareas de clasificación binaria. Priorizamos los conjuntos de datos que se habían utilizado inicialmente en los trabajos de referencia [287, 449, 176, 290, 436], a la vez teniendo en cuenta la diversidad de tamaños de conjuntos de datos y escenarios de desequilibrio de clases. Los conjuntos de datos seleccionados se procesaron siguiendo el mismo procedimiento utilizado para ZINC, descrito anteriormente. En la tabla 6.1 brindamos detalles cuantitativos sobre los conjuntos de datos empleados. A saber, estos conjuntos de datos son los siguientes:

- *SR-ARE*, un ensayo para agonistas de moléculas pequeñas de la vía de señalización del *elemento de respuesta antioxidante* (ARE)<sup>2</sup>. Este conjunto de datos forma parte del conjunto de datos del *Desafío Tox21*, que consta de mediciones cualitativas de toxicidad en doce objetivos biológicos<sup>3</sup>.
- *SR-MMP*, un ensayo de respuesta al estrés para disruptores de moléculas pequeñas del *potencial de membrana mitocondrial* (MMP)<sup>4</sup>. Este conjunto de datos también se incluye en el conjunto de datos del *Desafío Tox21*.

---

<sup>2</sup><https://pubchem.ncbi.nlm.nih.gov/bioassay/743219>

<sup>3</sup><https://tripod.nih.gov/tox21/challenge/about.jsp>

<sup>4</sup><https://pubchem.ncbi.nlm.nih.gov/bioassay/720637>

- *SR-ATAD5*, un conjunto de pequeñas moléculas que inducen genotoxicidad en células de riñón embrionario humano que expresan ATAD5 marcado con luciferasa<sup>5</sup>. Este conjunto de datos también pertenece al conjunto de datos del *Desafío Tox21*.
- *HIV*, un conjunto de datos introducido por el programa *Drug Therapeutics Program AIDS Antiviral Screen* que contiene información sobre la capacidad molecular para inhibir la replicación del *HIV*. Los resultados del cribado fueron clasificados como *inactivos confirmados (CI)*, *activos confirmados (CA)* y *moderadamente activos confirmados (CM)*. Fusionamos los compuestos categorizados bajo las últimas dos etiquetas, lo que dio lugar a dos clases: *inactivo (CI)* y *activo (CA y CM)*<sup>6</sup>.
- *PCBA-686978*, un bioensayo recuperado de la base de datos pública PubChem que contiene información sobre la capacidad molecular para inhibir la tirosil-ADN fosfodiesterasa 1 humana<sup>7</sup>.
- *ESOL*, un conjunto de datos que consta de datos de solubilidad en agua para 1.128 compuestos, utilizado para entrenar modelos que estiman la solubilidad directamente a partir de estructuras químicas [87].
- *FreeSolv*, la base de datos de solvatación libre (*Free Solvation Database*)—un conjunto de datos que comprende valores de energía libre de hidratación experimentales y calculados para moléculas pequeñas en agua [260].
- *Lipophilicity*, un conjunto de datos seleccionado de la base de datos pública ChEMBL [41] que incluye resultados experimentales del coeficiente de distribución octanol/agua (*LogD 7,4*) para 4.200 compuestos<sup>8</sup>.

### 6.3.2. Métodos de representación molecular

Tal y como explicamos anteriormente, luego de identificar las técnicas para extracción de *embeddings* moleculares más destacadas en la bibliografía, elegimos *SMILESVec* [290], *Mol2Vec* [176], *Seq2Seq Fingerprint* [436]—en adelante denominada *Seq2Seq* a lo largo del capítulo—, el codificador basado en *Self-Attention* de *PaccMann* [287]—denominado *PaccMann* a lo largo del

---

<sup>5</sup><https://pubchem.ncbi.nlm.nih.gov/bioassay/720516>

<sup>6</sup><https://wiki.nci.nih.gov/display/NCIDTPdata/AIDS+Antiviral+Screen+Datos>

<sup>7</sup><https://pubchem.ncbi.nlm.nih.gov/bioassay/686978>

<sup>8</sup>[https://www.ebi.ac.uk/chembl/document\\_report\\_card/CHEMBL3301361/](https://www.ebi.ac.uk/chembl/document_report_card/CHEMBL3301361/)

Conjunto de datos	Tarea	# Compuestos	# Activos	# Inactivos	Tasa de desbalance
SR-ARE	clasificación	5.956	941	5.015	18,76
SR-MMP	clasificación	5.937	925	5.012	3,68
SR-ATAD5	clasificación	7.251	258	6.993	18,46
HIV	clasificación	41.127	1.443	39.684	27,50
PCBA-686978	clasificación	302.175	62.800	239.375	3,81
ESOL	regresión	1.128	-	-	-
FreeSolv	regresión	642	-	-	-
Lipophilicity	regresión	4.200	-	-	-

Tabla 6.1: Detalles cuantitativos de los conjuntos de datos etiquetados. La tasa de desbalance de clases para los conjuntos de datos asociados a tareas de clasificación fue computada como el número de compuestos activos cada 100 compuestos inactivos.

presente capítulo—, y el codificador *SA-BiLSTM* [449] como nuestros métodos de referencia. Seleccionamos estos métodos con el fin de evaluar diferentes arquitecturas neuronales y estrategias de aprendizaje profundo, al mismo tiempo ponderando el impacto del tamaño del modelo en términos de sus parámetros entrenables y considerando escenarios de entrenamiento tanto supervisados como no supervisados. Realizamos una fase de entrenamiento que consistió en múltiples ejecuciones de cada método para tener en cuenta su varianza intrínseca en el análisis de los resultados. Para ello, probamos una amplia gama de combinaciones de hiperparámetros para cada técnica de *embeddings* y modelo de clasificación/regresión, a la vez empleando diferentes formas canónicas de las fórmulas SMILES. Todos los métodos de *embeddings* no supervisados se entrenaron a partir de los 40 millones de compuestos recuperados y seleccionados del conjunto de datos ZINC, mientras que los métodos supervisados se entrenaron con cada uno de los ocho conjuntos de datos etiquetados descritos anteriormente. A continuación, describimos las arquitecturas neuronales de cada método de *embeddings* y las principales características de las representaciones moleculares obtenidas.

### 6.3.2.1. Métodos de representación molecular no supervisados

La arquitectura de *SMILESVec* [290] consta de un *autoencoder* basado en una red neuronal multicapa <sup>9</sup>, según lo propuesto por Mikolov et al. [258]. Siguiendo el artículo original, primero tokenizamos las fórmulas SMILES en subcadenas superpuestas de ocho caracteres y compilamos un

<sup>9</sup><https://github.com/hkmztrk/SMILESVecProteinRepresentation/tree/master/source/word2vec>

alfabeto comprendiendo todos los *tokens* extraídos del conjunto de entrenamiento. Considerando las interdependencias entre caracteres o *tokens* usualmente presentes en las secuencias SMILES, tal y como ocurre con las ramificaciones o con la información estereoquímica y de aromaticidad en las estructuras moleculares, y que la tokenización de 8-gramas originalmente propuesta por lxs autorxs podría limitar la capacidad del modelo para identificarlas, empleamos dos procesos de canonicalización diferentes en los datos de entrenamiento de ZINC. De esta forma, obtuvimos dos conjuntos de entrenamiento diferentes para este método. Por un lado, empleamos SMILES canónicos de RDKit [211], mientras que, por otro lado, probamos las fórmulas canónicas de *DeepSMILES* [81], las cuales se construyen eliminando ciclos en el grafo molecular a fin de reducir las interdependencias de largo rango en las secuencias SMILES. *DeepSMILES* también había sido empleado por lxs autorxs que propusieron *SMILESVec* en un artículo posterior al de referencia [289], exhibiendo un mejor desempeño con respecto a los resultados obtenidos con las fórmulas canónicas de RDKit [211].

Tal y como propusieron lxs autorxs de *SMILESVec* [290], se utilizó un modelo *word2vec Skip-gram* [258], en el que la dimensionalidad de los *embeddings* finales está determinado por el número de nodos en su capa oculta. Entrenamos dos modelos empleando *embeddings* de diferentes dimensionalidades para cada una de las técnicas de canonicalización mencionadas anteriormente: *embeddings* de 100 dimensiones, siguiendo el artículo original, y de 300 dimensiones, a fin de comparar *SMILESVec* en mejores condiciones con *embeddings* de mayores dimensionalidades obtenidos por medio de otros métodos considerados en este capítulo. Siguiendo el trabajo realizado por Öztürk et al. [290], entrenamos *SMILESVec* durante 20 épocas y las representaciones moleculares finales se obtuvieron calculando el promedio de los vectores aprendidos para cada token en las fórmulas SMILES.

*Mol2Vec* [176] es también un método de *embeddings* moleculares no supervisados basado en una arquitectura *word2vec* [258]. La arquitectura del modelo desarrollado por lxs autorxs <sup>10</sup> consiste en un modelo *word2vec Skip-gram*, donde la dimensionalidad del *embedding* coincide con la cantidad de nodos neuronales de la capa oculta del *autoencoder*. Siguiendo los pasos de preprocesamiento indicados por Jaeger et al. [176], tokenizamos las fórmulas SMILES canónicas de RDKit del conjunto de datos ZINC en secuencias de palabras químicas obtenidas empleando una adaptación del algoritmo de Morgan para calcular *fingerprints ECFP* [99]. En el caso de *Mol2Vec* no empleamos canonicalización *DeepSMILES*, ya que el algoritmo de Morgan no es capaz de reconstruir el grafo molecular de un compuesto a partir de dicha forma canónica. Siguiendo el artículo original, compilamos un alfabeto incluyendo todas las palabras químicas obtenidas del conjunto de datos de entrenamiento ZINC y entrenamos dos modelos por 5 épocas cada uno, con *embeddings* de 100 y 300

---

<sup>10</sup><https://github.com/samoturk/mol2vec>

dimensiones, respectivamente. Finalmente, tal y como lo hicieron lxs autorxs del trabajo original, obtuvimos los *embeddings* moleculares calculando la suma de los vectores aprendidos de todos los tokens en cada molécula.

El tercer y último método no supervisado que reproducimos es *Seq2Seq* [436], un modelo de *embeddings* que consta de un *autoencoder* basado en RNNs <sup>11</sup>. Computamos fórmulas SMILES canónicas de ambas formas (RDKit y *DeepSMILES*) para el conjunto de entrenamiento ZINC y luego las tokenizamos separándolas en caracteres individuales. La arquitectura del modelo propuesto por Xu et al. [436] consiste en un *autoencoder GRU* bidireccional multicapa y los *embeddings* se obtienen concatenando los estados ocultos del modelo entrenado. Por lo tanto, su dimensionalidad está determinada por el número de unidades ocultas y el número de capas de *autoencoder*. Entrenamos dos modelos diferentes para cada una de las formas canónicas de las cadenas SMILES, obteniendo *embeddings* de 100 y 384 dimensiones en cada caso, siendo esta última la dimensionalidad de las representaciones computadas por lxs autorxs del método. En la tabla 6.2 proporcionamos un resumen de los *embeddings* moleculares obtenidos a partir de todas las variantes mencionadas de los métodos no supervisados abordados en nuestro trabajo.

Modelo	Canonicalización	Dimensionalidad de los <i>embeddings</i>	Denominación
SMILESVec	RDKit	100	SMILESVec_100
		300	SMILESVec_300 (*)
	DeepSMILES	100	Deep_SMILESVec_100
		300	Deep_SMILESVec_300 (*)
Mol2Vec	RDKit	100	Mol2Vec_100
		300	Mol2Vec_300
Seq2Seq	RDKit	100	Seq2Seq_100 (*)
		384	Seq2Seq_384
	DeepSMILES	100	Deep_Seq2seq_100 (*)
		384	Deep_Seq2seq_384 (*)

Tabla 6.2: Resumen de los modelos no supervisados de *embeddings* moleculares. La columna *Denominación* denota el nombre que empleamos para referirnos a cada una de las representaciones obtenidas por los métodos en cuestión a lo largo del presente capítulo. Aquellas denominaciones marcadas con un asterisco (\*) corresponden a *embeddings* que decidimos computar en adición a los originalmente propuestos en los artículos de referencia.

<sup>11</sup><https://github.com/XericZephyr/seq2seq-fingerprint>

### 6.3.2.2. Métodos de representación molecular supervisados

*PaccMann* [287] es un método de *embedding* supervisado basado en un conjunto de *autoencoders* neuronales multimodales que son entrenados a partir de fórmulas SMILES y de información de expresión génica, originalmente diseñados para predecir la sensibilidad a compuestos anticancerígenos. En el trabajo de referencia, los autores desarrollaron una serie de modelos adoptando diferentes arquitecturas. En nuestro trabajo reproducimos únicamente el *autoencoder* basado en *Self-Attention* (SA), el cual arrojó los mejores resultados, según reportaron los autores en el trabajo original. En nuestros experimentos no empleamos información de expresión génica en la fase de entrenamiento y, en cambio, entrenamos al *autoencoder* utilizando únicamente fórmulas SMILES canónicas de RDKit. Esta decisión responde a la necesidad de llevar a cabo una comparación justa entre modelos, teniendo en cuenta que los demás modelos de *embeddings* reproducidos en nuestro trabajo solo son entrenados con fórmulas SMILES.

Para lograr el modelo *PaccMann*, adaptamos el código fuente proporcionado por Oskooei et al. [287]<sup>12</sup>. Siguiendo los lineamientos en el artículo de referencia, tokenizamos las fórmulas SMILES siguiendo un algoritmo [338] que consiste en separar las fórmulas SMILES en caracteres y realizar un paso de filtrado básico. Después de realizar dicho paso de tokenización, cada secuencia se completó con caracteres de relleno para que su longitud final coincidiera con la secuencia más larga del conjunto de datos. La arquitectura de *PaccMann* consta de una capa de entrada que recibe las fórmulas SMILES tokenizadas. A continuación, se aplica opcionalmente una función de codificación posicional sinusoidal [397] a estas entradas, seguida de una capa de *Self-Attention* de un único cabezal de atención, cuya implementación se replicó exactamente según se indica en el artículo de referencia [287]. La salida de dicha capa de *Self-Attention* es luego transmitida a una red neuronal multicapa poco profunda para la predicción de propiedades. Los *embeddings* moleculares aprendidos se extraen de la salida de la capa de *Self-Attention*, por lo que su dimensionalidad se calcula como el producto entre el número de unidades ocultas  $u$  en la capa de *Self-Attention* y la longitud de la secuencia de entrada  $n$ . En consecuencia, la dimensionalidad de *embedding*  $n \times u$  es diferente para cada uno de los ocho conjuntos de datos etiquetados en los que se entrenó este modelo.

El segundo modelo supervisado que reproducimos en nuestro trabajo es *SA-BiLSTM* [449]. Siguiendo la estrategia formulada por los autores en el artículo de referencia, entrenamos este modelo empleando *embeddings* computados a partir de un modelo *Mol2Vec* preentrenado, provisto por sus autores [176]. Calculamos dichos *embeddings* para los ocho conjuntos de datos etiquetados descritos en la tabla 6.1, empleando fórmulas SMILES canónicas de RDKit para la representación inicial de

<sup>12</sup><https://github.com/drugilsberg/paccmann>

los compuestos químicos. A diferencia de los otros cuatro métodos considerados en este capítulo, lxs autorxs de *SA-BiLSTM* no han proporcionado el código fuente ni un paquete instalable para su modelo, por lo que procedimos a implementar el modelo a partir de las ecuaciones y los detalles proporcionados en el artículo original.

Según lxs autorxs, la arquitectura de *SA-BiLSTM* consta de una capa de entrada que recibe los *embeddings Mol2Vec*, seguida de una red neuronal recurrente LSTM bidireccional de una única capa. La salida de dicha red recurrente es transmitida luego a una capa de *Self-Attention* de múltiples cabezales, la cual produce los *embeddings* basados en *Self-Attention* finales de los compuestos. Estos *embeddings* son finalmente alimentados a una red neuronal multicapa poco profunda para la predicción de propiedades, que consta de una sola capa neuronal completamente conectada. Los *embeddings* obtenidos a partir de este modelo consisten en un conjunto de  $r$  vectores de dimensionalidad  $2u$ , donde  $r$  es el número de cabezales de atención y  $u$  es el número de nodos ocultos en la capa LSTM (el factor 2 se debe a que la red recurrente es bidireccional). Según el artículo de referencia [449], los  $r$  vectores obtenidos pueden ser analizados tanto por separado como de forma conjunta. Dado que nuestra intención era la de utilizar la información aprendida por todos los cabezales de atención para la predicción, concatenamos los  $r$  vectores para suministrarlos como entrada al paso de clasificación/regresión posterior. Por lo tanto, los *embeddings* resultantes de este modelo consisten en vectores de dimensionalidad  $r \times 2u$ . De manera similar al caso de *PaccMann*, la dimensionalidad de los *embeddings* varía para cada uno de los ocho conjuntos de datos etiquetados utilizados para entrenar *SA-BiLSTM*. En la tabla 6.3 se proporciona un resumen de la dimensionalidad de los *embeddings* obtenidos a partir de las dos técnicas supervisadas para cada conjunto de datos etiquetado.

### 6.3.3. Diseño experimental

En esta sección brindamos una descripción general del diseño experimental que realizamos para obtener y evaluar los *embeddings* moleculares a partir de cada uno de los modelos replicados. Un resumen del mismo se puede encontrar en la figura 6.2.

#### 6.3.3.1. Entrenamiento y obtención de *embeddings* moleculares

El primer paso de nuestra configuración experimental consistió en entrenar un modelo para cada uno de los métodos de *embedding* revisados. Entrenamos cada método respetando la cantidad de tiempo o épocas y la parametrización especificada por lxs autorxs en los artículos de referencia.



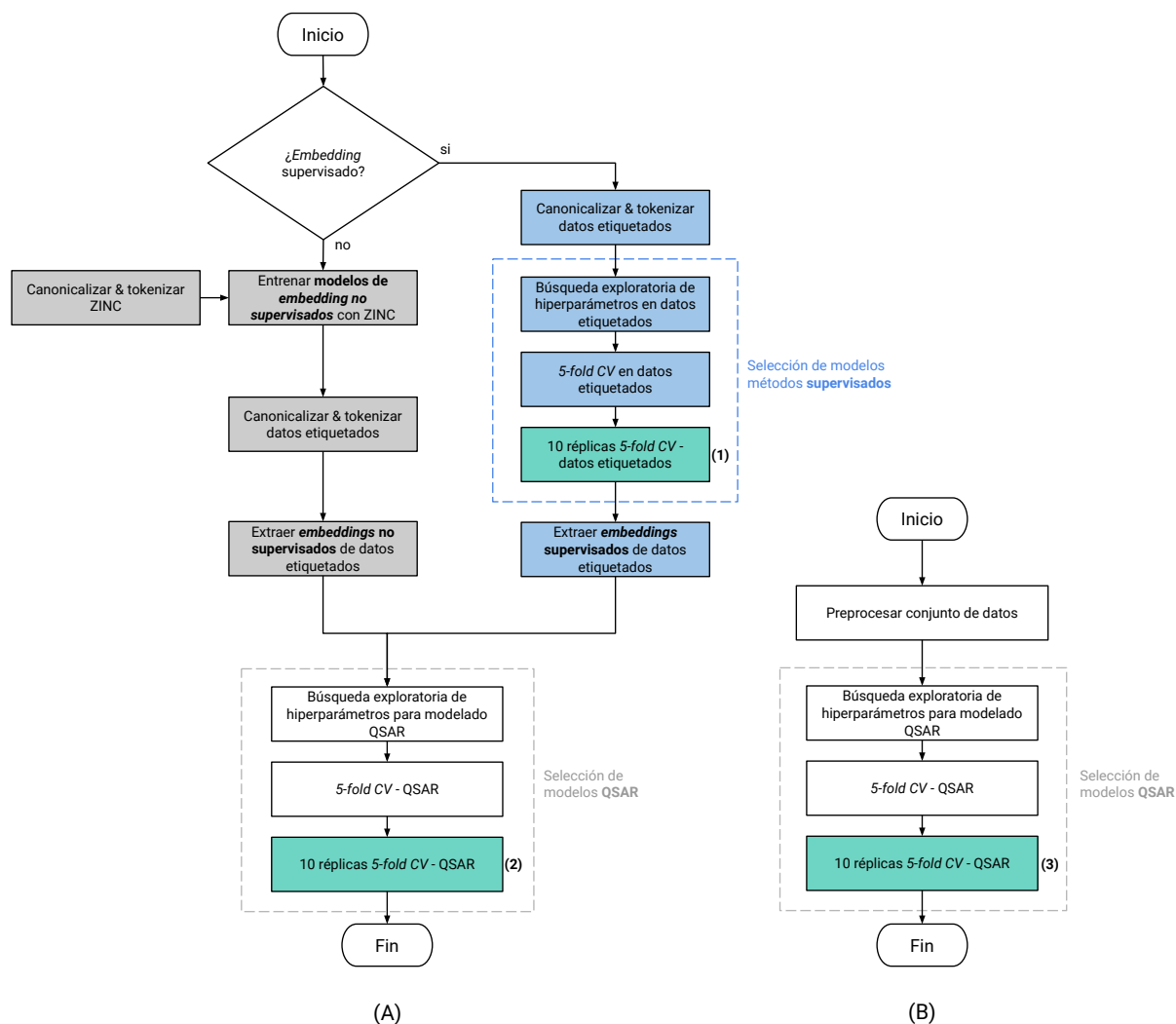


Figura 6.2: Descripción general del diseño experimental: (A) corresponde al diseño experimental para *embeddings* moleculares, mientras que (B) representa el diseño experimental para representaciones moleculares tradicionales, de acuerdo con la figura 6.1. Las celdas del diagrama en color gris corresponden a los experimentos desarrollados sobre los modelos de *embeddings* no supervisados, mientras que las celdas del diagrama en color azul corresponden a los experimentos realizados sobre las técnicas supervisadas. Las tres celdas finales (encerradas por líneas punteadas) corresponden a la etapa de modelado QSAR. Las celdas en verde indican los resultados de clasificación: (1) los resultados de *ajuste*; es decir, los resultados de clasificación y regresión obtenidos como resultado del proceso de entrenamiento inherente de los modelos supervisados para obtener sus *embeddings*; (2) los resultados de clasificación y regresión obtenidos de los modelos QSAR base entrenados usando *embeddings* supervisados o no supervisados. (3) los resultados de clasificación y regresión obtenidos de los modelos QSAR base entrenados utilizando representaciones moleculares tradicionales.

Modelo / Denominación	Conjunto de datos	Dimensionalidad de los <i>embeddings</i>
PaccMann	SR-ARE	12.000
	SR-MMP	24.000
	SR-ATAD5	24.000
	HIV	24.200
	PCBA-686978	6.976
	ESOL	9.700
	FreeSolv	5.300
	Lipophilicity	10.250
SA-BiLSTM	SR-ARE	2.560
	SR-MMP	1.920
	SR-ATAD5	1.280
	HIV	2.560
	PCBA-686978	1.280
	ESOL	1.280
	FreeSolv	1.280
	Lipophilicity	1.280

Tabla 6.3: Resumen de los modelos supervisados implementados y la dimensionalidad de los *embeddings* moleculares resultantes para cada conjunto de datos etiquetado.

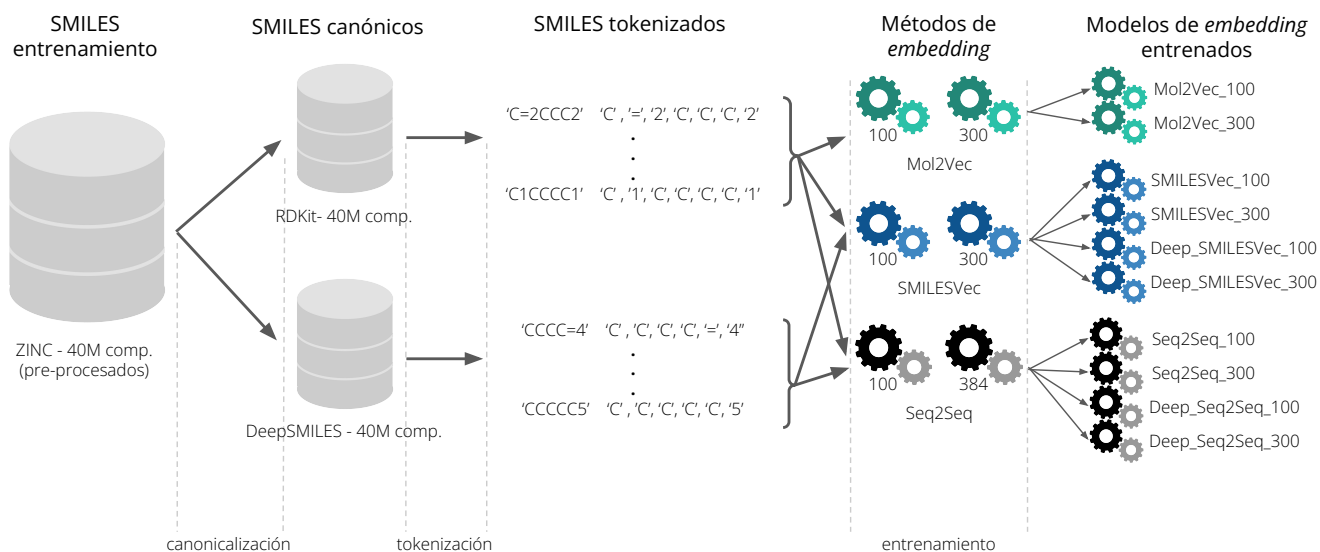
Obtuvimos diez modelos de *embeddings* no supervisados, de acuerdo con las combinaciones (método, dimensionalidad, canonicalización SMILES) indicadas en la tabla 6.2. Finalmente, tokenizamos las fórmulas SMILES en cada uno de los conjuntos de datos etiquetados de acuerdo a los requerimientos específicos de cada método y extrajimos los *embeddings* moleculares de los diez modelos.

Con respecto a los métodos supervisados (*PaccMann* y *SA-BiLSTM*), debido a que se entrenaron con conjuntos de datos etiquetados más pequeños que ZINC, realizamos una gama más amplia de experimentos durante su etapa de entrenamiento que con los métodos no supervisados, a fin de lograr modelos cuyo desempeño predictivo no pudiera atribuirse únicamente a la variación en las particiones de datos, en la inicialización de los pesos entrenables del modelo o a los hiperparámetros elegidos. Por esta razón, entrenamos los modelos supervisados siguiendo los pasos detallados a continuación:

1. Filtramos y tokenizamos los compuestos de los cinco conjuntos de datos etiquetados de acuerdo con los requisitos de cada uno de los métodos supervisados.

2. Realizamos una etapa de selección de modelos que constó de dos pasos:
  - a) Realizamos una búsqueda exploratoria de la combinación de hiperparámetros con mejor desempeño para cada método. Consideramos los rangos de valores de hiperparámetros evaluados en los artículos de referencia [287, 449], además de valores adicionales que consideramos que podrían mejorar los resultados. Seleccionamos las combinaciones de hiperparámetros que arrojaron los mejores resultados en cada conjunto de datos etiquetado. En el Apéndice A se proporciona una lista completa de todos los hiperparámetros probados en PaccMann y *SA-BiLSTM*.
  - b) Evaluamos cada una de las combinaciones de hiperparámetros seleccionadas por medio de un entrenamiento de validación cruzada de cinco pliegos estratificados (*five-fold cross-validation*), a fin de asegurar que los resultados encontrados durante la búsqueda exploratoria no fueran atribuibles únicamente al particionado de los datos utilizado durante ese proceso. Como resultado, seleccionamos una única combinación de hiperparámetros para cada conjunto de datos etiquetado en función de los resultados promedio del proceso de validación cruzada.
3. Replicamos diez veces el proceso de validación cruzada empleando diferentes semillas aleatorias para la inicialización de los parámetros entrenables de los modelos. El objetivo de este paso fue el de evitar sesgos en los resultados debido a la inicialización de los pesos en los pasos anteriores. Producto de este paso, obtuvimos los resultados de *ajuste* de los modelos supervisados, es decir, los resultados de clasificación/regresión obtenidos del proceso de entrenamiento de los métodos de *embedding* supervisados, los cuales cuentan con capas neuronales para predicción de propiedades en sus arquitecturas.
4. Seleccionamos la mejor inicialización de parámetros entrenables y entrenamos un modelo de *embeddings* utilizando una partición de datos estratificada, para finalmente extraer los *embeddings* moleculares de cada conjunto de datos etiquetados.

Además de la figura 6.2, que resume el diseño experimental de nuestro trabajo, proporcionamos un resumen gráfico más detallado de la etapa de entrenamiento de los métodos supervisados y no supervisados en las figuras 6.3 y 6.4, respectivamente.



(a) Entrenamiento no supervisado utilizando 40 millones de compuestos del conjunto de datos ZINC

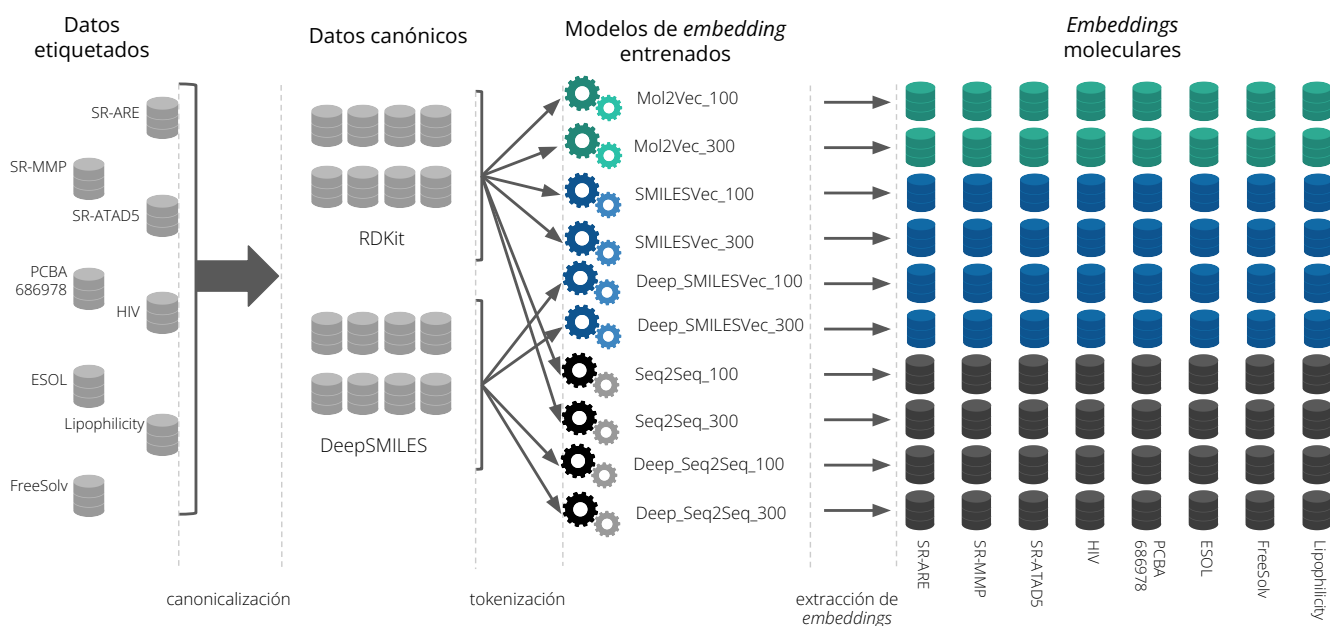
(b) Extracción de *embeddings* moleculares

Figura 6.3: Entrenamiento y extracción de *embeddings* no supervisados: (a) Procesamos y canonicalizamos las fórmulas SMILES del conjunto de datos ZINC siguiendo dos procedimientos de canonicalización diferentes. Luego tokenizamos las fórmulas canónicas según las especificaciones de cada método de *embeddings* no supervisado y procedimos a su entrenamiento. (b) Canonicalizamos y tokenizamos los ocho conjuntos de datos etiquetados de acuerdo con los requisitos de cada método. Posteriormente, obtuvimos *embeddings* moleculares para cada uno de ellos a partir de cada modelo de *embeddings* no supervisados entrenados en el paso anterior.

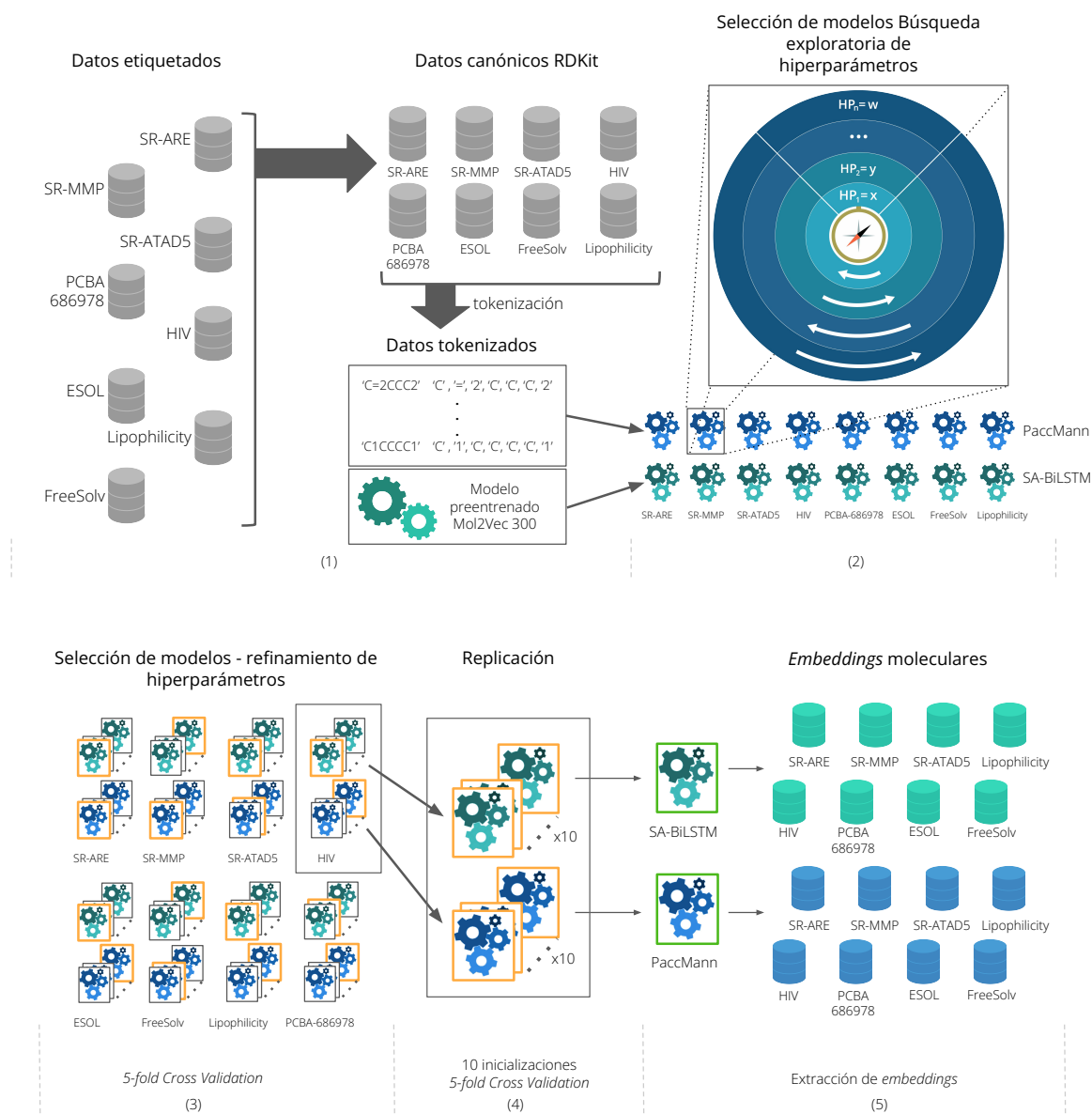


Figura 6.4: Entrenamiento y extracción de *embeddings* moleculares supervisados: (1) Procesamos y tokenizamos los compuestos de los ocho conjuntos de datos etiquetados. (2) Primer paso de selección de modelos: realizamos una búsqueda exploratoria de la combinación de hiperparámetros con mejor desempeño para cada método y elegimos una parametrización para cada conjunto de datos etiquetado. (3) Segundo paso de selección de modelos: cada parametrización hallada en el paso 2 se evaluó en un proceso de validación cruzada de cinco pliegos. Seleccionamos una única combinación de hiperparámetros para cada conjunto de datos etiquetado (resaltado en amarillo). (4) Se entrenaron diez réplicas de la parametrización seleccionada empleando diferentes semillas aleatorias y validación cruzada. (5) Entrenamos un modelo de *embeddings* usando particiones de datos estratificadas y obtuvimos *embeddings* moleculares para los compuestos de los ocho conjuntos de datos etiquetados.

### 6.3.3.2. Evaluación de representaciones moleculares

A fin de evaluar los diferentes *embeddings* moleculares aprendidos en tareas de modelado QSAR, los empleamos como representaciones moleculares en cinco tareas de clasificación y tres de regresión, definidas por los ocho conjuntos de datos etiquetados descritos en la tabla 6.1. Obtuvimos un total de diez *embeddings* moleculares no supervisados diferentes y dos *embeddings* moleculares supervisados diferentes para cada conjunto de datos etiquetado, tal y como ilustramos en las figuras 6.3 y 6.4. Además, calculamos tres representaciones moleculares tradicionales para cada conjunto de datos etiquetado: un *fingerprint ECFP* de 1.024 bits empleando RDKit [211], un *fingerprint* de claves *MACCS* de 166 bits, también calculado con RDKit, y un vector de descriptores moleculares computados usando Mordred [268]. Calculamos descriptores *0D*, *1D* y *2D* para todos los conjuntos de datos y descartamos aquellos que tenían más del 5 % de valores nulos. El número final de descriptores moleculares es el siguiente: 1.018 para *SR-ARE*, 1.016 para *SR-MMP* y *SR-ATAD5*, 1.150 para *HIV*, 1.428 para *PCBA-686978*, 1.226 para *ESOL*, 1.176 para *FreeSolv* y 1.429 para *Lipophilicity*. Todas las representaciones tradicionales se calcularon a partir de fórmulas SMILES canónicas de RDKit.

En resumen, calculamos un total de 15 (10+2+3) representaciones para cada conjunto de datos, las cuales luego empleamos para entrenar nuestros modelos QSAR. Para cada una de las tareas de clasificación, entrenamos cuatro clasificadores base diferentes, mientras que en el caso de las tareas de regresión, entrenamos tres modelos de regresión base diferentes. Para las tareas de clasificación, construimos y entrenamos un clasificador *Naïve Bayes* (NB), una *Support Vector Machine* (SVM) con un kernel RBF [333], un clasificador *Random Forest* (RF) y una red neuronal multicapa de alimentación hacia adelante (FFNN). Para las tareas de regresión, entrenamos un modelo de regresión *Ridge*, un modelo de regresión basado en *Gradient Boosting* (GBR) y una red neuronal multicapa de alimentación hacia adelante (FFNN). Utilizamos las herramientas proporcionadas por Scikit-learn [293], Keras [71] y Tensorflow [1] para implementar y entrenar dichos modelos base. Para cada modelo realizamos una búsqueda exploratoria de hiperparámetros utilizando un particionado estratificado de los datos. En el Apéndice A se proporciona una lista completa de todos los hiperparámetros probados en la etapa de selección de modelos para cada uno de dichos clasificadores y modelos de regresión.

A partir de la búsqueda de hiperparámetros, seleccionamos las mejores parametrizaciones para cada modelo QSAR base y los entrenamos empleando una estrategia de validación cruzada de cinco pliegos. En función de los resultados obtenidos durante dicha etapa, entrenamos diez réplicas, variando la inicialización de los parámetros entrenables de los modelos base por medio del uso de diferentes semillas aleatorias. Este paso se llevó a cabo solo para los métodos estocásticos (es decir, RF, GBR, FFNN y conjuntos de SVM). Durante el proceso de entrenamiento, empleamos los mismos

pliegos en cada etapa de los experimentos y en todos los casos medimos los resultados promedio de las cinco particiones de validación y sus intervalos de confianza del 95 %. Nuestra estrategia de evaluación permite una comparación justa de los resultados, minimizando los sesgos potenciales introducidos por realizar un particionado fijo de los datos tal y como se ha demostrado en estudios previos [182, 34], especialmente para conjuntos de datos pequeños.

## 6.4. Evaluación del desempeño de las representaciones moleculares en modelado QSAR

A través de todas las etapas de nuestro diseño experimental medimos el desempeño de nuestros modelos utilizando diferentes métricas, todas ellas descritas en el capítulo 2 de la presente tesis. Los resultados de las tareas de regresión se evaluaron por medio de tres métricas: *Raíz del Error Cuadrático Medio (RMSE)*, *Error Absoluto Medio (MAE)* y *Coefficiente de Determinación ( $R^2$ )*. Por su parte, los resultados de los modelos de clasificación fueron evaluados por medio de ocho métricas: *Sensibilidad ( $S_n$ )*, *Especificidad ( $S_p$ )*, *Precisión*, *Exactitud ( $Acc$ )*, *Exactitud Balanceada ( $BAcc$ )*, *Puntaje  $F1$* , *Puntaje  $H1$*  y *Área Bajo la Curva ROC ( $AUC$ )*. Para la selección de modelos y configuraciones de hiperparámetros, priorizamos *RMSE* para los modelos de regresión, y *Puntaje  $F1$*  y *Puntaje  $H1$*  para los modelos de clasificación. Estas últimas dos métricas resultan especialmente adecuadas en el caso de conjuntos de datos muy desequilibrados (como lo son las tareas de clasificación *HIV* y *SR-ATAD5*), en contraste con métricas como *Acc* [355, 65].

A fin de evaluar la significancia estadística de los resultados, realizamos una serie de pruebas estadísticas a través de las cuales comparamos los resultados obtenidos por las diferentes representaciones moleculares en cada tarea predictiva. En primera instancia, realizamos un análisis de la varianza de dos vías (ANOVA, por sus siglas en inglés) donde la representación molecular y la elección del modelo de predicción se consideraron las dos variables independientes. Ante el eventual hallazgo de que los resultados de diferentes representaciones moleculares fueran significativamente diferentes, realizamos una prueba de Tukey por pares *a posteriori* [389] con un nivel de confianza global del 95 %.

Tuvimos en cuenta los posibles efectos del desequilibrio de clases en los resultados de tareas de clasificación e incorporamos medidas de compensación, siguiendo las heurísticas detalladas en el capítulo 2. En primer lugar, entrenamos y evaluamos los modelos QSAR por medio del uso de particiones estratificadas, en las que las proporciones globales de compuestos activos e inactivos

(según hemos descrito en la tabla 6.1) son preservadas en las particiones y pliegos. En segundo lugar, empleamos funciones de costo ponderado durante el entrenamiento de los modelos de *embeddings* supervisados (*SA-BiLSTM* y PaccMann) y de las redes neuronales multicapa, de forma tal que los errores de clasificación y regresión cometidos por las redes se penalizaran siendo ajustados con un peso proporcional al desequilibrio de clase. Por último, seleccionamos métricas adecuadas a contextos de desbalance de clase para la selección de modelos y la evaluación de los resultados, como se discutió anteriormente en esta sección.

Uno de nuestros objetivos de investigación consistió en determinar si los *embeddings* moleculares serían capaces de superar a las representaciones moleculares tradicionales en tareas de modelado QSAR. Para responder a esta pregunta, comparamos los resultados obtenidos utilizando las representaciones moleculares tradicionales (descriptores moleculares, *fingerprints ECFP* y claves *MACCS*) con los resultados obtenidos utilizando *embeddings* moleculares supervisados y no supervisados. Después de realizar la prueba ANOVA de dos vías, se procedió a comparar las 15 representaciones en estudio, un modelo QSAR a la vez, realizando una prueba de Tukey por pares.

Para las tareas de clasificación, tal y como se puede ver en la figura 6.5, las representaciones tradicionales se encontraron entre las representaciones de mejor desempeño, seguidas de *SA-BiLSTM* y *Mol2Vec\_300*. Las representaciones moleculares tradicionales arrojaron resultados significativamente mejores que los obtenidos por medio de la mayoría de los *embeddings* moleculares para todos los conjuntos de datos en los clasificadores NB, SVM y RF. Sin embargo, no hubo diferencias significativas entre las tres representaciones tradicionales. Los mejores resultados para FFNN en casi todos los conjuntos de datos se obtuvieron usando *SA-BiLSTM*. En general, los resultados de los *embeddings* no supervisados no estuvieron a la altura de las representaciones tradicionales, a excepción de los *embeddings Mol2Vec* en los clasificadores FFNN. Las pruebas estadísticas mostraron en todos los casos que las diferencias entre los resultados aquí mencionados fueron significativas.

En el caso de las tareas de regresión, como se muestra en la figura 6.6, se observa un escenario similar: las representaciones moleculares tradicionales exhibieron el mejor desempeño en todos los conjuntos de datos y para todos los métodos de regresión. En particular, los descriptores moleculares y *SA-BiLSTM* en general obtuvieron los mejores resultados en GBR y FFNN en los tres conjuntos de datos. Tanto *SA-BiLSTM*, que fue la representación de mejor desempeño en FFNN, como *Mol2Vec\_100* y *Mol2Vec\_300* arrojaron resultados significativamente mejores que los obtenidos por los *embeddings* no supervisados.



La figura 6.7 muestra el desempeño medio de los modelos QSAR basados en FFNN. Los resultados se expresan en términos de  $RMSE$  y  $PuntajeF_1$  y se muestran agrupados por letras según los resultados del análisis de Tukey por pares. Aquellos desempeños medios agrupados bajo una misma letra no resultaron significativamente diferentes. En las tareas de clasificación, los *embeddings Mol2Vec* tienden a mostrar diferentes resultados dependiendo de la dimensionalidad de *embedding* y no son significativamente diferentes a los resultados de *SA-BiLSTM* o representaciones tradicionales. En los conjuntos de datos *ESOL* y *FreeSolv*, los resultados arrojados por la mayoría de las representaciones fueron significativamente diferentes entre sí, mientras que en el conjunto de datos *Lipophilicity* no hubo diferencias estadísticamente significativas entre las representaciones tradicionales, los *embeddings* supervisados y los *embeddings* no supervisados *Mol2Vec* y *SMILESVec*.

Otro de nuestros objetivos de investigación apuntó a determinar si la incorporación de información del perfil de bioactividad de los compuestos en el proceso de aprendizaje de los *embeddings* moleculares suponía algún tipo de mejora en su posterior rendimiento en modelado QSAR. Para ello, comparamos el desempeño de los *embeddings* supervisados con el de los *embeddings* no supervisados en tareas de clasificación y regresión. Para ello, realizamos una prueba ANOVA de dos vías entre todos los *embeddings* moleculares y luego realizamos una prueba de Tukey por pares a fin de comparar los resultados de ambos tipos de *embeddings*.

Como se muestra en las figuras 6.5 a 6.7, los resultados obtenidos usando la representación supervisada *SA-BiLSTM* fueron en general significativamente mejores que los resultados obtenidos empleando *embeddings* no supervisados. En las tareas de clasificación, *SA-BiLSTM* fue la representación molecular de mejor desempeño en todos los conjuntos de datos balanceados (*SR-ARE*, *SR-MMP* y *PCBA-686978*). En el caso de los conjuntos de datos *SR-ATAD5*, *HIV* y *PCBA-686978*, *SA-BiLSTM* obtuvo resultados similares a los obtenidos por *Mol2Vec\_100* y *Mol2Vec\_300*. Esto podría explicarse por el hecho de que el modelo empleado para obtener los *embeddings SA-BiLSTM* se entrena a su vez empleando una representación molecular de entrada basada en el algoritmo *Mol2Vec*. En el caso de las tareas de regresión, *SA-BiLSTM* también fue el *embedding* de mejor desempeño en todos los conjuntos de datos, mostrando resultados significativamente mejores que los obtenidos por las representaciones no supervisadas.

Por su parte, tal y como se puede observar en las figuras 6.5 y 6.7, los resultados obtenidos empleando *PaccMann* en las tareas de clasificación no mostraron diferencias significativas con los resultados obtenidos empleando los *embeddings* no supervisados. La única excepción a esta observación se da en el caso de los modelos QSAR basados en FFNN, donde los resultados alcanzados por *PaccMann* son significativamente inferiores a los obtenidos por todas las representaciones

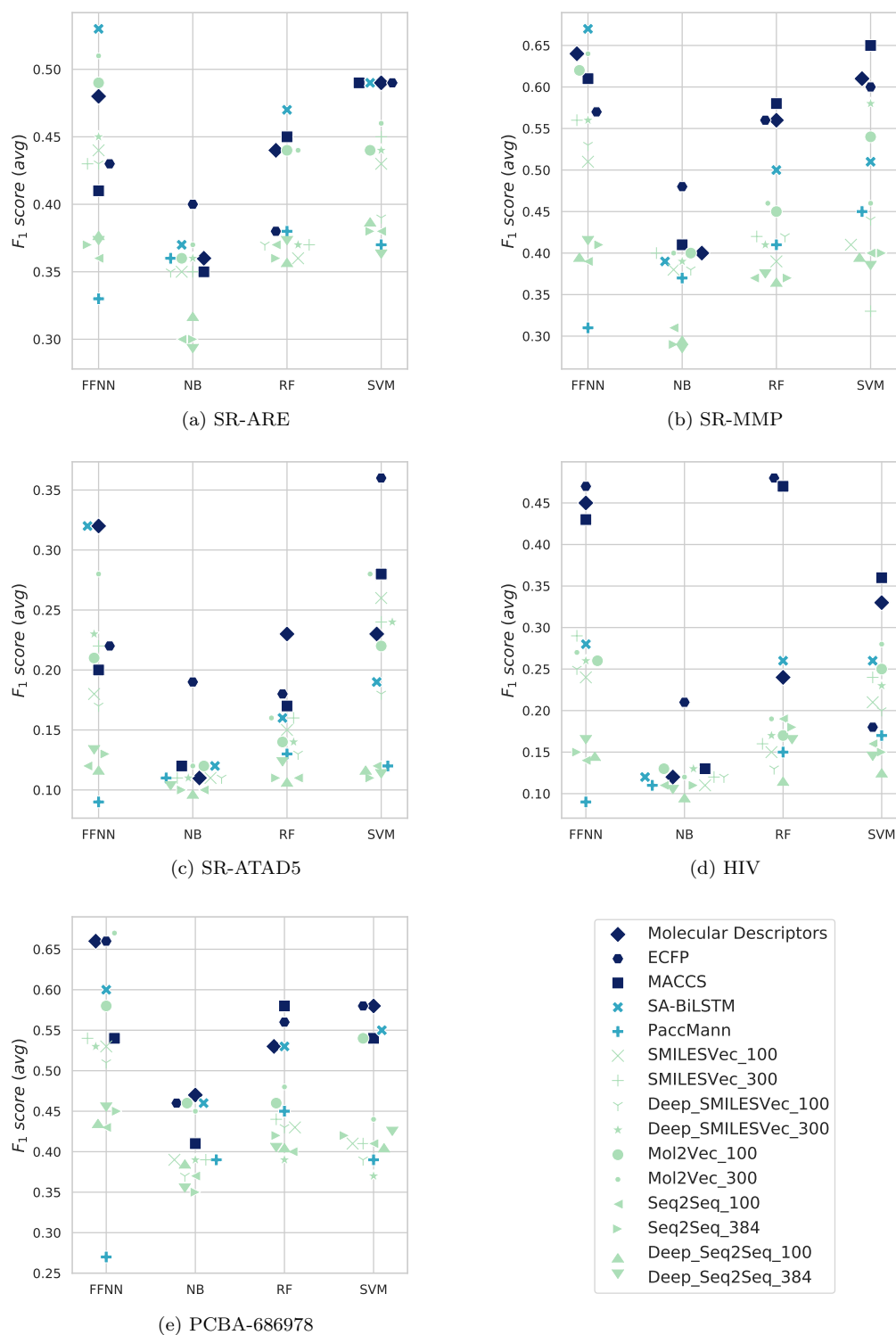


Figura 6.5: Desempeño medido en  $PuntajeF_1$  para las cinco tareas de clasificación. El color azul oscuro corresponde a las representaciones tradicionales; el azul claro corresponde a *embeddings* supervisados y el verde denota *embeddings* no supervisados. Aplicamos ruido aleatorio en la coordenada horizontal de las visualizaciones para evitar la superposición de marcas. Las representaciones tradicionales y los *embeddings* SA-BiLSTM y Mol2Vec lograron los mejores desempeños.

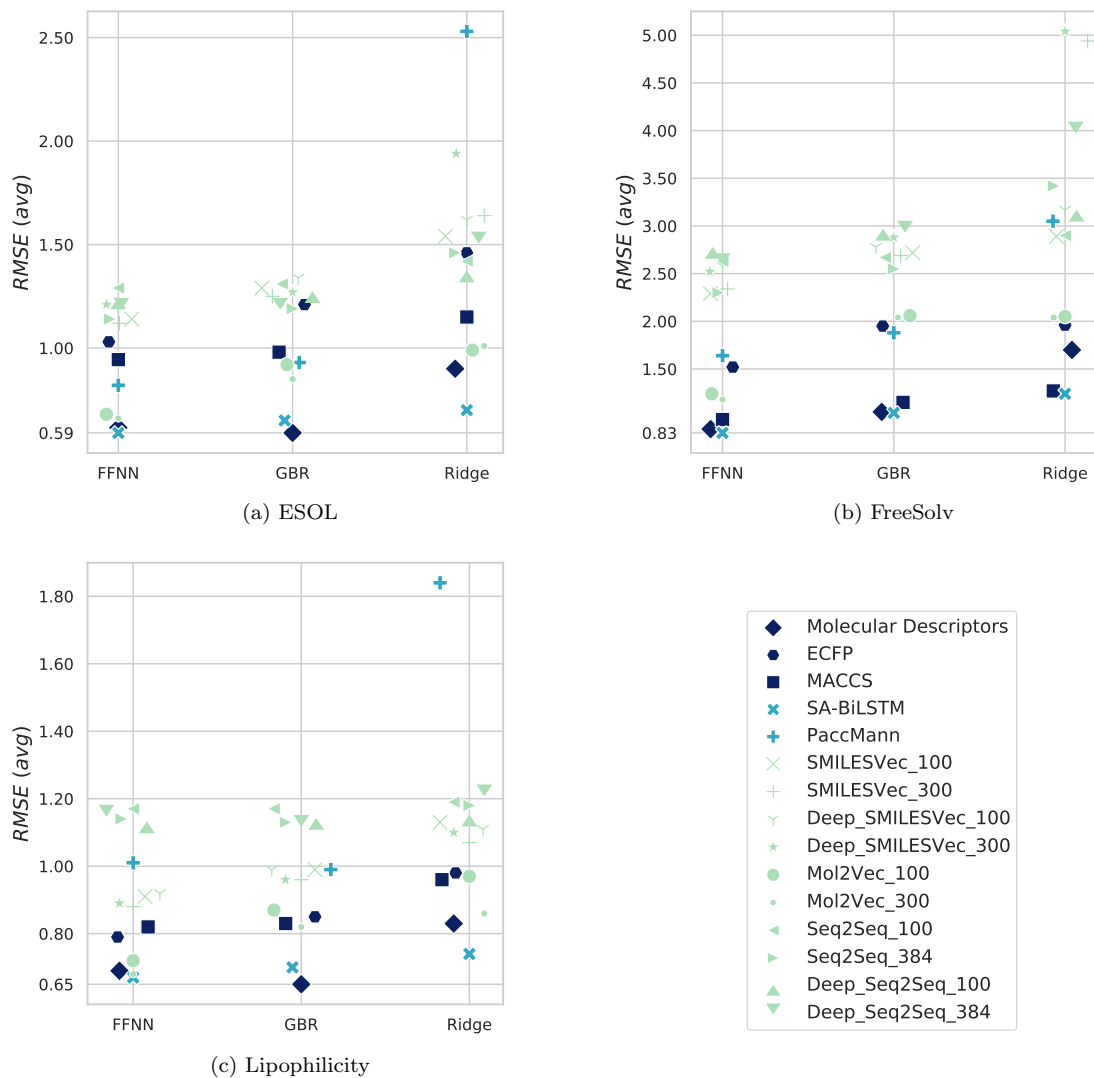


Figura 6.6: Desempeño en términos de  $RMSE$  para los tres conjuntos de datos de regresión. El color azul oscuro corresponde a las marcas de representaciones tradicionales; el azul claro corresponde a *embeddings* supervisados y el verde denota *embeddings* no supervisados. Aplicamos ruido aleatorio en la coordenada horizontal de las visualizaciones para evitar la superposición de marcas. Las representaciones tradicionales arrojaron los mejores resultados, acompañadas o seguidas por los *embeddings* SA-BiLSTM y Mol2Vec. En particular, los *descriptores moleculares* y SA-BiLSTM arrojaron los mejores resultados en FFNN y GBR para los tres conjuntos de datos.

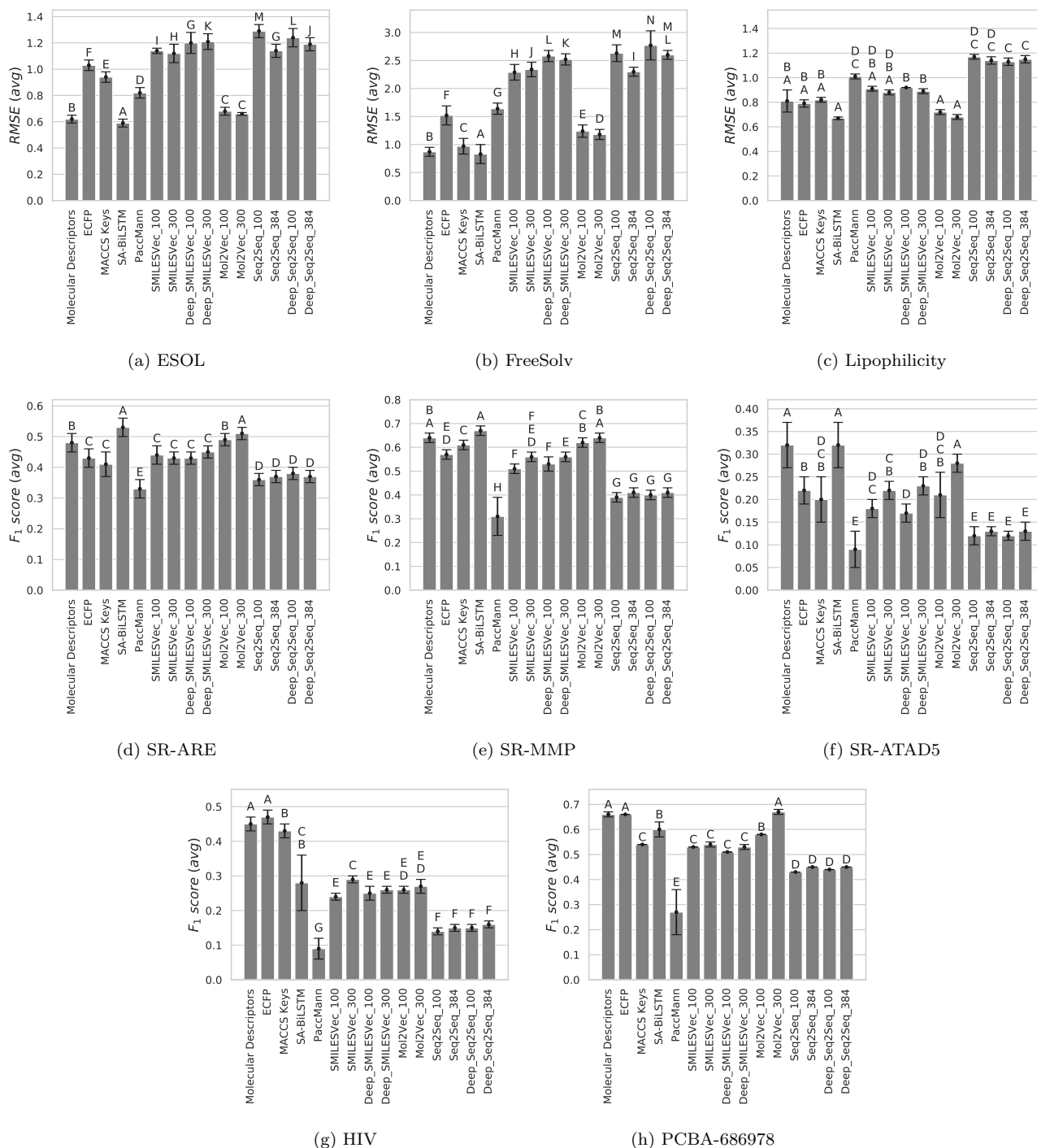


Figura 6.7: Desempeño medio expresado en términos de  $RMSE$  y  $PuntajeF_1$  de todas las representaciones moleculares en modelos FFNN para las tareas de regresión (a, b, c) y clasificación (d, e, f, g, h), respectivamente. Aquellos desempeños medios agrupados bajo la misma letra no resultaron significativamente diferentes, según las pruebas de Tukey por pares.

restantes. Sin embargo, se observa un escenario diferente en las tareas de regresión, tal y como se muestra en la figura 6.6, para las que los *embeddings PaccMann* exhibieron un desempeño significativamente mejor que todos los *embeddings* no supervisados en la mayoría de los casos.

Al comparar los métodos de *embeddings* supervisados entre sí, como se muestra en las figuras 6.8 y 6.9, *SA-BiLSTM* fue significativamente mejor que *PaccMann* para todos los conjuntos de datos. Con respecto a los resultados de *ajuste*, es decir, los resultados de clasificación obtenidos por medio de la capa de salida de los modelos de *embeddings* dedicada a la predicción de propiedades, los resultados obtenidos por *PaccMann* fueron significativamente mejores que los obtenidos en los modelos de clasificación y regresión base en todos los conjuntos de datos. En el caso de *SA-BiLSTM*, sus resultados de *ajuste* fueron superados por al menos otra técnica de clasificación en la mayoría de las tareas.

Nuestro último objetivo de investigación consistió en analizar si la forma canónica de las fórmulas SMILES utilizadas durante el entrenamiento de las representaciones moleculares o las dimensionalidades de *embedding* tenían un impacto significativo en el desempeño de los modelos QSAR. Para responder a esta pregunta, analizamos las tres técnicas de *embeddings* no supervisados por separado. Luego del análisis ANOVA de dos vías y las correspondientes pruebas de Tukey por pares para comparar los resultados obtenidos, se observaron diferentes escenarios para cada técnica no supervisada. En el caso de *SMILESTVec* en general no se observaron diferencias significativas entre los resultados de las cinco tareas de clasificación al cambiar formas canónicas, pero sí al incrementar la dimensionalidad de los *embeddings* de 100 a 300. En las tareas de regresión, las diferencias sí resultaron significativas. Al considerar *Mol2Vec*, el incremento en la dimensionalidad de los *embeddings* también supuso una mejora significativa en el desempeño predictivo para la mayoría de los conjuntos de datos. Por último, los *embeddings* obtenidos por medio de *Seq2Seq* no mostraron diferencias significativas en las tareas de regresión; sin embargo, sí se observaron diferencias significativas al cambiar la forma canónica de las fórmulas SMILES en el desempeño predictivo de las tareas de regresión.

## 6.5. Análisis del estado del arte

Adicionalmente a los resultados arrojados por nuestro análisis comparativo, en las tablas 6.4 y 6.5 presentamos los resultados obtenidos por las representaciones moleculares de mejor desempeño en cada tarea de clasificación y regresión, junto con los resultados informados en los artículos de referencia [449, 176] y otros artículos científicos [430, 182, 410] en los que se reportan desempeños de modelos QSAR para dichas tareas. Presentamos estas tablas en un esfuerzo por visibilizar otros

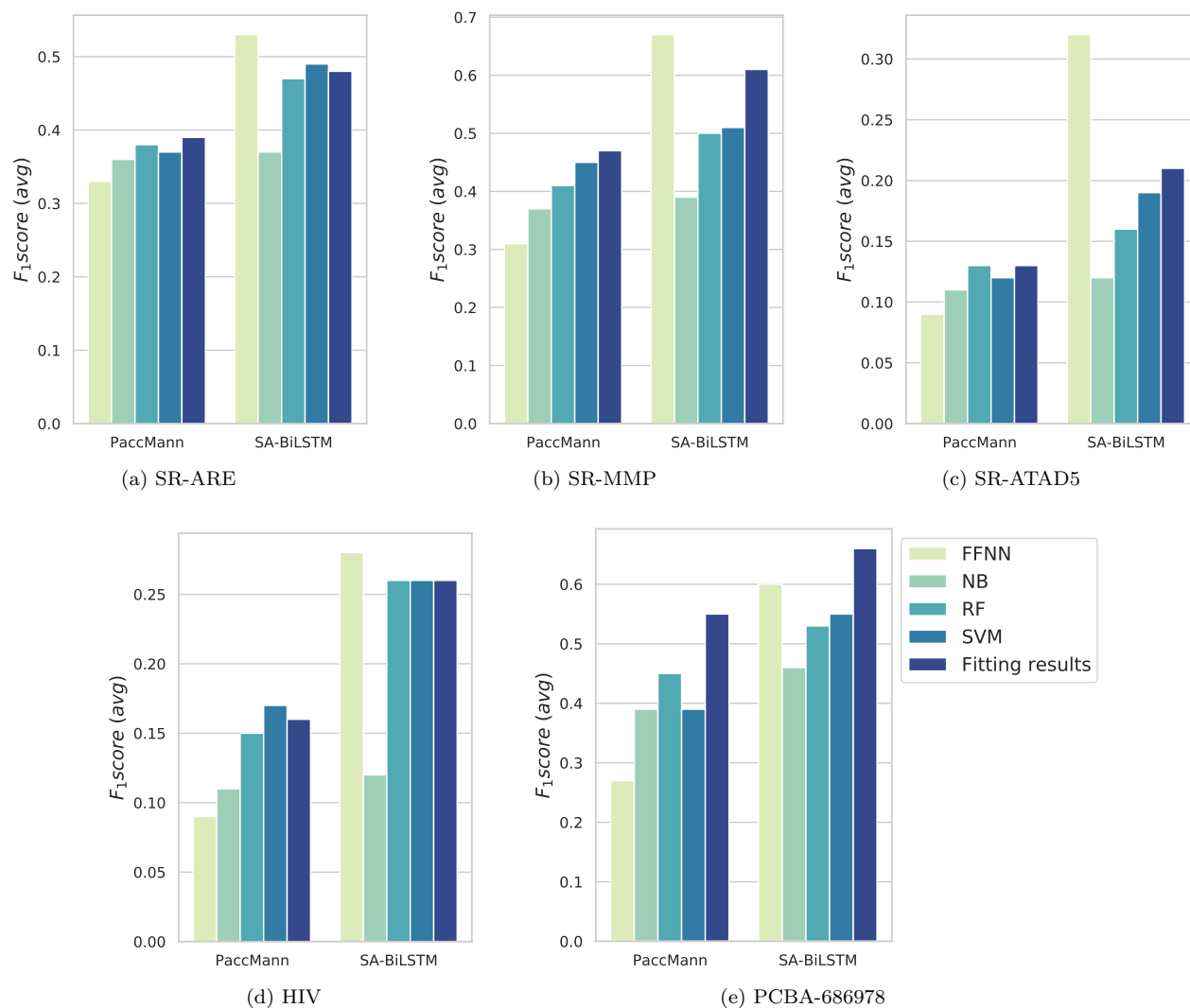


Figura 6.8: Desempeño en términos de  $Puntaje F_1$  para las cinco tareas de clasificación empleando *embeddings* supervisados. *SA-BiLSTM* obtuvo los mejores resultados, especialmente en los modelos QSAR basados en FFNN. Si bien los resultados de *PaccMann* fueron significativamente inferiores que los obtenidos por *SA-BiLSTM*, sus resultados de *ajuste* —es decir, los resultados de clasificación obtenidos al entrenar el modelo de *embeddings*— fueron en general mejores que los obtenidos en otros clasificadores.

resultados informados en la literatura; sin embargo, es importante notar que no es posible establecer una comparación directa entre nuestros resultados y los de referencia, considerando que la gran mayoría de los trabajos científicos allí citados reportan resultados en particiones fijas de validación diferentes a las empleadas en nuestro estudio.

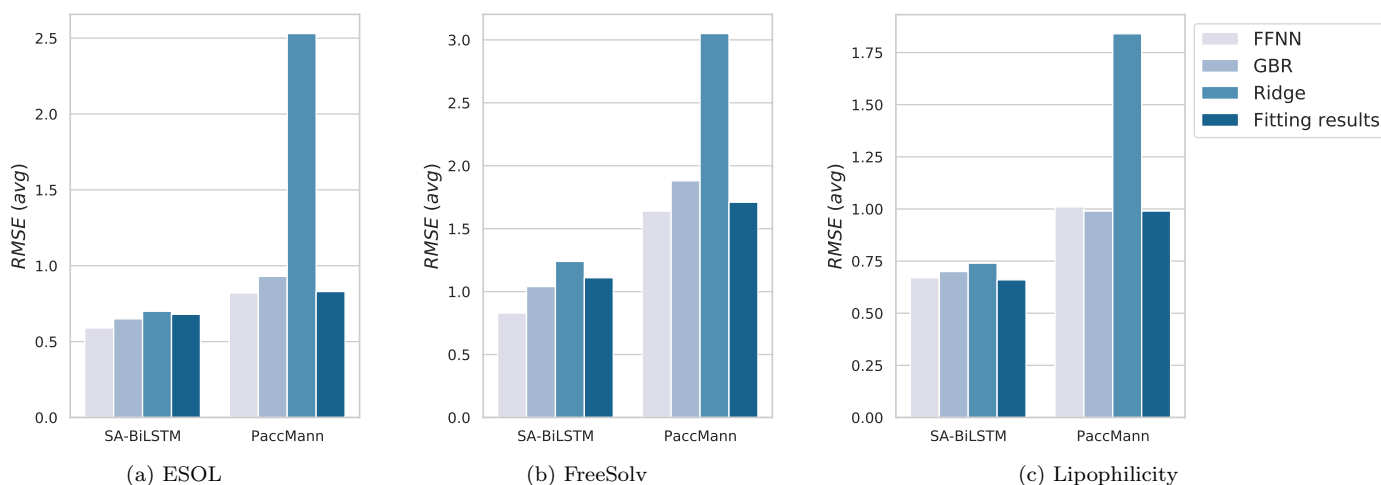


Figura 6.9: Resultados en términos de  $RMSE$  para las tres tareas de regresión obtenidos empleando *embeddings* supervisados. *SA-BiLSTM* obtuvo resultados significativamente mejores que *PaccMann*. Los resultados de *ajuste* de *PaccMann* superaron todos los demás resultados de regresión obtenidos en los modelos QSAR de regresión base.

Representación	Modelo	AUC				Acc
		SR-ARE	SR-MMP	SR-ATAD5	HIV	PCBA-686978
SA-BiLSTM [449]	<i>ajuste</i>	0,81 ± <i>na</i>	0,90 ± <i>na</i>	0,86 ± <i>na</i>	0,81 ± <i>na</i>	-
Mol2Vec [176]	RF	0,83 ± 0,05	0,83 ± 0,05	0,83 ± 0,05	-	-
Weave [430]	<i>ajuste</i>	<b>0,83 ± 0,01</b>	0,83 ± 0,01	0,83 ± 0,01	0,74 ± 0,04	-
ECFP [430]	XGBoost	0,78 ± 0,02	0,78 ± 0,02	0,78 ± 0,02	<b>0,84 ± 0,00</b>	-
SMILES-BERT [410]	<i>fine-tuning</i>	-	-	-	-	<b>0,88 ± <i>na</i></b>
SA-BiLSTM (*)	FFNN	<b>0,83 ± 0,02</b>	<b>0,91 ± 0,01</b>	<b>0,87 ± 0,02</b>	0,83 ± 0,01	0,80 ± 0,04
Mol2Vec_300 (*)	FFNN	0,81 ± 0,01	0,88 ± 0,01	0,84 ± 0,01	0,81 ± 0,01	0,83 ± 0,00
ECFP (*)	FFNN	0,74 ± 0,02	0,84 ± 0,01	0,70 ± 0,04	0,78 ± 0,01	0,87 ± 0,00

Tabla 6.4: Resultados informados en los artículos científicos de referencia para los cinco conjuntos de datos de clasificación, en términos de  $AUC$  y  $Acc$ , junto con los mejores resultados obtenidos a través de nuestro diseño experimental (marcados con un asterisco). Los mejores resultados por conjunto de datos se destacan en **negrita**. La leyenda *na* denota que los autores del artículo de referencia no proporcionaron la información correspondiente en el artículo.

Los artículos de referencia informaron el desempeño de la regresión en términos de  $RMSE$  para las tareas *ESOL*, *FreeSolv* y *Lipophilicity*, y el desempeño de la clasificación en términos de  $AUC$  para las tareas *SR-ARE*, *SR-MMP*, *SR-ATAD5* y *HIV*. En el caso del conjunto de datos *PCBA-686978*

Representación	Modelo	RMSE		
		ESOL	FreeSolv	Lipophilicity
Attentive FP [182]	<i>ajuste</i>	<b>0,48 ± na</b>	0,52 ± na	<b>0,52 ± na</b>
MOE + FPs [182]	SVM	0,62 ± na	<b>0,42 ± na</b>	0,55 ± na
MOE + FPs [182]	XGBoost	0,51 ± na	0,69 ± na	<b>0,52 ± na</b>
MPNN [430]	<i>ajuste</i>	0,55 ± 0,02	1,20 ± 0,02	0,76 ± 0,03
Weave [430]	<i>ajuste</i>	0,57 ± 0,04	1,19 ± 0,08	0,73 ± 0,01
GC [430]	<i>ajuste</i>	1,05 ± 0,15	1,35 ± 0,15	0,68 ± 0,04
SA-BiLSTM (*)	FFNN	0,59 ± 0,03	0,83 ± 0,17	0,67 ± 0,01
Mol2Vec_300 (*)	FFNN	0,66 ± 0,01	1,18 ± 0,09	0,68 ± 0,02
Descriptores moleculares (*)	FFNN	0,62 ± 0,03	0,87 ± 0,08	0,69 ± 0,03
Descriptores moleculares (*)	GBR	0,59 ± 0,04	1,05 ± 0,12	0,65 ± 0,02

Tabla 6.5: Resultados informados en los artículos científicos de referencia para los tres conjuntos de datos de regresión, en términos de *RMSE*, junto con los mejores resultados obtenidos a través de nuestro diseño experimental (marcados con un asterisco). Los mejores resultados por conjunto de datos se destacan en **negrita**. La leyenda *na* denota que lxs autorxs del artículo de referencia no proporcionaron la información correspondiente en el artículo.

los resultados fueron reportados en términos de *Acc* [410], aunque dicha métrica no es apropiada para conjuntos de datos desbalanceados [355, 65]. Como se muestra en la tabla 6.4, los resultados de nuestros experimentos en los conjuntos de datos de clasificación están a la par o superan los resultados reportados en los artículos de referencia en los conjuntos de validación interna, lo que evidencia la solidez de nuestros resultados y pone de manifiesto la importancia de desarrollar una cuidadosa etapa de selección de modelos. En el caso de las tareas de regresión, que se muestran en la tabla 6.5, nuestros resultados experimentales igualaron o superaron los resultados de solamente uno de los dos artículos de referencia [430]. Sin embargo, al intentar reproducir los resultados reportados en el otro artículo [182] utilizando el código fuente proporcionado por lxs autorxs, llamativamente los desempeños obtenidos no se correspondieron con los reportados en dicho artículo y, en cambio, estuvieron a la par de los nuestros.

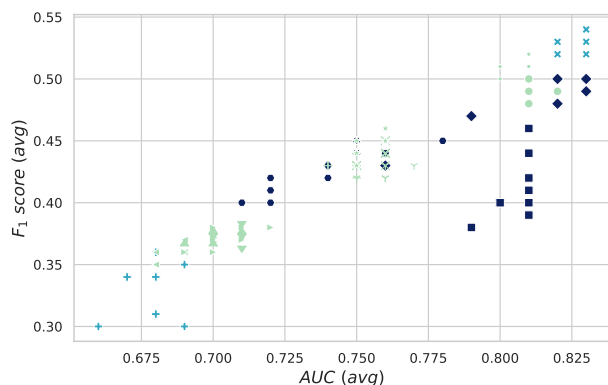


## 6.6. Análisis de dispersión de resultados de clasificación

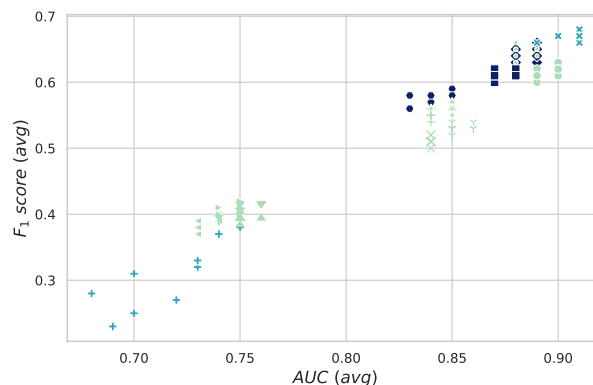
Finalmente, en la figura 6.10 presentamos los resultados de clasificación de los modelos QSAR basados en FFNN. Estos resultados fueron los obtenidos en diez réplicas por representación molecular, cada una ejecutada a partir de una inicialización aleatoria diferente de los parámetros entrenables de la FFNN. En la figura, los resultados se expresan en términos de *Puntaje F1* y *AUC*, y muestran la dispersión de los valores de desempeño obtenidos por cada representación molecular. Curiosamente, mientras que ECFP obtuvo los mejores resultados en términos de *Puntaje F1* para la mayoría de las tareas de clasificación, *SA-BiLSTM* coincide o supera a ECFP al analizar los resultados en términos de *AUC* en la mayoría de las mismas. Como se puede ver en la tabla 6.4 y en la figura 6.10, *SA-BiLSTM* exhibe los mejores desempeños medidos en *AUC* para modelos QSAR basados en FFNN, a menudo a la par con los descriptores moleculares y los *embeddings Mol2Vec*. Observamos una baja dispersión a través de diferentes ejecuciones empleando las mismas representaciones moleculares, a excepción de algunos de los desempeños para la tarea *SR-ATAD5*, caracterizada por un conjunto de datos pequeño y muy desbalanceado.

Sobre la base de nuestras observaciones, podemos concluir que, en general, los *embeddings* moleculares no superaron ampliamente los resultados obtenidos por las representaciones moleculares tradicionales. Además, la mayoría de los métodos de *embedding* no supervisados no lograron alcanzar los resultados obtenidos por los *embeddings* supervisados. Entre las técnicas de *embeddings* no supervisados, *Mol2Vec* arrojó los mejores resultados, por lo general con un desempeño significativamente mejor que los resultados obtenidos con *SMILESVec* o *Seq2Seq* desde un punto de vista estadístico. Esto puede deberse al paso de preprocesamiento de las fórmulas SMILES empleado por el algoritmo de *Mol2Vec*, que se basa en el algoritmo para calcular los *fingerprints ECFP*, en contraste con los simples pasos de tokenización aplicados a las fórmulas SMILES en el caso de las otras dos técnicas.

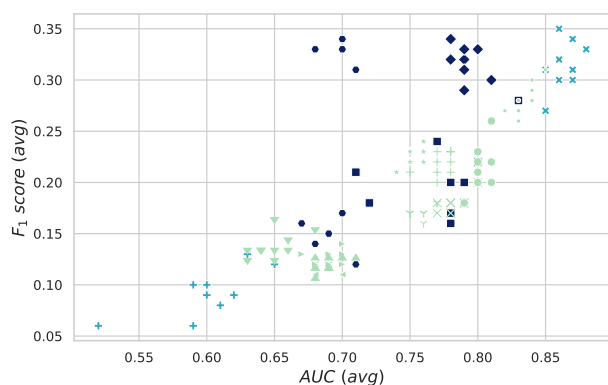
En el caso de los *embeddings* supervisados, por un lado, *SA-BiLSTM* arrojó los mejores resultados entre dichas representaciones, encontrándose generalmente a la par de los resultados obtenidos por las representaciones moleculares tradicionales. Por su parte, *PaccMann* no dio resultados satisfactorios en las tareas de clasificación. Esto podría deberse a la alta dimensionalidad de los *embeddings* de *PaccMann*, lo que suele estar relacionado con un bajo desempeño predictivo de los modelos tradicionales de aprendizaje automático [126, 9]. Otra posible razón de la diferencia estadísticamente significativa entre los resultados de las dos técnicas de *embeddings* supervisados es que la arquitectura neuronal de *SA-BiLSTM* consta de una capa de *Self-Attention* de múltiples cabezales, mientras que *PaccMann* es un modelo de *Self-Attention* de un único cabezal de atención.



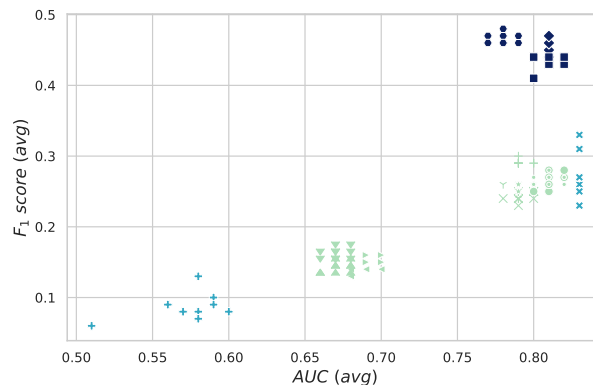
(a) SR-ARE



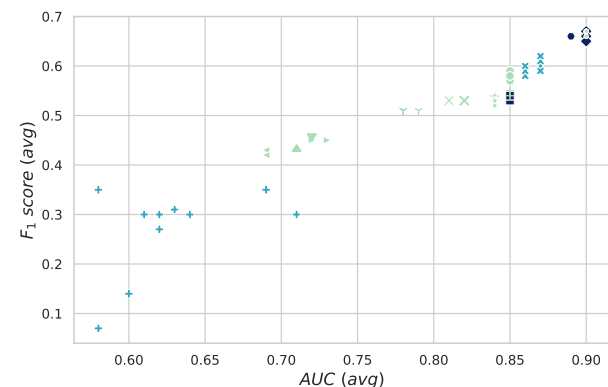
(b) SR-MMP



(c) SR-ATAD5



(d) HIV



(e) PCBA-686978

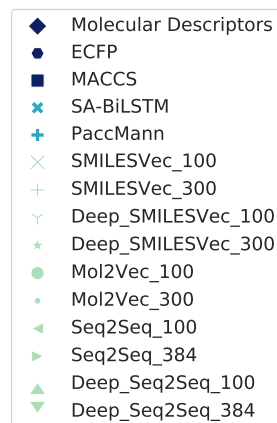


Figura 6.10: Desempeños en términos de  $PuntajeF_1$  y  $AUC$  para diez réplicas de modelos de clasificación FFNN, cada uno con una inicialización aleatoria diferente. *SA-BiLSTM* obtuvo los mejores resultados en la mayoría de las tareas de clasificación, a menudo en par con los obtenidos por los *descriptores moleculares* o *embeddings Mol2Vec*. Las representaciones tradicionales arrojaron mejores resultados medidos en  $PuntajeF_1$  que *SA-BiLSTM* en los conjuntos de datos más grandes (*HIV* y *PCBA-686978*).

Nuestro diseño experimental permitió realizar una comparación exhaustiva y precisa de diferentes representaciones moleculares, considerando diferentes técnicas de clasificación y regresión, tamaños de conjuntos de datos y escenarios de desbalance de clases. También probamos diferentes parametrizaciones y variaciones de los datos de entrada para cada modelo de *embeddings* moleculares. Las conclusiones derivadas de nuestro trabajo provienen del hallazgo de resultados consistentes a lo largo de las diferentes etapas de experimentación, y coincidieron con otros resultados previamente informados en la literatura [449, 430, 182, 176, 410]. El hallazgo de que los *embeddings* moleculares no lograron mejorar significativamente el desempeño predictivo de los modelos QSAR con respecto al uso de representaciones tradicionales también se puede encontrar en algunos trabajos previos relacionados [182, 176, 124, 122, 440, 439, 69]. Sin embargo, tales resultados no podrían haberse considerado concluyentes, ya que no están respaldados por ninguna prueba de significancia estadística y no hubo una comparación sistemática donde los hiperparámetros se ajustaran de manera meticulosa. Argumentamos que nuestros resultados demuestran la importancia de realizar una comparación experimental minuciosa y cuidadosa de las técnicas de *embeddings* moleculares y de evaluar el rol potencial de los *embeddings* moleculares en las tareas de modelado de propiedades.

## 6.7. Síntesis y conclusiones

En los últimos años, se han propuesto numerosos algoritmos basados en aprendizaje profundo para aprender *embeddings* moleculares. Sin embargo, paralelamente al desarrollo de nuevos algoritmos, resulta fundamental realizar comparaciones sistemáticas y cuidadosas entre los diferentes métodos para la obtención de representaciones moleculares que permitan arrojar luz sobre los aspectos clave de dicho proceso. En este capítulo presentamos nuestro trabajo de investigación [324] en el que llevamos a cabo un extenso análisis comparativo de diversas técnicas de representación molecular que derivó en el entrenamiento y evaluación de más de 25.000 modelos. Evaluamos el desempeño de cinco técnicas diferentes para aprender *embeddings* moleculares en el contexto de modelado QSAR de cinco tareas de clasificación y tres de regresión. Para dicha evaluación, tuvimos en consideración aspectos clave como la dimensionalidad de la representación, la canonicalización de las fórmulas SMILES y la complejidad de los algoritmos de aprendizaje profundo en términos de sus parámetros entrenables, y las comparamos con representaciones moleculares tradicionales y de uso establecido en la comunidad científica.

Como resultado de nuestro amplio análisis experimental, no encontramos evidencia que soporte la hipótesis de que los *embeddings* moleculares superen significativamente en desempeño a los

descriptores moleculares o *fingerprints* como representaciones moleculares para entrenar modelos QSAR. Nuestros resultados indican en principio que las representaciones tradicionales resultan tan expresivas como los *embeddings* moleculares para representar compuestos químicos en tareas de predicción QSAR. Por otra parte, los *embeddings* moleculares no supervisados en general exhibieron un desempeño más bajo que las técnicas de *embedding* supervisadas.

A pesar de los resultados hallados, que en principio parecen contradecir la tendencia actual en la investigación sobre representaciones moleculares, el diseño de fármacos tiene numerosas áreas de aplicación florecientes donde las representaciones moleculares basadas en aprendizaje profundo tienen gran potencial, como lo es el diseño de fármacos *de novo* o el estudio de acoplamiento molecular (*docking*) [73, 103, 332], lo que abre una amplia gama de posibilidades experimentales y trabajo futuro. Además, técnicas basadas en aprendizaje profundo como *Self-Attention* tienen el potencial de permitir el aprendizaje de *embeddings* especialmente idóneos para la identificación de fragmentos o subestructuras moleculares asociados a perfiles farmacológicos específicos [449].

Mientras que las representaciones tradicionales son calculadas siguiendo algoritmos estándar y, en algunos casos, incluso de a una molécula a la vez, los *embeddings* moleculares se pueden calcular a partir de grandes conjuntos de compuestos químicos, lo que puede dar lugar a representaciones más ricas que podrían ser adecuadas para el análisis de similitud molecular [73, 167]. En el proceso de cribado virtual de fármacos, el análisis de similitud estructural es fundamental para el hallazgo de nuevos compuestos candidatos. Las herramientas de analítica visual resultan particularmente útiles como soporte a expertos en este proceso, con el consecuente desafío de contar con representaciones moleculares que permitan encontrar patrones de similitud entre compuestos químicos de interés por medio de la exploración visual. A la vez, la multiplicidad y complementariedad de representaciones moleculares permite aprovechar las fortalezas de cada representación y elegir la o las que mejor se adecúan a cada conjunto de datos específico. En el siguiente capítulo de la presente tesis doctoral examinamos *ChemVA*, una herramienta interactiva de analítica visual propuesta por nuestro equipo para la exploración de conjuntos de compuestos químicos empleando múltiples vistas coordinadas, que permite inspeccionar y comparar compuestos candidatos por medio de diferentes representaciones moleculares.

# Capítulo 7

## Analítica visual aplicada a cribado virtual

Las diferentes representaciones moleculares codifican información complementaria sobre la estructura y propiedades de un compuesto químico. Su análisis conjunto resulta valioso para la identificación de patrones de similitud entre compuestos en cribado virtual. Las herramientas de analítica visual son indispensables para extraer sentido de los datos químicos, comúnmente de alta dimensionalidad y de semántica abstracta. En este capítulo exploramos *ChemVA*, una herramienta de analítica visual que permite la exploración interactiva de conjuntos de compuestos químicos empleando técnicas de visualización y múltiples representaciones moleculares.

---

### 7.1. Introducción

El diseño de fármacos se basa en el análisis de pequeños compuestos químicos orgánicos. Como hemos discutido en el capítulo 3, el hallazgo de nuevos fármacos se logra mediante la exploración de quimiotecas de gran volumen y por medio de la aplicación de conocimiento experto para el diseño de nuevas estructuras moleculares con funciones terapéuticas específicas. En las últimas décadas, el cribado de compuestos químicos ha sido el procedimiento principalmente aplicado durante las etapas iniciales del proceso de descubrimiento de fármacos [233, 156]. Este proceso requiere de realizar síntesis química, pruebas experimentales de grandes bibliotecas de compuestos frente a un blanco biológico, y tiene una alta tasa de deserción, lo que hace que el proceso sea costoso y consuma mucho tiempo. Estas desventajas estimularon el desarrollo de métodos de cribado virtual, que consisten en la integración de conocimiento experto a técnicas computacionales para facilitar y acelerar la identificación de compuestos candidatos. El cribado virtual permite analizar, filtrar y seleccionar

un gran número de compuestos candidatos a un ritmo significativamente más rápido y a un menor costo [226, 443]. Además, las técnicas computacionales empleadas en el cribado virtual permiten simular y medir la aptitud del compuesto candidato para la función farmacológica deseada sin la necesidad de síntesis química y costosos trabajos de laboratorio [368]. Estas razones hacen que el cribado virtual sea hoy en día una parte esencial del proceso de descubrimiento de fármacos en etapas tempranas.

En el proceso moderno de desarrollo de fármacos, los expertos se enfrentan a la vastedad del espacio de compuestos candidatos, tanto en términos de la cantidad de compuestos a considerar como de sus representaciones computacionales. En los últimos años, se han propuesto muchas herramientas de visualización para el cribado virtual, las cuales se han centrado en proporcionar medios adecuados para explorar el espacio químico y mejorar la interpretabilidad de los resultados, de modo que se puedan tomar decisiones confiables y explicables [302, 161]. Al trabajar con representaciones tradicionales de alta dimensionalidad generalmente se requiere del uso de estrategias de reducción dimensional para procesar de manera eficiente el espacio de características moleculares. Una estrategia comúnmente adoptada por dichas herramientas es la aplicación de técnicas de reducción de dimensionalidad, que permiten establecer un mapeo de compuestos entre un espacio químico de alta dimensión hacia una representación de menor dimensión (generalmente 2D o 3D) [142]. Sin embargo, como hemos visto anteriormente, los cálculos subyacentes involucrados en dicho proceso pueden sesgar o dificultar la interpretabilidad de los resultados del proceso de desarrollo de un nuevo fármaco candidato, lo cual constituye un desafío no resuelto por las herramientas de analítica visual para cribado virtual preexistentes.

Otro desafío por resolver es estudiar la similitud de compuestos bajo la premisa del principio de similitud estructural, expuesto en el capítulo 3, según el cual moléculas estructuralmente similares tienden a tener perfiles de bioactividad similares [185]. Las herramientas de visualización deben permitir que el experto en el campo interactúe con diferentes fuentes de información molecular y proporcionen vistas complementarias que ayuden a encontrar los determinantes de similitud. La mayoría de las herramientas existentes se centran en un único mapeo basado en un conjunto arbitrariamente elegido de características moleculares. En este sentido, los *embeddings* moleculares se computan por medio de modelos entrenados con grandes conjuntos de compuestos químicos diversos. Además de ser compactos, los *embeddings* son representaciones contextuales, donde cada compuesto químico es representado en función no solo de su propia estructura molecular, sino de la información química y estructural de los demás compuestos en la quimioteca utilizada en el proceso de entrenamiento. Esta característica los convierte en representaciones ricas en información del espacio

molecular en estudio, por lo que resultan adecuados para estudiar la similitud entre compuestos [73, 167].

Una herramienta de visualización para el cribado virtual debe también proporcionar vistas e interacciones que permitan a lx expertx en el campo evaluar la relación entre los perfiles de bioactividad de los compuestos analizados y su organización espacial en el espacio proyectado de baja dimensión. Considerando la complejidad del espacio molecular que se debe explorar en un proceso de cribado virtual, las herramientas de analítica visual deben agilizar la exploración de espacios moleculares y agrupamientos de compuestos químicos, a la vez que permitan flexibilidad e independencia de la representación molecular empleada. En este contexto, y teniendo en cuenta las bondades de los *embeddings* a la hora de analizar patrones de similitud estructural, resulta de interés incorporar este tipo de representaciones moleculares en herramientas de cribado virtual por medio de la exploración visual. Sin embargo, dada la amplia variedad de representaciones moleculares con semánticas muchas veces complementarias, en el diseño de una estrategia de analítica visual es importante permitir el análisis del espacio químico en estudio por medio de la comparación de múltiples representaciones y sus visualizaciones, de forma tal de evaluar la relación entre espacialidad, localidad y bioactividad.

En este sentido, como consecuencia de la reducción de dimensionalidad, las distancias entre pares de compuestos en el mapeo de baja dimensión pueden diferir de las correspondientes distancias en el espacio de alta dimensión. Estas discrepancias podrían llevar a interpretaciones erróneas al evaluar similitud estructural en la representación de baja dimensionalidad. La información sobre la confiabilidad de estas proyecciones debe presentarse visualmente a lx expertx en el campo de manera interpretable. A pesar de su importancia, la mayoría de las herramientas existentes no implementan estrategias para cuantificar la confiabilidad de las proyecciones o métricas asociadas a esta información.

En el presente capítulo de la tesis introducimos *ChemVA*, una herramienta interactiva de analítica visual cuyo objetivo es brindar soporte a expertxs en el proceso de cribado virtual de fármacos<sup>1,2</sup>. *ChemVA* permite explorar y comparar compuestos candidatos por medio de múltiples vistas coordinadas, que muestran las proyecciones en dos y tres dimensiones de diferentes representaciones moleculares combinadas con información de actividad biológica, características moleculares seleccionadas y estimaciones de confianza para cada una de dichas proyecciones. Más

---

<sup>1</sup>Video promocional de *ChemVA* para la conferencia internacional *IEEE Visualization 2020*: <https://www.youtube.com/watch?v=vKMRGer-pAY>

<sup>2</sup>Presentación de *ChemVA* en la conferencia internacional *IEEE Visualization 2020*: <https://www.youtube.com/watch?v=Pz32r3DdjQ&t=3325s>

aún, *ChemVA* permite comparar distintas proyecciones por medio de una vista coordinada novedosa, denominada *vista de Contraste*, e incorpora una codificación por medio de *puntajes de correlación* en los gráficos que permite estimar la confiabilidad de una proyección de baja dimensión en función de la distorsión con respecto a las distancias en el espacio de alta dimensionalidad entre los compuestos proyectados. Los gráficos se complementan con una tabla interactiva, que brinda información detallada sobre los compuestos y permite ordenar y filtrar los mismos. La herramienta también contiene una *vista 3D* que permite explorar la similitud estructural entre los compuestos seleccionados, realizando una yuxtaposición tridimensional de los compuestos seleccionados con respecto a su subestructura común, lo que ayuda a identificar similitudes y diferencias entre ellos. Finalmente, *ChemVA* permite cargar y proyectar nuevos compuestos a un conjunto de datos existente, lo que posibilita el estudio de posibles nuevos compuestos candidatos de forma contextual.

Nuestra propuesta fue evaluada en dos casos de estudio de identificación de ligandos estructuralmente similares a proteínas objetivo, así como por medio de una evaluación cualitativa desde el punto de vista de la interfaz de usuario. Nuestra herramienta permitió inspeccionar efectivamente el espacio molecular en estudio y comparar diferentes representaciones moleculares, asistiendo además en el diseño de compuestos *de novo* a partir de la identificación de subestructuras moleculares relevantes.

## 7.2. Trabajo relacionado con la propuesta

En esta sección del capítulo realizamos un breve estudio de los enfoques y herramientas preexistentes en diferentes áreas relacionadas con nuestro trabajo. En particular, nos concentramos en trabajos de analítica visual empleando técnicas de reducción de dimensionalidad, en sus prestaciones para la incorporación de nuevos compuestos candidatos y para la exploración visual avanzada de grandes quimiotecas.

### 7.2.1. Exploración visual de datos multi-dimensionales

En las últimas décadas, han proliferado una amplia variedad de métodos de visualización para datos multi-dimensionales. El descubrimiento de patrones de similitud entre instancias de datos en contextos de alta dimensionalidad utilizando una combinación de técnicas visuales y de aprendizaje automático representa un desafío estudiado en el área. La introducción de técnicas de reducción dimensional permite involucrar ciertas habilidades perceptuales y cognitivas únicas de lxs humanxs



en el proceso de descubrimiento de patrones multi-dimensionales por medio de la exploración visual. El desafío principal radica en realizar un manejo eficiente de múltiples variables y sus potenciales interrelaciones, siendo que la capacidad de la usuario para comprender interacciones y correlaciones entre variables es inversamente proporcional a la cantidad de variables involucradas.

Típicamente, se realiza una transformación del conjunto de datos original utilizando una técnica de reducción dimensional y luego se codifica visualmente los datos en el espacio reducido. La técnica de codificación visual más frecuentemente utilizada para esta tarea es el *gráfico de dispersión* (*scatterplot* en inglés). Sedlmair et al. [340] llevaron a cabo una investigación exhaustiva sobre la efectividad de las elecciones de codificación visual, que incluyen gráficos de dispersión en 2D, gráficos de dispersión interactivos en 3D y matrices de gráficos de dispersión. Sus hallazgos sugieren que el gráfico de dispersión en 2D es el enfoque más adecuado para explorar los resultados de diferentes algoritmos de reducción de dimensionalidad.

En este contexto, se han introducido una variedad de enfoques para capturar visualmente información de alta dimensionalidad mediante proyecciones bidimensionales o tridimensionales, a la vez intentando preservar las condiciones de espacialidad del conjunto de datos de origen (por ejemplo, los agrupamientos y relaciones de vecindad entre las instancias) [394, 434]. Entre estos métodos, *t-distributed Stochastic Neighbor Embedding* (t-SNE) [393] ha sido una de las técnicas más ampliamente adoptadas, específicamente para la visualización de datos de muy alta dimensionalidad [255, 307, 366, 427].

Los métodos lineales para reducción dimensional, como PCA [425], LDA [109] o MSR [361], han sido ampliamente utilizados; sin embargo, están limitados a transformaciones lineales de los datos originales, las cuales otorgan resultados interpretables casi exclusivamente en aquellos casos de conjuntos de datos linealmente separables. Las técnicas no lineales, como SOMs [201], MDS [77] o autoencoders [205], si bien permiten mapear datos no lineales, padecen de un problema recurrente en la proyección de datos multi-dimensionales denominado *problema de aglomeración* (*crowding problem* en inglés), que consiste en la acumulación de múltiples instancias de datos en una región reducida del espacio de baja dimensionalidad producto de una transformación deficiente, lo que complica el análisis visual. En contraste con dichas técnicas, t-SNE aborda efectivamente este problema mediante el uso de una distribución de cola pesada en la organización de las instancias de datos en el espacio de baja dimensionalidad [393]. En términos prácticos, abordar el problema de aglomeración produce visualizaciones de mejor apariencia y más fáciles de explorar.

El propósito de máxima del cribado virtual de fármacos es facilitar el descubrimiento de nuevos compuestos candidatos, basando la búsqueda en similitud con ligandos de propiedades

bioquímicas y físico-químicas conocidas. Por lo tanto, idealmente las herramientas de visualización para cribado virtual deberían permitir a lxs expertxs analizar cómo interactúa o se relaciona química y espacialmente un compuesto nuevo con otros compuestos conocidos, a la vez permitiendo un uso interactivo.

Una limitación importante de muchos métodos de reducción dimensional propuestos radica en que son *no paramétricos*, es decir, son capaces de encontrar mapeos a baja dimensionalidad para un conjunto de datos específico, pero, una vez encontrado dicho mapeo, no permiten mapear nuevas instancias desde el espacio de alta dimensionalidad original al espacio latente. Por lo tanto, las herramientas de analítica visual que integran dichos métodos no permiten incorporar instancias nuevas de datos para un análisis contextual en un conjunto preexistente. Si bien t-SNE fue concebida inicialmente como una técnica no paramétrica, existe una versión paramétrica del algoritmo denominada pt-SNE [392], que busca superar esta limitación. No obstante, a menudo es difícil encontrar una configuración óptima de hiperparámetros para estos modelos, lo cual produce proyecciones ruidosas en comparación con las obtenidas utilizando la versión no paramétrica del algoritmo [259, 450, 437].

### 7.2.2. Visualización de estructuras moleculares

La visualización de información molecular es una de las ramas más antiguas del estudio de técnicas de visualización, dado que consta de una base de representaciones visuales establecida y ampliamente adoptada en bioquímica, biología y farmacia. A grandes rasgos, el formato de visualización estándar de estructuras moleculares consiste en la representación del grafo molecular por medio de glifos esféricos o circulares para los átomos, y glifos cilíndricos o rectas para los enlaces, variando colores de acuerdo a un código estandarizado. Dichas representaciones visuales, en su mayoría en 2D y 3D, se encuentran integradas en numerosas herramientas de visualización molecular disponibles, tales como PyMOL [334], VMD [171], Chimera [296] o YASARA [208]. Además de estas herramientas, también existen algunas herramientas web que pueden integrarse en otras aplicaciones; por ejemplo, Jmol [184], JSmol [146] o 3Dmol.js [308]. Sin embargo, ninguna de estas herramientas es directamente aplicable al problema de la exploración visual de grandes conjuntos de estructuras moleculares y la detección de patrones de similitud en su estructura y propiedades. Kozlíková et al. [204] presentaron una amplia revisión bibliográfica de los enfoques actualmente disponibles para visualización molecular.

Para realizar inspección visual de moléculas pequeñas se cuenta con algunas herramientas disponibles que permiten representar su estructura en 2D, como es el caso de *LigPlot+* [212]. Esta herramienta se utiliza principalmente para explorar las interacciones entre un ligando y su blanco terapéutico, proyectando su estructura molecular y los aminoácidos de la proteína en el plano 2D. Sin embargo, esta herramienta no permite analizar múltiples compuestos moleculares en simultáneo.

La exploración de grandes conjuntos de compuestos químicos constituye una tarea clave desde el punto de vista del análisis visual, por lo que existe una variedad de propuestas para ello. La herramienta más cercana a nuestra propuesta es *CheS-Mapper* [142], que permite cargar datos de varias bases de datos químicas, calcular descriptores moleculares, observar grupos de compuestos en un espacio reducido en 2D y mostrar la representación 3D de los compuestos. Sin embargo, la herramienta no proporciona una opción para comparar de forma interactiva diferentes proyecciones y no admite la inspección de las propiedades físico-químicas de un compuesto recién agregado. Considerando que, en palabras de sus autorxs, *CheS-Mapper* ha sido especialmente diseñada para la validación de modelos QSAR, carece de una serie de otras características deseables en el contexto de cribado virtual.

Una herramienta similar a *CheS-Mapper* es *Data Warrior* [326], que integra varios métodos básicos de visualización y vistas especializadas, permitiendo una visión general del espacio químico y de los farmacóforos del conjunto de datos de entrada. Sin embargo, la herramienta no brinda soporte para tareas específicas relacionadas con el proceso de descubrimiento de fármacos en etapas tempranas, que sí son abordadas por nuestra herramienta. Por su parte, Yoshimori et al. [442] propusieron una técnica para condensar visualmente la información de múltiples matrices de relaciones estructura-actividad, basada en mapas moleculares y vistas tridimensionales de actividad. Sin embargo, debido a su especificidad, no brinda suficiente flexibilidad como para incorporar propiedades y descriptores físico-químicos adicionales.

Janssen et al. [180] propusieron un método que utiliza un mapeo visual basado en proyecciones t-SNE para encontrar nuevos inhibidores potenciales de quinasas [60]. Las vistas incorporadas en la herramienta muestran los resultados de los agrupamientos entre compuestos por medio de una estructura en forma de árbol de los mismos. Sin embargo, estas vistas no permiten extraer información precisa sobre la similitud estructural entre dos compuestos de forma intuitiva. Probst and Reymond [303] introdujeron una propuesta similar orientada a la codificación de conjuntos grandes de compuestos por medio de estructuras en forma de árbol. Su enfoque está diseñado para procesar conjuntos de datos muy grandes y de alta dimensionalidad, donde la similitud entre compuestos se expresa en su proximidad a través de las ramas del árbol.

Naveja and Medina-Franco [278] introdujeron el concepto de *gráficos de constelaciones* de compuestos químicos, donde los mismos son agrupados según un esqueleto central compartido entre las proyecciones 2D de sus estructuras moleculares. Cada agrupamiento es anotado por medio de una vista reducida del esqueleto proyectado. Si bien estos gráficos de constelaciones proporcionan un enfoque innovador para la visualización de compuestos, no cumplen con muchos de los requisitos para la exploración visual en el contexto del cribado virtual, proporcionando únicamente medios para la identificación de subestructuras comunes. *Synergy Maps* [220] es otra aplicación basada en herramientas web que permite explorar las relaciones entre compuestos y comprender las sinergias potenciales entre ellos. La herramienta muestra combinaciones de propiedades de compuestos de pares utilizando una representación en forma de red. Sin embargo, los diagramas de nodos y enlaces no escalan adecuadamente ante conjuntos grandes de datos, lo que dificulta su aplicación para la exploración de grandes quimiotecas.

Aunque se han propuesto métodos para evaluar la confiabilidad de las técnicas de reducción dimensional [21, 80, 294], tras una revisión exhaustiva de la literatura no hemos hallado enfoques preexistentes para la visualización de estructuras moleculares que permitan este tipo de análisis. Además, ninguna de las herramientas propuestas previamente brinda medios para comparar los resultados de múltiples proyecciones en baja dimensionalidad. En síntesis, las herramientas aquí descritas presentaron una serie de limitaciones, las cuales constituyen requisitos en una herramienta de cribado virtual. Entre ellas, destacamos la necesidad de analizar múltiples compuestos moleculares en simultáneo, de contrastar diferentes proyecciones 2D, de permitir la adjunción y el análisis de compuestos nuevos a una proyección, y de permitir el estudio intuitivo de múltiples propiedades físico-químicas de forma integrada, todo en un entorno apto para grandes conjuntos de datos. Estos desafíos fueron abordados por *ChemVA*.

### 7.3. Marco teórico

En esta sección, proporcionamos detalles sobre el marco teórico que da sustento a nuestra herramienta, en el que se incluye una breve explicación de las representaciones moleculares utilizadas y su semántica, así como una enumeración de las características moleculares consideradas por su relevancia para el análisis del perfil farmacológico de compuestos candidatos (las cuales han sido desarrolladas en mayor profundidad en el capítulo 3 de esta tesis). Además, presentamos de manera sucinta el modelo de reducción de dimensionalidad aplicado en *ChemVA*.

### 7.3.1. Representaciones moleculares basadas en vectores

De acuerdo al estado del arte en investigación en química informática con respecto a similitud molecular, existen varios factores diferentes involucrados en la evaluación de la similitud de compuestos químicos, que van más allá de tener estructuras moleculares similares, es decir, en términos de sus disposiciones geométricas de átomos y enlaces [234, 31]. Dada la relevancia de evaluar la similitud de compuestos para el desarrollo de cribado virtual de fármacos, es importante que las herramientas desarrolladas para este fin brinden a lxs expertxs en química medicinal una variedad de representaciones moleculares basadas en vectores que capturen diferentes aspectos de los compuestos en estudio y que sean complementarias entre sí. En este sentido, *ChemVA* proporciona cuatro representaciones moleculares diferentes, que luego utilizamos como fuentes de datos para nuestras proyecciones bidimensionales, todas ellas detalladas en el capítulo 3.

Para cada conjunto de datos, calculamos *fingerprints* ECFP de radio 2 de 1.024 bits y *fingerprints Daylight* de 1.024 bits, utilizando los paquetes *Chem* y *AllChem* de RDKit [132]. También calculamos descriptores moleculares de 0D, 1D y 2D por medio de la librería Mordred [268], obteniendo un total de 1.613 descriptores. Eliminamos todos los descriptores que presentaran más del 10% de valores *NaN* y reemplazamos todos los *NaN* restantes con un valor fijo (el valor máximo exhibido por cada descriptor), lo que resultó en 1.454 descriptores moleculares. Por último, empleamos un modelo *Mol2Vec* preentrenado [176] para el cálculo de *embeddings* moleculares de 300 dimensiones a partir de las fórmulas SMILES de los compuestos. El cálculo de ECFPs, *fingerprints Daylight* y descriptores moleculares se realizó en un clúster de 32 núcleos CPU con 12 GB de memoria RAM. Esta etapa de preprocesamiento se realizó *offline*, lo que significa que se realizó una única vez para los conjuntos de datos vinculados a los casos de estudio, y los resultados obtenidos fueron almacenados para su uso posterior en las proyecciones bidimensionales.

Seleccionamos estas cuatro representaciones moleculares basadas en vectores como nuestras fuentes de datos para *ChemVA*, puesto que capturan información diferente y complementaria sobre los compuestos, lo que facilita un análisis amplio de los datos en estudio, y porque constituyen representaciones moleculares ampliamente estudiadas y empleadas en trabajos de investigación en la materia. No obstante, si bien *ChemVA* emplea estas cuatro representaciones moleculares, es importante destacar que la herramienta es independiente de las fuentes de datos y representaciones elegidas.

### 7.3.2. Características moleculares relacionadas con *drug-likeness*

Algunas características moleculares son de gran interés para los diseñadores de fármacos y expertos en química medicinal en el contexto del cribado virtual. En particular, tal y como hemos discutido en el capítulo 3, existen características moleculares asociadas a la afinidad de un compuesto candidato con las características deseables en un fármaco (*drug-likeness*), que permiten examinarlos en términos de su viabilidad como posibles nuevos fármacos. Para el desarrollo de *ChemVA* tuvimos en cuenta un subconjunto de estas características, todas ellas descritas en profundidad en el capítulo 3:

- Peso molecular (MW)
- Logaritmo del coeficiente de partición octanol-agua ( $\log P$ )
- Constante de disociación ácida ( $K_a$ ) y Constante de disociación básica ( $K_b$ )
- Regla de los Cinco de Lipinski (*Lipinski's RO5*)
- Puntaje QED (*Quantitative Estimate of Drug-likeness*)

*ChemVA* muestra este conjunto de características en una variedad de vistas interactivas, con el fin de proporcionar a los usuarios información complementaria sobre el perfil farmacológico de los compuestos estudiados. Los valores de estas características moleculares para los compuestos de los casos de estudio fueron obtenidos por medio de una API disponible en línea provista por la quimioteca *ChEMBL* [116].

## 7.4. Requisitos de nuestra herramienta de analítica visual

En el cribado virtual, es crucial para los expertos comprender adecuadamente los atributos y características moleculares del conjunto de datos. Además, el hecho de poder contrastar datos conocidos y previamente estudiados con nuevos compuestos, cuyas propiedades o perfiles de bioactividad pueden ser inciertos o desconocidos, tiene el potencial de mejorar ampliamente el proceso de diseño de fármacos en etapas tempranas. Entre las tareas comúnmente ejecutadas en el contexto del cribado virtual se encuentran la estimación y evaluación de las propiedades químicas de los compuestos, el análisis de su comportamiento con respecto al conjunto de datos y el análisis de la similitud entre compuestos.

A lo largo de un año, nuestro equipo de desarrollo de *ChemVA* llevó a cabo numerosas entrevistas con el grupo de ingenierxs de proteínas de los Laboratorios Loschmidt en la Universidad Masaryk, República Checa. Basándonos en sus aportes, identificamos varias limitaciones de los enfoques existentes para el cribado virtual y las resumimos en una lista de requisitos. Lxs expertxs estuvieron de acuerdo en que los requisitos identificados y objetivos delineados abarcan los aspectos más críticos de un flujo de trabajo de cribado virtual. Unx de ellxs, quien es también coautorx del artículo que da sustento al presente capítulo [323], utilizó y evaluó exhaustivamente la herramienta en calidad de usuarix expertx, a fin de asegurar que nuestra implementación cumple con los requisitos, revisando y comentando iterativamente sobre el progreso. Su experiencia de investigación de cuatro años en ingeniería de proteínas junto con la evaluación conjunta por parte de su grupo de investigación proporcionaron el conocimiento de dominio necesario para el diseño de la herramienta.

**R1: Visión general y análisis detallado de un conjunto molecular en un espacio de baja dimensionalidad.** Para grandes conjuntos de datos, los diagramas de dispersión comúnmente utilizados para representar los resultados de la proyección en baja dimensionalidad pueden sufrir problemas de oclusión, donde varios compuestos se solapan entre sí impidiendo visualizar correctamente sus posiciones en el espacio. Por lo tanto, la herramienta debe proporcionar soporte visual para el análisis de datos en diferentes niveles de abstracción, desde la distribución general de los compuestos dentro del espacio 2D hasta una visión en detalle de compuestos individuales para una región de interés seleccionada.

A fin de abordar este requisito en *ChemVA*, integramos dos gráficos 2D coordinados. En primer lugar, diseñamos una *vista Hexagonal*, descrita en la sección 7.5.2, que proporciona una visión general de la distribución de los compuestos en un espacio de baja dimensionalidad dado y también sirve para la navegación general en el conjunto de datos. En segundo lugar, desarrollamos una *vista Detallada*, descrita en la sección 7.5.2, que permite a lx usuarix explorar las características de los compuestos seleccionados en la vista Hexagonal con mayor detalle.

**R2: Inspección visual de múltiples proyecciones en simultáneo.** Un conjunto de compuestos candidatos puede representarse mediante diferentes representaciones moleculares basadas en vectores, cada una de las cuales produce una proyección en baja dimensionalidad diferente. La herramienta de analítica visual a implementar debe permitir a lx usuarix combinar intuitivamente la información codificada en las distintas proyecciones individuales a fin de posibilitar el estudio de la similitud entre compuestos, teniendo en cuenta diferentes semánticas de representación molecular en simultáneo. Esto incluye el explorar similitudes y diferencias entre compuestos químicos expresados por medio de diferentes proyecciones, donde la similitud pueda entenderse desde un punto de vista

de proximidad entre compuestos en el espacio de baja dimensionalidad. Además, las representaciones visuales e interacciones provistas por la herramienta deben ayudar a lxs expertxs en el dominio a evaluar la idoneidad del modelo de reducción dimensional seleccionado.

Este requisito fue principalmente abordado mediante el diseño de las vistas hexagonales (descriptas en la sección 7.5.2) y nuestra propuesta novedosa denominada *vista de Contraste* (*Difference view* en inglés), descrita en la sección 7.5.2, que permite la comparación de dos proyecciones 2D codificando la distribución de las aglomeraciones locales de compuestos de una proyección en la otra.

**R3: Evaluación de la confiabilidad de las proyecciones 2D.** Partiendo de la presunción del principio de localidad que dice que dos compuestos estructuralmente similares presentan características moleculares similares, lxs usuarixs requieren de un soporte visual adecuado para evaluar la confiabilidad de una proyección de baja dimensionalidad basada en la distorsión con respecto a las distancias entre compuestos en el espacio original, de alta dimensionalidad. Además, la herramienta debe permitir la comparación de la confiabilidad de diferentes proyecciones 2D, lo que permite a lxs usuarixs centrarse en un subconjunto de representaciones moleculares que resulte más confiable o adecuado en el caso de estudio bajo análisis.

Para abordar este requisito, expresamos la confiabilidad mediante dos *puntajes de correlación* asociados a la confiabilidad de la proyección 2D individual de cada compuesto, que se codifican visualmente en todas las vistas 2D y también se presentan en una *vista Tabular* para su posterior comparación con otras características moleculares.

**R4: Comparación entre compuestos según características moleculares relacionadas con su perfil farmacológico.** Los compuestos químicos son descritos por numerosas características y descriptores moleculares relacionados con su idoneidad para cumplir funciones farmacológicas. Resulta deseable proporcionar a lxs usuarixs la opción de visualizar estas características adicionales junto con otras representaciones moleculares complementarias.

Para cumplir con este requisito, ofrecemos una *vista Tabular*, descrita en la sección 7.5.3, que muestra información detallada sobre características seleccionadas de los compuestos (ver sección 7.3.2) y permite su exploración interactiva. Estas características moleculares también han sido integradas en las vistas 2D para brindar una visión general de la distribución espacial y posibles agrupamientos entre compuestos en términos de las mismas.

**R5: Visualización de la similitud estructural en 3D.** Una herramienta de analítica visual para cribado virtual debe admitir la inspección de compuestos individuales en términos de su geometría



3D, así como también la inspección visual de subestructuras 3D comunes entre compuestos de un conjunto seleccionado. En particular, al estudiar un grupo de compuestos, dicha vista debe transmitir la información sobre similitudes y diferencias en sus estructuras 3D.

Nuestra solución para este requisito es una *vista 3D*, descrita en la sección 7.5.4, que admite la exploración de la similitud geométrica entre los compuestos seleccionados mediante la aplicación de un alineamiento estructural de las conformaciones 3D más probables de los compuestos bajo análisis.

**R6: Soporte para la incorporación de nuevos compuestos y su contraste con datos existentes.** Suponiendo que la herramienta de analítica visual haya permitido la identificación de patrones y características moleculares de interés que den lugar al diseño de nuevos fármacos, es deseable que la misma admita la incorporación de dichos compuestos al conjunto de datos bajo análisis, posibilitando una exploración contextual de sus diferentes características moleculares y la estimación de su perfil de bioactividad. En tal caso, los nuevos compuestos deben ser proyectados utilizando el modelo de reducción de dimensionalidad y luego integrados en todas las vistas de la herramienta, para que el usuario pueda comparar sus características moleculares con las de los compuestos en el conjunto de datos existente.

Este requisito es abordado por *ChemVA*, permitiendo cargar nuevos compuestos, proyectarlos en las diferentes vistas y realizar un análisis visual de sus propiedades (ver sección 7.6).

## 7.5. Diseño e implementación de *ChemVA*

Siguiendo los requerimientos delineados anteriormente, el proceso de diseño de *ChemVA* consistió en la definición de un esquema de distribución de vistas coordinadas, el contenido de dichas vistas y los mecanismos de interacción con las mismas. Entre los desafíos abordados durante el diseño e implementación de nuestra herramienta tuvimos en consideración cuestiones de accesibilidad; facilidad de uso y familiaridad con respecto a herramientas preexistentes, considerando que se trata de una herramienta de soporte a expertos; escalabilidad, permitiendo la visualización de conjuntos grandes de datos químicos; interactividad, apuntando a minimizar las latencias de respuesta de la herramienta en línea; y flexibilidad, con respecto a brindar funcionalidades propias del proceso de cribado virtual de fármacos, pero también contemplando otros posibles usos, como diseño *de novo*.

*ChemVA* fue desarrollado en JavaScript utilizando D3.js v5 [47] y un servidor usando el entorno Node.js [376] construido sobre el framework Express y una API REST para la conexión a los servicios de soporte. Para el desarrollo de los componentes en 3D, utilizamos Unity3d [369] portado a WebGL

[195]. Dos servicios web backend, utilizados para el cálculo del alineamiento de estructuras 3D y para el cálculo de las coordenadas 2D de compuestos añadidos recientemente, fueron desarrollados utilizando el framework web Flask [314] e implementados en Python v3. Con excepción de la vista Tabular [112], todas las funcionalidades fueron diseñadas y desarrolladas por nuestro equipo de trabajo, priorizando la capacidad de respuesta del sistema y la experiencia en tiempo real con las interacciones. La herramienta utiliza el esquema de colores propuesto por Ichihara et al. [172], por lo que es accesible para usuarios con discromatopsia (diferencias o deficiencias en la percepción visual de los colores).

En esta sección del capítulo primeramente describimos en detalle el esquema de distribución de vistas y el diseño general de *ChemVA*, para luego dar lugar a una explicación detallada de cada una de las vistas de la herramienta. En segunda instancia, brindamos detalles del modelo de reducción de dimensionalidad paramétrico desarrollado para nuestra herramienta y de la estrategia de cómputo de los puntajes de correlación para la estimación de confiabilidad de las proyecciones 2D. Por último, presentamos la funcionalidad que sustenta la incorporación de nuevos compuestos químicos para diseño *de novo*. El código fuente desarrollado en el marco del trabajo de investigación presentado en este capítulo está públicamente disponible<sup>3</sup>.

### 7.5.1. Diseño general y esquema de distribución de vistas

El diseño general de nuestra herramienta consiste en un panel rectangular dividido en filas, cada una de las cuales contiene diferentes vistas interactivas. Tal y como se presenta en la figura 7.1, el diseño preliminar de *ChemVA* consta de dos filas: la fila superior contiene una vista Hexagonal, una vista Detallada y una vista 3D (figura 7.1 A, B y C, respectivamente), mientras que la fila inferior contiene la vista Tabular (figura 7.1 D). Estas vistas cumplen con los requisitos R1, R4 y R5, enumerados anteriormente (ver sección 7.4).

Aquellas tareas que implican la comparación de proyecciones 2D que dan soporte al requisito R2 demandan la incorporación de vistas adicionales. Estas vistas se ubican en una tercera fila, ubicada entre las filas superior e inferior, tal y como se muestra en la figura 7.9. Esta fila se divide en tercios y, a diferencia de las dos anteriores, puede mostrarse o colapsarse a demanda de lx usuario. Además, es personalizable permitiendo a lx usuario decidir qué vistas interactivas incluir en ella de entre las siguientes opciones: hexagonal, detallada y de contraste.

---

<sup>3</sup>Código fuente: <https://github.com/VirginiaSabando/ChemVA>

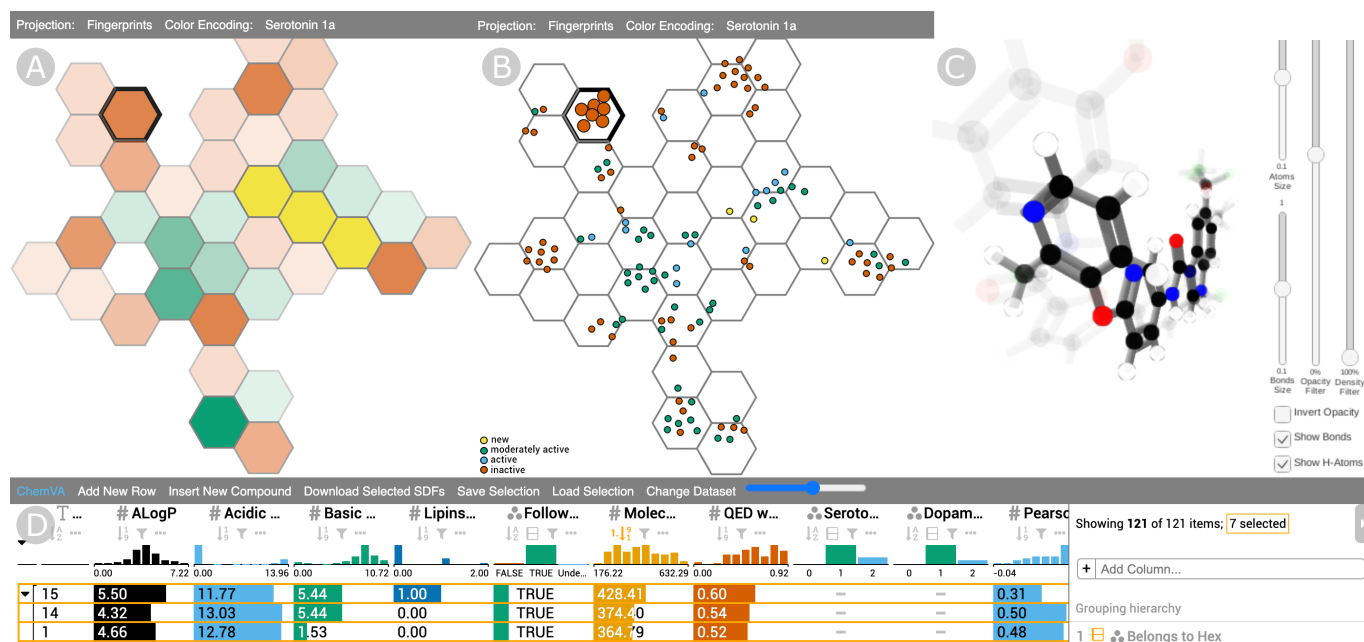


Figura 7.1: Descripción general de la interfaz preliminar de *ChemVA*: a) **vista Hexagonal**, que proporciona una visión general de una proyección 2D seleccionada, b) **vista Detallada**, que muestra todos los elementos de datos, que están coloreados según una característica molecular seleccionada, c) **vista 3D** que permite a lx usuario observar las similitudes en la estructura molecular 3D de compuestos seleccionados, d) **vista Tabular**, que permite listar características moleculares relacionadas con el perfil farmacológico de los compuestos, además de operaciones como ordenamiento y filtrado. Todas las vistas operan de forma coordinada y soportan interacción con lx usuario.

El propósito de este esquema de distribución de vistas es permitir a lx usuario observar los conjuntos de datos en estudio desde diferentes perspectivas, proporcionando una amplia variedad de información sobre el conjunto de datos en distintos niveles de abstracción, desde vistas generales hasta exploraciones detalladas de moléculas seleccionadas y su estructura 3D. En particular, la vista de Contraste muestra las diferencias entre dos proyecciones 2D y permite identificar cómo un conjunto de compuestos dentro de una proyección se encuentra diseminado o agrupado dentro de otra, por lo que resulta de especial utilidad contar con vistas hexagonales o detalladas a la par de una vista de Contraste, tal y como se muestra en la figura 7.9.

### 7.5.2. Vistas 2D

Las vistas 2D constituyen el núcleo de *ChemVA*, brindando a lx usuario una visión general de la distribución de los compuestos candidatos utilizando una representación molecular determinada

en una proyección 2D. La vista Hexagonal es la que da soporte a esta visión general del conjunto de datos [340]. La vista Hexagonal tiene como objetivo mitigar el problema de superposición de datos, en el que los elementos de datos proyectados se superponen y causan desorden visual, lo que puede limitar la interpretación de la información. En la vista Hexagonal de *ChemVA*, este problema se resuelve por medio de la agregación de grupos de compuestos en hexágonos. Lx usuariix puede seleccionar de forma interactiva un subconjunto de hexágonos de interés y explorar la distribución de los compuestos individuales comprendidos por ellos dentro de la *vista Detallada*. La combinación de la *vista Hexagonal* y la *vista Detallada* tiene como objetivo cumplir con el requisito R1, detallado en la sección 7.4. Finalmente, dado que *ChemVA* ha sido diseñada para dar soporte a la comparación visual de diferentes proyecciones 2D, también ofrece la vista de Contraste. Esta vista fue diseñada específicamente para abordar esa tarea, que se menciona en los requisitos R2 y R3.

Para evitar introducir sesgos no deseados en el proceso de análisis por parte de lx expertx, no aplicamos ningún método de agrupamiento (*clustering*) posterior al paso de reducción dimensional. La razón principal es que cada algoritmo de agrupamiento se configura y entrena a partir de parámetros adicionales que, evidentemente, incidirían en la detección de grupos de compuestos, con el potencial de sesgar la interpretación por parte de lx usuariix si lx mimsx no comprendiera en profundidad el método de agrupamiento utilizado.

#### 7.5.2.1. Vista Hexagonal

La *vista Hexagonal* (figura 7.2) brinda a lx usuariix una visión general de una proyección bidimensional particular del conjunto de datos. Para evitar la superposición de datos al tratar con conjuntos de datos grandes, los compuestos químicos son agregados en celdas hexagonales. Optamos por el enfoque de agrupamiento hexagonal, descrito por primera vez por Carr et al. [57], ya que ofrece ventajas significativas para la agregación eficiente de datos en comparación con otros enfoques. Esto está relacionado con el bajo ratio entre perímetro y área de los hexágonos regulares, lo que reduce el sesgo de muestreo. La elección de hexágonos regulares como figura geométrica para teselación de planos es un enfoque ampliamente adoptado y comúnmente utilizado para visualizar el resultado de técnicas de reducción dimensional, puesto que el hexágono regular es la figura geométrica más cercana a un círculo utilizable para la teselación regular de un plano, sin espacios libres y sin superposición.

La vista Hexagonal se puede aplicar a cualquiera de las representaciones moleculares basadas en vectores descritas en la sección 7.3.1. Cada hexágono aparece coloreado con una determinada opacidad, que se corresponde con el número de compuestos agregados dentro del hexágono. Una mayor opacidad indica un mayor número de compuestos dentro del hexágono. De esta manera,

la vista Hexagonal muestra la distribución de los compuestos obtenida mediante el paso de reducción dimensional, lo que permite a lx expertx reconocer a simple vista áreas con una alta densidad de compuestos. El color de un hexágono codifica la tendencia predominante entre sus compuestos para una característica molecular seleccionada: por defecto, el valor codificado es su perfil de bioactividad (activo vs. inactivo, según el caso de estudio), pero puede cambiarse a otras propiedades o características moleculares, incluyendo los puntajes de correlación que permiten estimar la confiabilidad de las proyecciones 2D (requisito R3). Si un hexágono contiene compuestos con diferentes valores para una determinada característica, el color se selecciona en función del valor predominante en el conjunto (en caso de valores categóricos) o del valor promedio (en caso de propiedades cuantitativas).

La granularidad de la vista Hexagonal se puede modificar mediante un control deslizante, lo que permite una visión más detallada de la distribución de los compuestos. Al operar el control deslizante, el tamaño de las celdas hexagonales en la grilla aumenta a derecha y disminuye a izquierda, lo que da lugar a una teselación del plano 2D en menos o más celdas hexagonales, respectivamente. Para mantener la legibilidad a través de diferentes niveles de granularidad, la opacidad de las celdas hexagonales se incrementa linealmente con respecto a la disminución del tamaño del hexágono. Esta prestación permite estudiar en mayor detalle las regiones pequeñas de la proyección, a la vez que la representación final mantiene un contraste entre las celdas hexagonales más pobladas y las menos pobladas. Además, la vista Hexagonal permite a lx usarix seleccionar varios hexágonos a la vez por medio del cursor. Dado que las vistas de *ChemVA* están coordinadas, los compuestos dentro de las celdas hexagonales seleccionadas son filtrados en las vistas vinculadas (vista Detallada, vista 3D y vista tabular), donde se pueden explorar con mayor detenimiento.

#### 7.5.2.2. Vista Detallada

Al seleccionar un subconjunto de compuestos en la vista Hexagonal, lx usarix puede explorar los datos seleccionados en la vista Detallada, representada en la figura 7.3. En esta vista, los compuestos pueden ser visualizados en un gráfico de dispersión (*scatterplot*), al cual se le yuxtapone una sutil teselación del plano 2D en celdas hexagonales idéntica a la mostrada en la vista Hexagonal, de forma tal de mantener la correspondencia entre los niveles de zoom en sendas vistas. Este aspecto resulta de particular importancia porque, una vez realizada una selección de celdas hexagonales en la vista Hexagonal, la vista Detallada muestra solamente los compuestos contenidos dentro de las celdas seleccionadas.

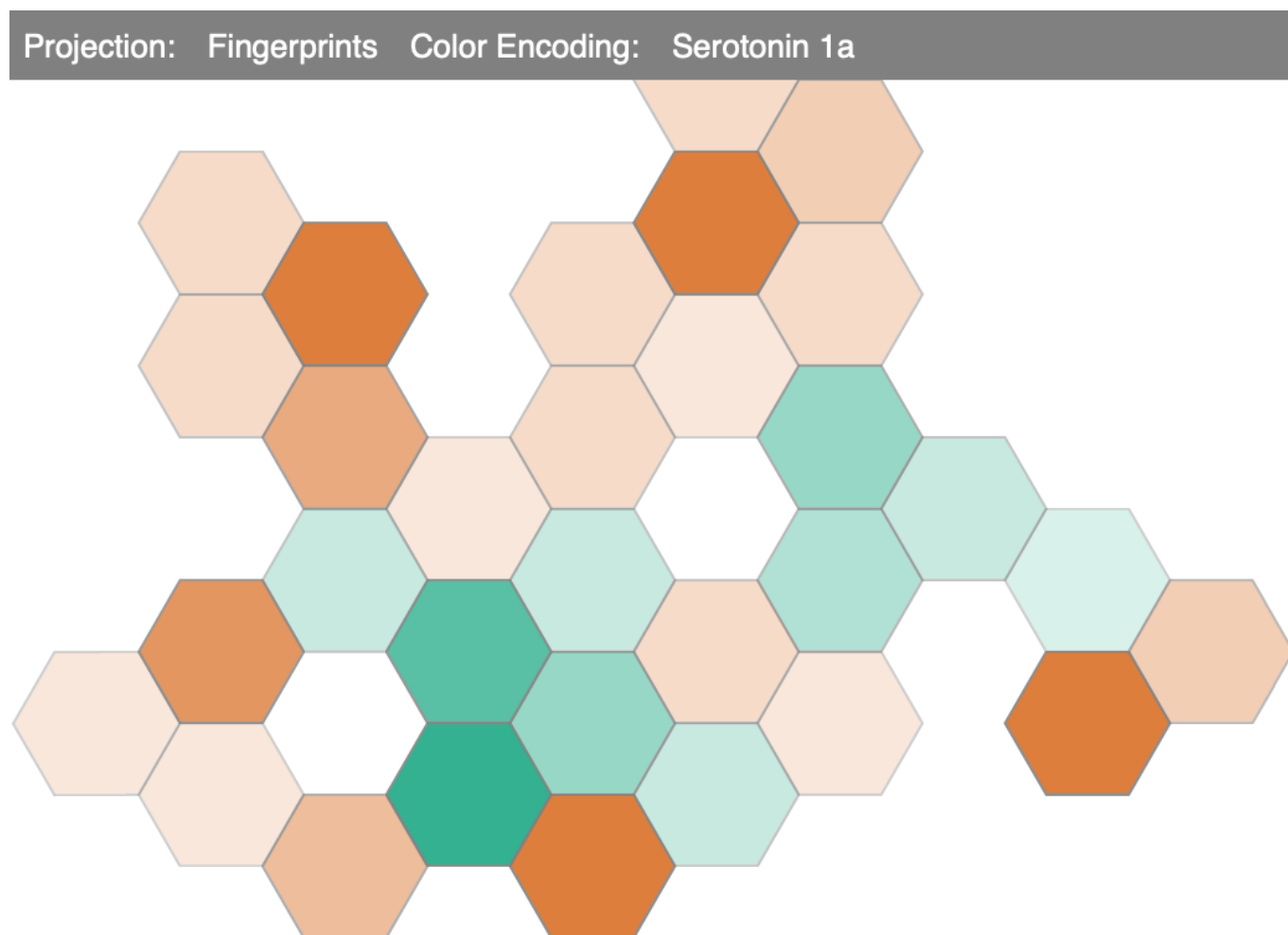


Figura 7.2: La vista Hexagonal muestra la densidad de la distribución de compuestos en la proyección 2D, la cual se codifica mediante la opacidad, y la prevalencia de la bioactividad de los compuestos, la cual se codifica mediante el color. El tamaño de las celdas hexagonales puede ajustarse para permitir una inspección más detallada de la información proyectada en la vista.

Las vistas hexagonal y detallada se hallan coordinadas, de forma tal que al pasar el cursor sobre una celda hexagonal en cualquiera de ambas vistas, dicha celda es resaltada en la vista complementaria. Desarrollamos un operador de selección de lazo (*lasso selection* en inglés) para seleccionar compuestos individuales y grupos de compuestos en esta vista. Una vez seleccionados, dichos compuestos se muestran en la vista 3D y también son destacados en la vista Tabular.

Al igual que en el caso de la vista Hexagonal, la vista Detallada puede ser configurada para mostrar cualquiera de las representaciones moleculares basadas en vectores descriptas en la sección 7.3.1. Además, todas las propiedades y características moleculares descriptas en la sección 7.3 pueden

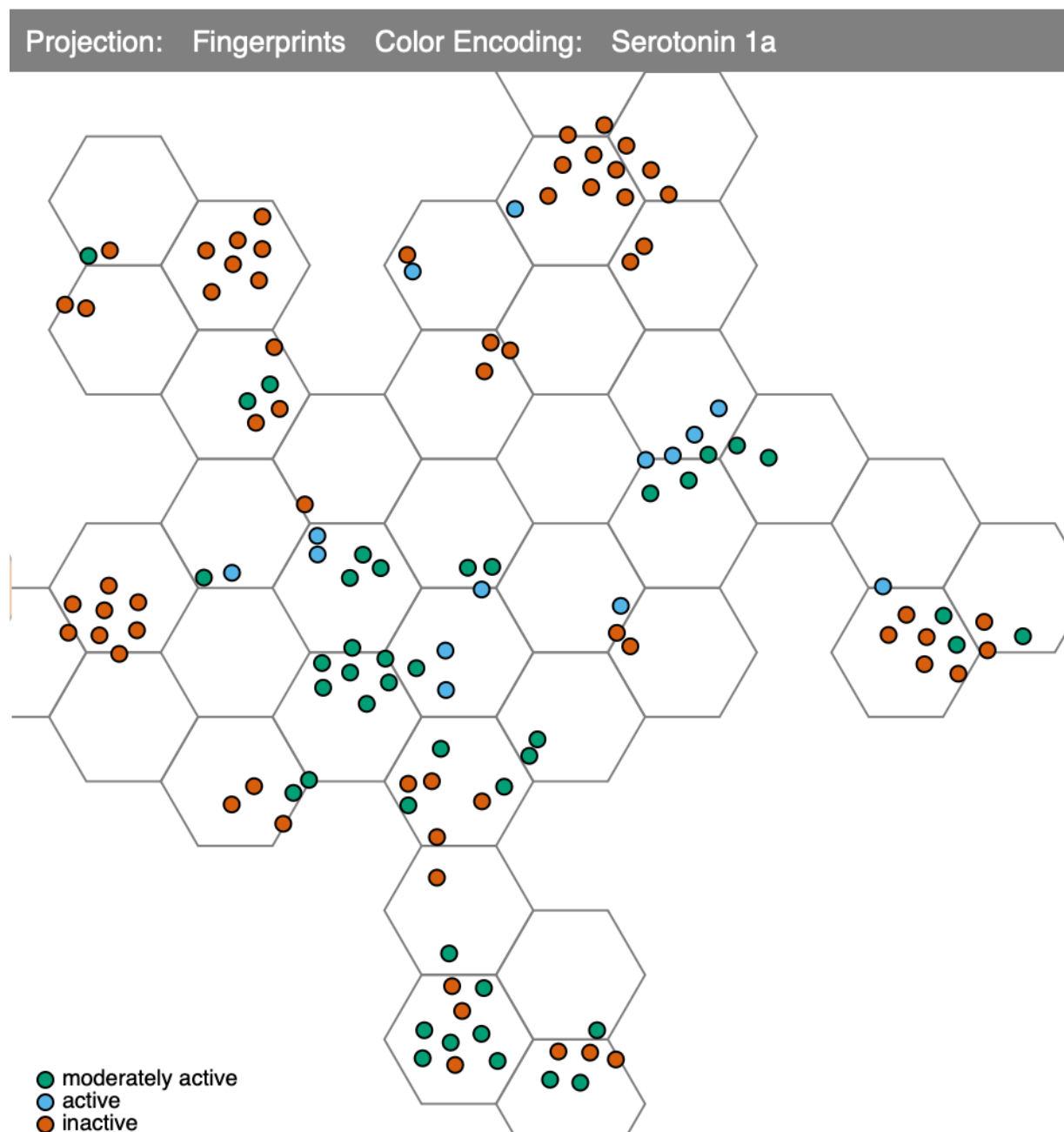


Figura 7.3: La *vista Detallada* muestra cada uno de los compuestos individuales como puntos en una proyección 2D seleccionada (en este caso, *fingerprints* ECFP). El mapeo de color de los puntos puede hacerse a diferentes características moleculares. En este caso, se muestra un mapeo correspondiente con el perfil de bioactividad de los compuestos con respecto al receptor Serotonina 1a.

ser codificadas por color en los puntos que representan cada compuesto. Estas características se seleccionan desde un menú desplegable, y su codificación de color se elige según su tipo, ya sea

cuantitativo, como el peso molecular, o categórico, como el perfil de bioactividad del compuesto con respecto a un blanco terapéutico. Otra propiedad cuantitativa que se puede utilizar para la codificación de color corresponde a los *puntajes de correlación*, que indican la confiabilidad de la proyección en 2D del compuesto (según establecimos en el requisito R3). El procedimiento que seguimos para computar dichos puntajes de correlación se detalla en la sección 7.5.6.

### 7.5.2.3. Vista de Contraste

Los gráficos de dispersión o *scatterplots*, tales como el empleado en la vista Detallada, son la técnica más comúnmente utilizada para visualizar los resultados de los procesos de reducción de dimensionalidad en grandes conjuntos de datos [340]. Sin embargo, al trabajar con múltiples representaciones vectoriales diferentes de los compuestos químicos, las visualizaciones obtenidas una vez transcurrido el proceso de reducción dimensional pueden ser muy diferentes, dificultando el análisis comparativo de la distribución espacial de los compuestos.

Con el fin de permitir la comparación de los resultados de diferentes proyecciones bidimensionales, tal y como se establece en nuestro requisito R2, propusimos una estrategia de visualización novedosa denominada vista de Contraste, la cual puede apreciarse en las figuras 7.4 y 7.5. Dadas dos vistas hexagonales 2D *A* y *B*, la vista de Contraste permite combinar y contrastar la distribución espacial de los compuestos entre ambas, proporcionando información valiosa para el análisis de similitudes y diferencias entre los resultados de las distintas representaciones moleculares. En particular, la vista de Contraste ilustra la fragmentación de los vecindarios o agrupamientos de compuestos de una proyección en otra, lo que permite a lx expertx del dominio evaluar en mayor detalle los determinantes de agrupamiento para los compuestos candidatos en estudio.

Inicialmente, la vista de Contraste muestra un diseño hexagonal similar al presentado en la vista Hexagonal, donde la opacidad de cada hexágono codifica el puntaje de correlación calculado a partir de las proyecciones 2D *A* y *B* (según el requisito R3). La vista de Contraste adopta el diseño hexagonal de una de las proyecciones a comparar, denominada *proyección de referencia*. Cuando elegimos la proyección *A* como referencia y realizamos una operación de selección en la vista Hexagonal de *A*, se identifican las posiciones de los compuestos químicos seleccionados en la proyección comparada *B*. Estas posiciones se codifican como hexágonos internos más pequeños inscriptos en la cuadrícula hexagonal original de la vista de Contraste, como se muestra en la figura 7.4. Ante la presencia de múltiples compuestos seleccionados de *A* en *B* en una misma celda hexagonal, el tamaño del hexágono inscripto en dicha celda aumenta a fin de señalar una mayor densidad de compuestos. En



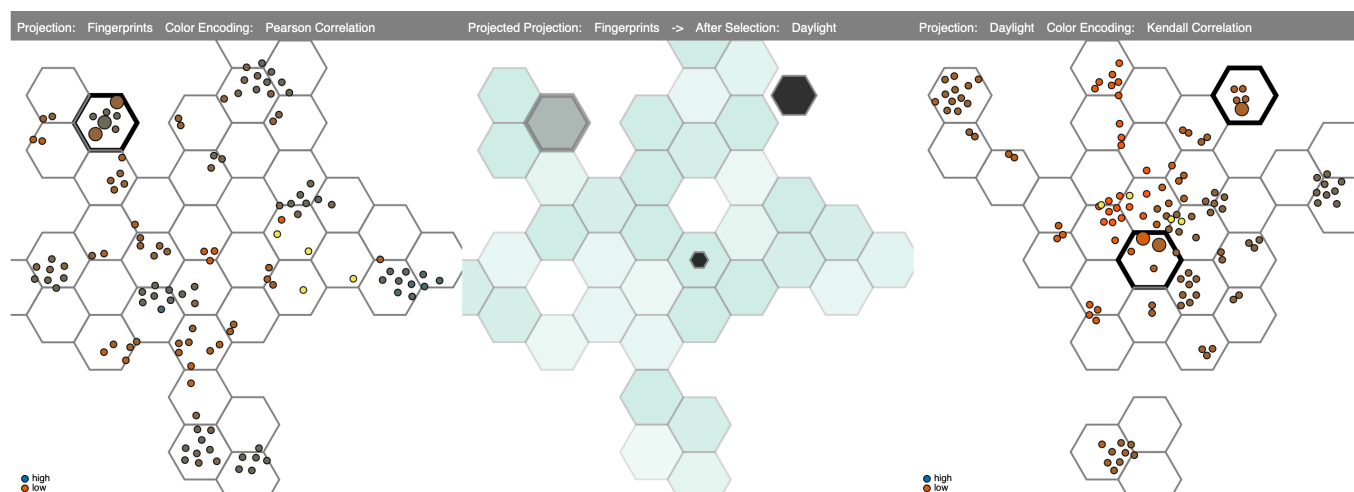


Figura 7.4: La vista de Contraste muestra la fragmentación de los agrupamientos de compuestos químicos de una proyección a otra. Aquí, la proyección de referencia *A* se basa en *fingerprints* ECFP; las celdas hexagonales seleccionadas en dicha proyección aparecen resaltadas en color gris en la vista de Contraste. La proyección comparada *B*, por su parte, corresponde a *fingerprints* Daylight; los compuestos seleccionados a partir de la proyección de referencia *A* son destacados en la proyección comparada *B* y visualizados en la vista de Contraste por medio de hexágonos inscriptos en negro. La opacidad de las celdas hexagonales en *A* codifica el valor de la métrica utilizada para la confiabilidad de la proyección (ver sección 7.5.6).

otras palabras, el tamaño de las celdas hexagonales inscriptas corresponde al número de compuestos que caen en la misma celda hexagonal.

El propósito principal de este gráfico es ayudar a ilustrar qué regiones del conjunto de datos conservan sus vecindarios o agrupamientos de compuestos al cambiar de una representación molecular basada en vectores a otra, habiendo ambas representaciones atravesado un procedimiento de reducción de dimensionalidad. De esta manera, *lx* *user* puede comparar rápidamente dos proyecciones diferentes y evaluar la confiabilidad de los vecindarios generados por las técnicas de reducción dimensional. Si el nivel de fragmentación de una proyección a otra es alto, es decir, se observa una segregación del contenido de un hexágono en muchos hexágonos pequeños, bastante separados y dispersos, se puede inferir que estas moléculas se comportan de manera diferente según la representación molecular elegida.

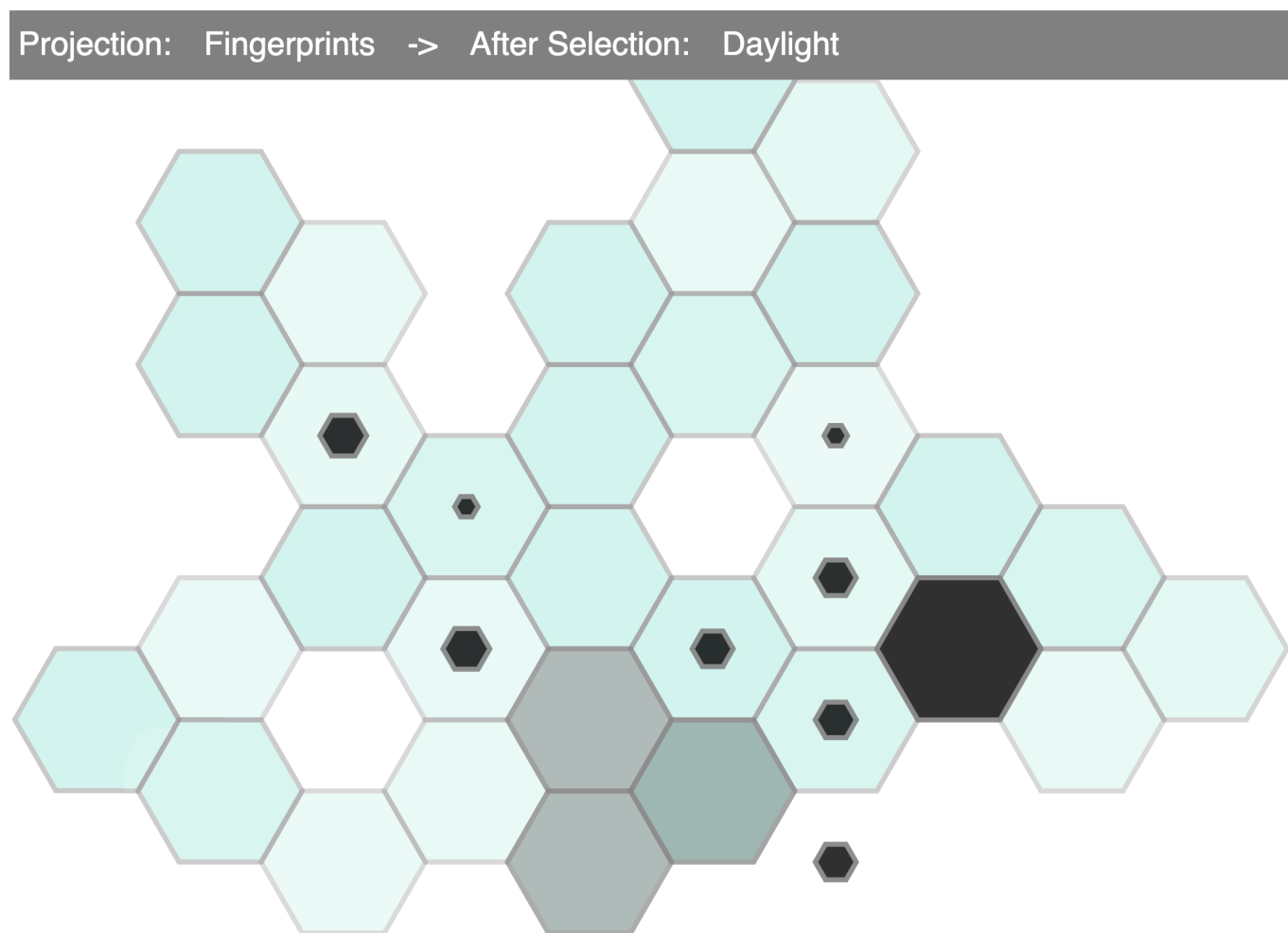


Figura 7.5: Acercamiento de la vista de Contraste mostrando una selección de compuestos diferentes para las mismas proyecciones 2D que las ilustradas en la figura 7.4.

### 7.5.3. Vista Tabular

Además de las representaciones moleculares basadas en vectores utilizadas en las proyecciones 2D, empleadas en las vistas hexagonal, detallada y de contraste, existen numerosas características moleculares relacionadas con el perfil farmacológico de los compuestos que es importante tener en cuenta durante su análisis, tal y como se establece en el requisito R4. *ChemVA* permite a los usuarios explorar tales características, enumeradas en la sección 7.3.2, a través de una vista Tabular que ofrece múltiples opciones de interacción avanzadas. Adoptamos una herramienta previamente desarrollada y establecida, publicada por Gratzl et al. [131], y una extensión desarrollada para la misma [112]. Estas herramientas fueron seleccionadas y adaptadas especialmente para *ChemVA*, dado que admiten múltiples interacciones avanzadas y contribuyen al proceso cognitivo de análisis de datos químicos.

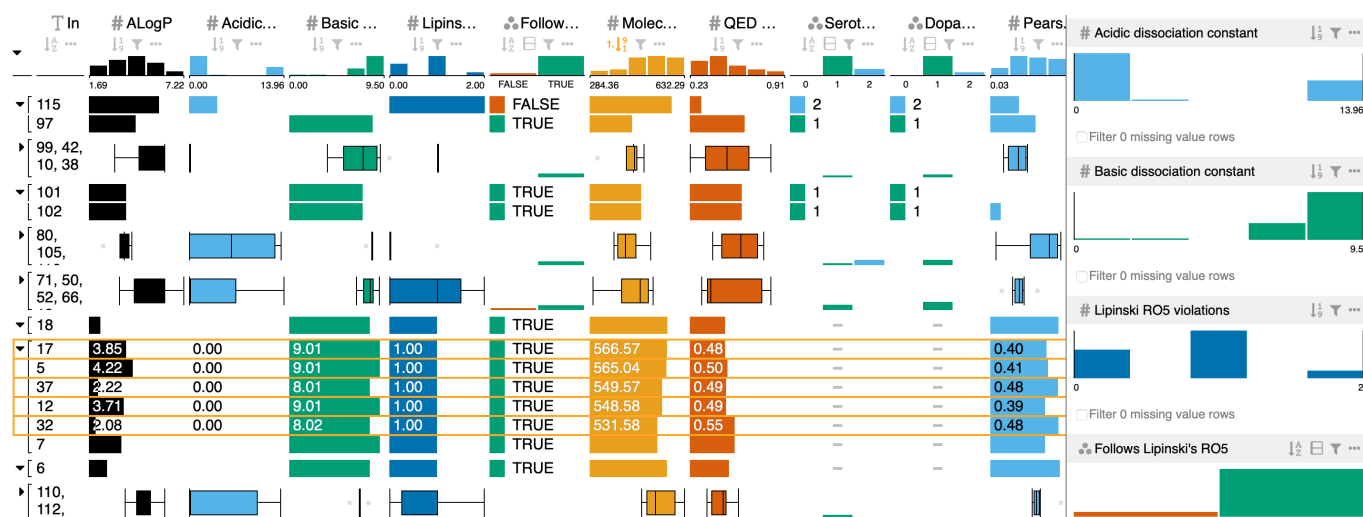


Figura 7.6: Vista tabular: cada fila corresponde a un compuesto y sus características seleccionadas. El panel lateral derecho muestra una descripción estadística de la distribución de valores en todo el conjunto de datos.

Su desarrollo original, detalles sobre su implementación y amplia gama de interacciones pueden encontrarse en sus respectivos artículos originales.

Además de listar los compuestos químicos presentes en el conjunto de datos en estudio, la vista Tabular proporciona a los usuarios pequeñas visualizaciones accesorias en forma de gráficos de barras y diagramas de caja y bigotes yuxtapuestos en su panel lateral derecho. Por defecto, los compuestos mostrados en la tabla se agrupan lógicamente según su pertenencia a celdas hexagonales en la vista Hexagonal (figura 7.1 A). Estos grupos se pueden expandir o colapsar, a fin de mostrar u ocultar los compuestos comprendidos. Cuando están colapsados, la vista muestra los diagramas de caja y bigotes de la distribución de los valores de cada característica en dicha celda hexagonal, como se muestra en la figura 7.6.

La vista Tabular es interactiva y está coordinada con los demás componentes visuales de *ChemVA*. Cuando se realiza una selección en las vistas 2D, los compuestos correspondientes son automáticamente destacados en la vista Tabular. Por otro lado, cuando el usuario selecciona compuestos en la vista Tabular, dichos compuestos son resaltados en la vista Detallada y mostrados en la vista 3D, además de resaltarse las celdas hexagonales que los contienen en la vista Hexagonal.

### 7.5.4. Vista 3D

La estructura geométrica de los compuestos seleccionados puede ser explorada mediante la vista 3D de estructuras moleculares integrada en *ChemVA*. La representación visual de átomos y enlaces se basa en representaciones estándar utilizadas en el dominio de la química molecular, y su mapeo de color fue realizado con colores específicos reservados para cada elemento químico. La vista 3D también ofrece un conjunto de interacciones básicas, tales como desplazamiento, zoom y rotación de las estructuras mostradas.

La vista 3D fue concebida para proporcionar un medio interactivo de visualización de la similitud estructural entre compuestos de interés por medio de la yuxtaposición de sus estructuras moleculares. La similitud entre compuestos puede ser mejor percibida cuando los compuestos están alineados estructuralmente en la vista. Para este propósito, utilizamos una funcionalidad de alineación estructural proporcionada por la herramienta OpenBabel [283]. Obtuvimos y calculamos las conformaciones 3D de los compuestos por medio de PubChem [196] y OpenBabel [283]. Las

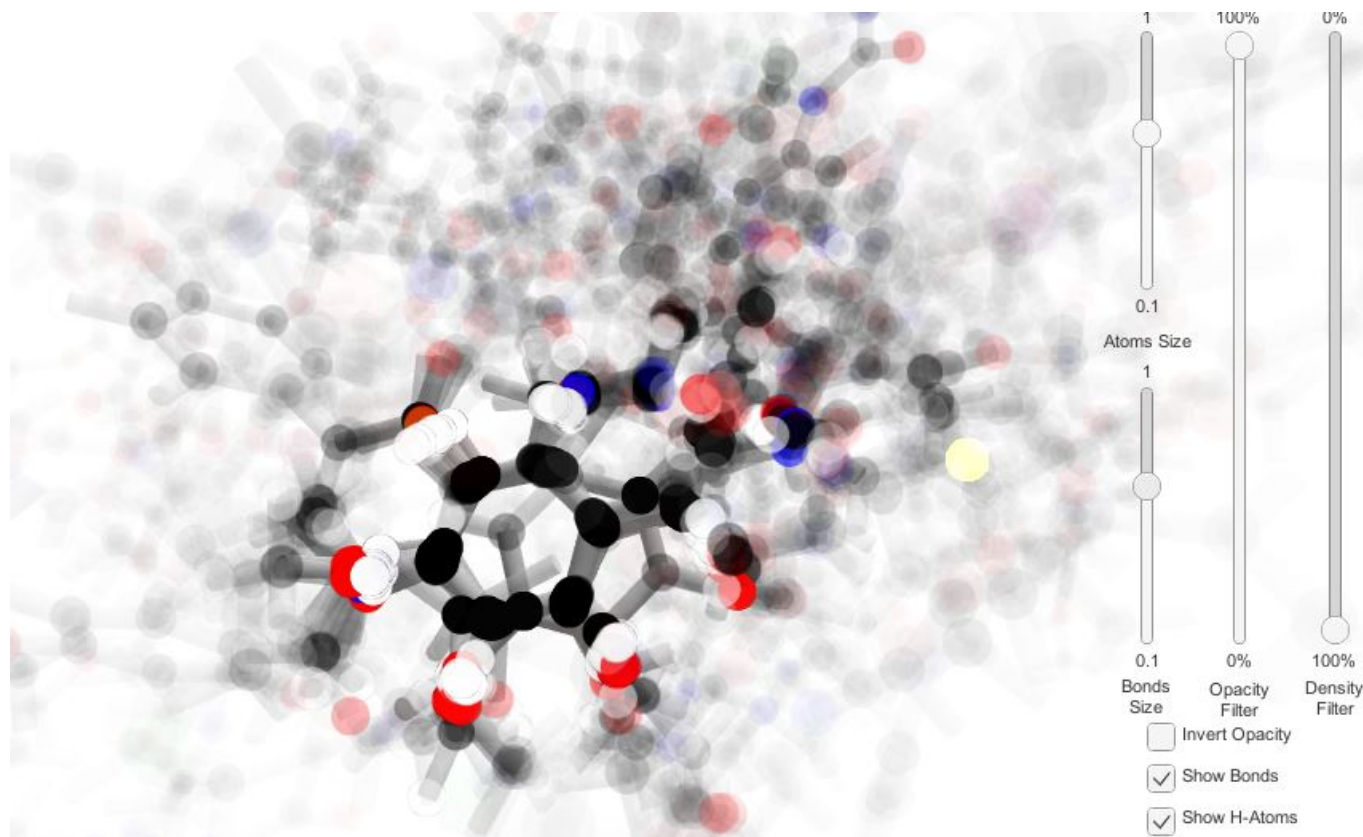


Figura 7.7: Vista 3D con 66 moléculas alineadas. Los átomos y enlaces comunes se representan con mayor opacidad que los componentes menos comunes.

conformaciones 3D calculadas se obtuvieron mediante minimización de la energía del campo de fuerza *Ghemical*, limpieza del campo de fuerza (500 ciclos) y búsqueda lenta de rotores. El cálculo de estas conformaciones 3D consistió en una tarea realizada de forma previa al lanzamiento de la herramienta, de forma *offline* y demoró aproximadamente entre 2 y 20 minutos por cada compuesto, dependiendo de la complejidad de su estructura molecular, en un clúster de 32 núcleos con 12 GB de memoria.

La función de alineación molecular involucró dos pasos. El primer paso consistió en encontrar la *subestructura común máxima* (MCS, por sus siglas en inglés) entre los compuestos seleccionados, paso desarrollado por medio de herramientas provistas por la librería RDKit [132]. El segundo paso consistió en alinear todas las moléculas seleccionadas con respecto a la MCS encontrada, y se realizó utilizando la función *obfit* proporcionada por OpenBabel [283]. La tarea de alineación molecular es realizada *a demanda*, por lo que se ejecuta en tiempo real al seleccionar un conjunto de compuestos en la vista Detallada. El tiempo de respuesta de este servicio web es de aproximadamente un segundo, probado en selecciones aleatorias de 5 a 30 compuestos.

Una vez que las moléculas están alineadas en su MCS, *lx* *usuariX* es capaz de identificar fácilmente sus partes comunes, es decir, los subconjuntos de átomos y enlaces que están presentes en la mayoría de los compuestos seleccionados, según establecimos en el requisito R5.

Además, incorporamos una modulación de opacidad con respecto a la frecuencia de ocurrencia de átomos y enlaces. En otras palabras, la opacidad de átomos y enlaces en la vista 3D se calcula en función del número de átomos del mismo tipo que están alineados en la misma posición espacial 3D. Como consecuencia, las subestructuras comunes se representan con mayor opacidad. La figura 7.7 muestra un ejemplo de alineación estructural entre 66 compuestos químicos, donde se puede apreciar la influencia de la opacidad de átomos y enlaces químicos en la visualización.

Dependiendo del número de compuestos seleccionados y sus similitudes, la representación visual 3D puede volverse difícil de estudiar en detalle por la presencia de numerosas subestructuras no alineadas (figura 7.7). Para mitigar este problema, la vista 3D ofrece la opción de ocultar y mostrar átomos e enlaces de hidrógeno a pedido, cambiar el tamaño de átomos y enlaces y también cambiar la opacidad de toda la estructura. La figura 7.8 muestra un ejemplo de uso de estas funcionalidades.

En algunos casos, *lx* *usuariX* desea enfocar su atención únicamente en las subestructuras comunes del conjunto de compuestos. Por lo tanto, también hemos incluido una funcionalidad para filtrar átomos y enlaces. La figura 7.8(a) muestra cómo se mejora la visibilidad de la subestructura común entre 66 compuestos mediante el uso de esta característica. Finalmente, la vista 3D proporciona una opción para invertir la opacidad, de modo que la parte común de la estructura se vea más

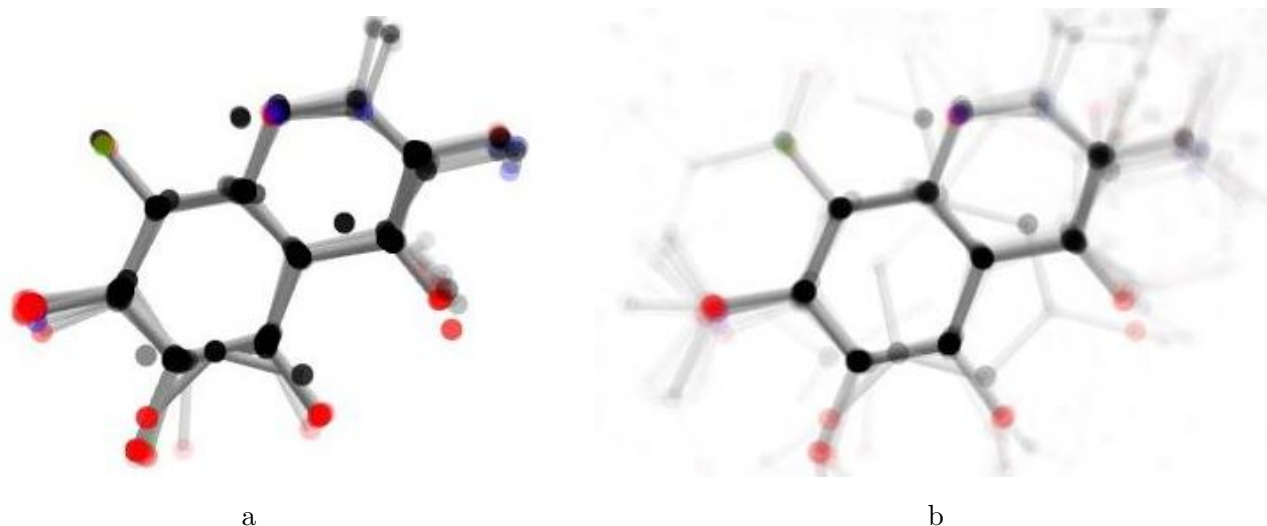


Figura 7.8: Reducción del ruido visual en la figura 7.7 mediante (a) filtrado de subestructuras no comunes del conjunto de compuestos seleccionados y (b) modificación de la opacidad de toda la estructura.

translúcida que el resto, de forma tal de permitir a lx usuarix analizar las subestructuras no comunes de los compuestos.

La vista 3D fue implementada utilizando el motor Unity [369] y desplegada en WebGL [195] para su integración en *ChemVA*.

### 7.5.5. Reducción dimensional paramétrica

La técnica *t-Distributed Stochastic Neighborhood Embedding* (t-SNE) [393] se ha convertido en el método estándar de reducción dimensional en contextos donde es necesario visualizar datos de alta dimensionalidad. Sin embargo, exhibe limitaciones por cuanto se trata de una técnica no paramétrica y, por ende, no brinda flexibilidad a la hora de incorporar datos nuevos a proyecciones previamente calculadas. En el contexto del desarrollo de *ChemVA*, probamos su variante paramétrica, denominada *Parametric t-SNE* [392], con el fin de aprovechar las características deseables de t-SNE a la vez que se permite incorporar nuevos compuestos a las proyecciones 2D una vez entrenado el modelo de reducción dimensional. Sin embargo, después de numerosos ensayos y pruebas de diferentes parametrizaciones de la técnica, no obtuvimos proyecciones que no estuvieran afectadas por el *problema de aglomeración* (*crowding problem*), situación previamente reportada en otros trabajos de investigación [120, 450, 437].

Considerando las bondades de t-SNE, desarrollamos nuestro propio modelo paramétrico de reducción dimensional basado en una red neuronal multicapa superficial de propagación hacia adelante, entrenada a partir de las coordenadas bidimensionales de una proyección previamente calculada de t-SNE de los datos. Para ello, calculamos una proyección t-SNE para cada una de las representaciones moleculares basadas en vectores, para ambos conjuntos de datos utilizados en los casos de estudio. Las proyecciones t-SNE se calcularon utilizando la clase *manifold* de Scikit-learn [293], y los parámetros utilizados fueron seleccionados mediante búsqueda en cuadrícula. Los valores de perplejidad probados variaron desde el 5 hasta el 10 % del total de compuestos en el conjunto de datos. La obtención de cada proyección t-SNE fue realizada en un clúster de 32 núcleos con 12 GB de memoria. Proporcionamos un resumen de la parametrización utilizada para t-SNE en la tabla 7.1.

Conjunto de datos	P-glicoproteína	Serotonina-Dopamina
Perplejidad	45	5~10
No. <i>epochs</i> máximo	10.000	10.000
Paciencia para <i>early stopping</i>	500~1.000	1.000

Tabla 7.1: Resumen de los parámetros empleados en el cómputo de las proyecciones t-SNE para cada conjunto de datos.

Después de entrenar cada proyección t-SNE, entrenamos un modelo paramétrico para aprender las coordenadas bidimensionales dadas por cada una de estas proyecciones. Los modelos paramétricos se construyeron utilizando las librerías Keras y Tensorflow [71]. La parametrización de estos modelos se resume en las tablas 7.2 y 7.3. El entrenamiento de estos modelos paramétricos de reducción dimensional fue también realizado en un clúster de 32 núcleos con 12 GB de memoria. Tanto el ajuste de las proyecciones t-SNE como el entrenamiento de los modelos paramétricos fueron realizados *offline*, siendo los modelos paramétricos preentrenados luego utilizados interactivamente para la obtención de coordenadas 2D ante la eventual carga de compuestos nuevos en la plataforma. En las tablas 7.2 y 7.3 proporcionamos un resumen de los parámetros empleados para entrenar los modelos paramétricos para las cuatro representaciones moleculares vectoriales adoptadas en *ChemVA*.

### 7.5.6. Cómputo de puntajes de correlación

Tal y como describimos anteriormente, los puntajes de correlación son la estrategia que elegimos como estimador cuantitativo de la confianza en las proyecciones 2D de los compuestos químicos empleando diversas representaciones moleculares basadas en vectores. Estos puntajes no solo pueden

Hiperparámetros	ECFPs	Daylight fps	Descriptores moleculares	Embeddings moleculares
# Nodos por capa oculta	50 / 10 / 2	50 / 10 / 2	200 / 50 / 2	50 / 10 / 2
Función de activación	relu	relu	tanh	sigmoid
Tasa de <i>dropout</i>	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1
Paciencia para <i>early stopping</i>	70	70	100	70
$\delta$ <i>early stopping</i>	0,005	0,005	0,005	0,005
<i>Learning rate</i> (optimización Adam)	0,0001	0,0001	0,0001	0,0001

Tabla 7.2: Hiperparámetros utilizados para construir y entrenar los cuatro modelos paramétricos para el conjunto de datos asociados a los receptores *serotonina-dopamina*.

Hiperparámetros	ECFPs	Daylight fps	Descriptores moleculares	Embeddings moleculares
# Nodos por capa oculta	100 / 10 / 2	100 / 10 / 2	200 / 20 / 2	100 / 10 / 2
Función de activación	relu	relu	relu	relu
Tasa de <i>dropout</i>	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1	0,25 / 0,15 / 0,1
Paciencia para <i>early stopping</i>	70	130	120	50
$\delta$ <i>early stopping</i>	0,005	0,005	0,005	0,005
<i>Learning rate</i> (optimización Adam)	0,0001	0,0001	0,0001	0,0001

Tabla 7.3: Hiperparámetros utilizados para construir y entrenar los cuatro modelos paramétricos para el conjunto de datos del caso de estudio asociado al receptor *P-glicoproteína*.

visualizarse como codificación de color en las vistas detallada y hexagonal, sino que además son empleados en la vista de Contraste, siendo codificados en la opacidad de las celdas hexagonales inscriptas (es decir, en la proyección de vista comparada sobre la vista de referencia).

Calculamos dos puntajes de correlación diferentes. El primer puntaje, denotado como  $r$ , se calculó comparando las distancias por coseno entre cada compuesto  $k$  y el resto de sus pares en el espacio de alta dimensionalidad. Luego, medimos la correlación de Pearson (Ecuación 7.1) entre estas distancias en el espacio de alta dimensión y las distancias en el espacio de baja dimensión obtenido por medio de t-SNE. La correlación de Pearson es una medida estadística ampliamente utilizada para evaluar la relación lineal entre dos conjuntos de datos [339].

El segundo puntaje, denotado como  $\tau$ , se calculó generando rangos para cada compuesto  $k$ , en los que se estableció un ordenamiento total entre todos los compuestos del conjunto de datos con respecto a  $k$  de acuerdo a su distancia coseno al compuesto  $k$ . Se generaron dos rangos: uno en el espacio original de alta dimensionalidad, tomado como referencia, y otro en el espacio t-SNE de baja dimensionalidad. Luego, se procedió a comparar cuántos compuestos en el rango de baja dimensionalidad se ubicaban en la misma posición relativa de cercanía a  $k$  ( $n_{concordants}$ ) y cuántos se ubicaban en posiciones relativas diferentes ( $n_{discordants}$ ) con respecto a sus posiciones en el rango de referencia, correspondiente al espacio de alta dimensionalidad. El puntaje  $\tau$  fue luego calculado



comparando los dos rangos utilizando la correlación de rangos de Kendall (Ecuación 7.2), que mide la similitud entre los ordenamientos de dos conjuntos de datos [217].

Ambos puntajes de correlación se calcularon individualmente para cada compuesto y para cada una de las cuatro representaciones moleculares basadas en vectores utilizadas en *ChemVA*, lo que dio como resultado cuatro vistas diferentes para evaluar la confiabilidad de cada proyección. Para computar ambos puntajes empleamos herramientas provistas por las librerías SciPy [401] y Scikit-learn [293], que proporcionan implementaciones eficientes y confiables de las medidas estadísticas requeridas.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \quad (7.1)$$

$$\tau = \frac{n_{concordants} - n_{discordants}}{n(n-1)/2} \quad (7.2)$$

Los puntajes de correlación proporcionan un medio para que lxs expertxs en el dominio evalúen la confiabilidad de las proyecciones t-SNE. Los puntajes de correlación en diferentes representaciones moleculares pueden ser comparados para elegir la representación más adecuada para su tarea específica de cribado virtual. La vista de Contraste utiliza los puntajes de correlación de rangos de Kendall  $\tau$  como codificación predeterminada para permitir a lxs usuarixs comparar visualmente la similitud de los compuestos en el espacio de alta dimensión original con su disposición en la proyección de t-SNE. Esto ayuda a detectar posibles distorsiones y comprender las limitaciones del método de proyección.

## 7.6. Soporte para diseño *de novo*

Uno de los pasos clave durante la selección y validación de nuevos compuestos candidatos es contrastarlos con compuestos ya conocidos y estudiados. En este sentido, *ChemVA* también ofrece la opción de agregar nuevos compuestos al conjunto de datos en estudio con el fin de explorar sus características y compararlos con otros compuestos. Mediante esta funcionalidad, lxs expertxs puede evaluar el potencial de estos nuevos compuestos antes de realizar pruebas posteriores en laboratorio. El soporte para esta característica importante cumple con el requisito R6 previamente establecido para la herramienta.

Un compuesto nuevo se carga en *ChemVA* especificando su fórmula SMILES [419] en un formulario web integrado a la herramienta. Luego, un servicio en línea que corre en el servidor de *ChemVA* calcula las características moleculares y representaciones vectoriales listadas en la sección 7.3.2. Después de calcular estas representaciones y características moleculares, *ChemVA* utiliza los modelos paramétricos t-SNE (descritos en la sección 7.5.5) para obtener las coordenadas en 2D del compuesto nuevo en cada proyección 2D.

Finalmente, el compuesto recién agregado se muestra en todas las vistas admitidas en color amarillo para facilitar su identificación. Esta codificación de color prevalece en todas las vistas, como se puede ver en la figura 7.9. La adición de nuevos compuestos a la vista Detallada es una tarea que se realiza en línea y a demanda. Los modelos paramétricos son cargados previamente, lo que reduce significativamente el tiempo de respuesta (aproximadamente cuatro segundos por compuesto nuevo).

## 7.7. Evaluación de *ChemVA* y casos de estudio

La evaluación de *ChemVA* constó de dos etapas principales. La primera etapa fue realizada por unx expertx en el dominio que participó en el diseño funcional de la herramienta (ver sección 7.4), y consistió en dos casos de estudio. La segunda etapa fue llevada a cabo por unx expertx en visualización y dos expertxs en el dominio que no estuvieron involucrados en ningún momento en el diseño o desarrollo de *ChemVA*. Esta etapa consistió en una sesión donde las diferentes vistas fueron evaluadas cualitativamente y luego se recopilaron comentarios generales sobre la usabilidad y funcionalidad de la herramienta.

### 7.7.1. Primera etapa: Casos de estudio

Durante la primera etapa de evaluación de *ChemVA*, nuestrx expertx en el dominio, quien fue también co-autorx del trabajo de investigación [323], brindó comentarios valiosos e identificó varios requisitos funcionales adicionales para la herramienta en sí. Estos requisitos, como cargar y almacenar selecciones, descargar conformaciones 3D y mejorar la interacción entre diferentes vistas, se abordaron antes de la evaluación de los casos de estudio, y se resumen en la sección 7.4 del presente capítulo.

El rendimiento de *ChemVA* se evaluó mediante dos casos de estudio utilizando conjuntos de datos obtenidos de ChEMBL [116]. El primer conjunto de datos empleado combina ligandos dirigidos al *receptor de serotonina 1a* y al *receptor de dopamina D2*. El segundo conjunto de datos consiste en ligandos para la *P-glicoproteína 1*. Cada compuesto se categorizó según su valor de actividad biológica

*IC50* medido experimentalmente hacia el objetivo respectivo. Siguiendo los estándares internacionales y el estado del arte, lx expertx etiquetó aquellos compuestos con valores de *IC50* inferiores a 10 *nM* como *Activos*, aquellos entre 10 y 1.000 *nM* como *Moderadamente Activos*, y aquellos superiores a 1.000 *nM* como *Inactivos*. El conjunto de datos *serotonina-dopamina* contiene 118 compuestos, mientras que el conjunto de datos de *P-glicoproteína* consta de 893 compuestos. Detalles sobre la composición de ambos conjuntos de datos se presentan en las tablas 7.4 y 7.5

	Clase	Serotonina 1a	Dopamina D2
	Activos	14	5
# compuestos	Moderadamente activos	42	28
por clase	Inactivos	62	85
	<b>Total</b>	<b>118</b>	<b>118</b>

Tabla 7.4: Composición por perfil de bioactividad del conjunto de datos de ligandos contra *serotonina-dopamina*.

	Clase	P-glicoproteína
	Activos	42
# compuestos	Moderadamente activos	178
por clase	Inactivos	673
	<b>Total</b>	<b>893</b>

Tabla 7.5: Composición por perfil de bioactividad del conjunto de datos de ligandos contra *P-glicoproteína*

#### 7.7.1.1. Caso de estudio 1: Análisis de determinantes químicos de la actividad hacia receptores de serotonina y dopamina

Este caso de estudio se basa en el conjunto de datos *serotonina-dopamina*, un conjunto de compuestos químicos que contiene información sobre la actividad antagonista contra los receptores de serotonina 5HT1A y dopamina D2. Estos receptores de neurotransmisores son dianas biológicas de muchos medicamentos psicoactivos, tales como antidepresivos, antipsicóticos y ansiolíticos. En el campo de la psicofarmacología, se utilizan pequeñas moléculas con diferentes perfiles de bioactividad hacia varios de estos receptores para aliviar los efectos secundarios [277, 351]. Los medicamentos psicoactivos a menudo exhiben actividad hacia múltiples receptores, lo que puede causar efectos

secundarios no deseados, por lo que resulta de vital importancia realizar estudios preliminares del perfil de bioactividad de los compuestos candidatos teniendo en cuenta estas posibles interacciones.

El objetivo de este caso de estudio fue encontrar los determinantes químicos de la actividad biológica hacia los receptores de serotonina y dopamina. Según lx integrante de nuestro equipo de desarrollo experto en el dominio que llevó a cabo el caso de estudio, múltiples grupos de compuestos similares pudieron encontrarse fácilmente en la vista Detallada en las cuatro proyecciones, teniendo en consideración su proximidad en el espacio 2D. Después de identificar grupos de compuestos potencialmente similares, lx expertx buscó grupos de compuestos que fueran *activos antagonistas* hacia ambos receptores y que tuvieran propiedades farmacológicas deseables, tales como seguir la Regla de los Cinco de Lipinski y tener un alto Puntaje QED, ambos indicadores de perfiles farmacológicos deseables. Esta búsqueda exploratoria se realizó por medio de la vista Hexagonal, ajustando el tamaño de las celdas hexagonales para que coincidieran con los grupos de compuestos observados en las diferentes proyecciones.

Después, lx expertx utilizó la codificación de colores de correlación de Kendall y Pearson para observar si las proyecciones eran confiables, y así determinar si las celdas hexagonales consideradas agruparían efectivamente compuestos similares. Un conjunto de celdas fue preseleccionado y se utilizó la vista Tabular, en la que lx expertx analizó las distribuciones de todas las características moleculares asociadas a perfiles farmacológicos deseables de los compuestos seleccionados utilizando la información resumida mostrada en los encabezados de las columnas. El objetivo de este procedimiento era encontrar compuestos similares, tanto en términos de su estructura como de su bioactividad, para observar sus determinantes químicos de actividad. Como resultado, lx expertx identificó una celda hexagonal que agrupaba compuestos activos hacia ambos receptores con características farmacológicas deseables.

Posteriormente, lx expertx seleccionó esos compuestos y realizó una alineación en la vista 3D, encontrando que sus estructuras eran muy similares. También contrastó estos compuestos con compuestos inactivos dentro del mismo hexágono usando la vista 3D. Esto le permitió obtener una idea de las subestructuras relevantes de un posible nuevo candidato a medicamento. En este punto, lx expertx destacó las ventajas de esta vista 3D en lo que respecta a la facilidad para encontrar y visualizar la estructura tridimensional común de un grupo de compuestos.

Finalmente, las estructuras de tres compuestos activos hacia ambos receptores fueron descargadas de *ChemVA* en formato *Structure-Data Format* (SDF, por sus siglas en inglés), que permite representar estructuras químicas utilizando un bloque de texto que enumera los átomos, enlaces, conectividad y coordenadas del compuesto químico. Lx expertx creó cinco nuevos compuestos con

herramientas de diseño de fármacos, los cuales fueron luego cargados en *ChemVA* por medio de la funcionalidad provista para agregar nuevos compuestos, como se muestra en la figura 7.9.

Las estructuras moleculares de los compuestos creados y descargados fueron validadas por medio de la técnica de acoplamiento molecular (*molecular docking*), brevemente introducida en el capítulo 3. Para este procedimiento, en primera instancia se construyeron las estructuras de los receptores serotonina 5HT1A y dopamina D2, utilizando el servidor web *I-TASSER* con configuraciones predeterminadas [438] y descargando información estructural de PDB [43] (ID de PDB 6CM4 [409]). Hidrógenos, iones y solventes fueron eliminados utilizando PyMOL [334]. Asimismo, basándose en las tres estructuras moleculares seleccionadas en *ChemVA*, se diseñaron cinco nuevas estructuras de ligandos utilizando Avogadro [147]. El acoplamiento molecular tenía como objetivo comparar las energías de unión y los modos de unión de las tres estructuras del conjunto de datos con las de las cinco estructuras diseñadas. El acoplamiento molecular se realizó utilizando Autodock Vina [387]. Se añadieron tipos de átomos de Autodock y cargas de Gasteiger a los dos receptores y los ligandos utilizando MGLTools [269, 328].

Las ocho estructuras se unieron con éxito a los sitios de unión de los dos receptores. Tres de las cinco estructuras diseñadas mostraron un valor más bajo de energía de unión que aquellos exhibidos por las tres estructuras descargadas de *ChemVA*, que ya eran altamente activas, lo que indica una unión más fuerte. Las energías de unión de los mejores compuestos se muestran en la tabla 7.6, y las fórmulas SMILES de cada uno de los compuestos diseñados se proporcionan en la tabla 7.7. Estos hallazgos demuestran que *ChemVA* podría utilizarse eficazmente como asistencia al proceso de diseño *de novo* de fármacos.

En esta comparación, los compuestos diseñados **2**, **4** y **5** exhibieron afinidades predichas más altas para ambos receptores. Estos resultados predichos no necesariamente significan una mayor actividad inhibitoria, dado que dicha bioactividad puede verse influenciada por otros factores como la flexibilidad de las proteínas receptoras.

#### **7.7.1.2. Caso de estudio 2: Análisis de determinantes estructurales de inhibidores de la P-glicoproteína**

Este caso de estudio se basó en el conjunto de datos *P-glicoproteína*, que consta de pequeñas moléculas con actividad inhibitoria contra la P-glicoproteína humana. Esta proteína se sobre-expresa exclusivamente en células de muchos tipos de cáncer, causando resistencia a múltiples fármacos en células cancerígenas y, por lo tanto, afectando el rendimiento de los tratamientos de quimioterapia.

Compuesto	Afinidad a 5HT1A	Afinidad a dopD2
Compuesto descargado 45	-8,0	-8,7
Compuesto descargado 79	-8,0	-8,3
Compuesto descargado 117	-9,9	-11,0
Compuesto diseñado 1	-8,4	-11,5
<b>Compuesto diseñado 2</b>	<b>-10,5</b>	<b>-11,9</b>
Compuesto diseñado 3	-9,9	-10,8
<b>Compuesto diseñado 4</b>	<b>-10,3</b>	<b>-11,2</b>
<b>Compuesto diseñado 5</b>	<b>-10,2</b>	<b>-11,2</b>

Tabla 7.6: Afinidades de unión predichas hacia el receptor serotoninérgico **5HT1A** y el receptor de dopamina **dopD2**, tanto para los compuestos seleccionados del conjunto de datos como para los compuestos diseñados, medidas en *kcal/mol*. Las filas en **negrita** muestran los compuestos que exhiben las afinidades predichas más altas para ambos receptores.

Compuesto Diseñado	Fórmula SMILES
1	<chem>Fc1ccc(cc1)c2cncc(CNCC3CCc4ccccc4C3)c2</chem>
<b>2</b>	<b><chem>Fc1ccc(cc1)c2cncc(CNCC3CCc4cc(C)c(C)cc4C3)c2</chem></b>
3	<chem>N(Cc1cncc(c1)c2cc(F)ccc2F)CC3CCC4=C(C=CC=C4)O3</chem>
<b>4</b>	<b><chem>N(Cc1cncc(c1)c2ccc(F)cc2)C(N(C)C)C3CCC4=C(C=CC=C4)O3</chem></b>
<b>5</b>	<b><chem>Fc1ccc(cc1)c2cncc(c2)C(N)CCC3CCc4ccccc4O3</chem></b>

Tabla 7.7: Fórmulas SMILES de los compuestos diseñados. Las filas en **negrita** muestran los compuestos que exhiben las afinidades de unión predichas más altas para ambos receptores.

También afecta la efectividad de muchos fármacos al alterar sus propiedades farmacocinéticas de absorción, distribución, metabolismo, excreción y toxicidad (*ADME-Tox*) [356], sintéticamente presentadas en el capítulo 3 de la presente tesis. Se han realizado múltiples esfuerzos de investigación para descubrir nuevos inhibidores de la P-glicoproteína que permitan desarrollar estrategias de quimioterapia más efectivas. La P-glicoproteína ha demostrado tener la capacidad de unirse a muchos sustratos estructuralmente diferentes, por lo que un caso de estudio interesante para nuestra herramienta fue encontrar determinantes estructurales de un buen inhibidor de la P-glicoproteína y comparar ligandos activos conocidos para este objetivo.

Según *lx expertx* en el dominio a cargo del caso de estudio, el conjunto de datos de la P-glicoproteína contiene una gran cantidad de compuestos diversos que interactúan con el blanco terapéutico. El objetivo de este estudio fue encontrar determinantes químicos de los compuestos

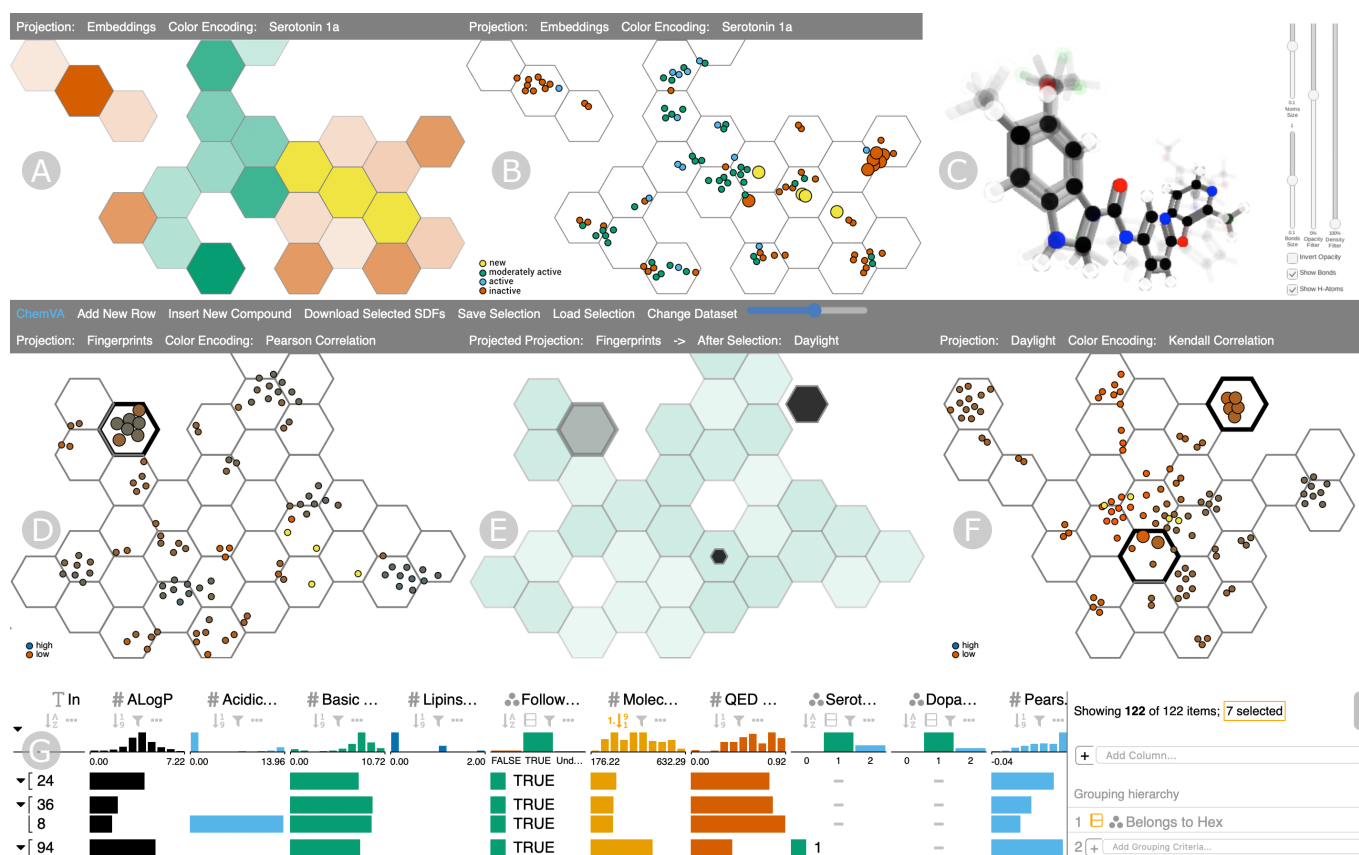


Figura 7.9: Los compuestos diseñados fueron incorporados a las vistas y sus proyecciones efectivamente se localizan cercanas a las de los compuestos seleccionados del conjunto de datos en la proyección basada en *fingerprints Daylight* (arriba), ocupando los mismos hexágonos, y ligeramente más alejados de ellos en la proyección *fingerprints ECFP* (abajo). La vista de Contraste en la esquina inferior derecha ilustra la redistribución de las regiones seleccionadas. a) vista hexagonal, b-d-f) vista detallada, c) vista 3D, e) vista de contraste, g) vista tabular.

con un valor de  $\log P$  muy alto, que son escasos en el conjunto de datos. Para lograr este objetivo, lx expertx utilizó la vista Tabular para filtrar aquellos compuestos con valores de  $\log P$  superiores a 6,75, y luego los ordenó en función de dicho valor y de sus puntajes de correlación, que miden la confiabilidad de la representación molecular que se proyecta en las vistas 2D.

Utilizando la vista Tabular de *ChemVA* y sus opciones de filtrado, y según indicó lx expertx en el dominio, resultó sencillo encontrar un conjunto de compuestos que cumplan con todos los criterios solicitados, es decir, un valor alto de  $\log P$ , actividad inhibitoria hacia la P-glicoproteína y el cumplimiento de las Reglas de Lipinski. Posteriormente, se seleccionaron compuestos químicamente similares de este subconjunto y se analizaron en profundidad utilizando la vista Hexagonal. Se

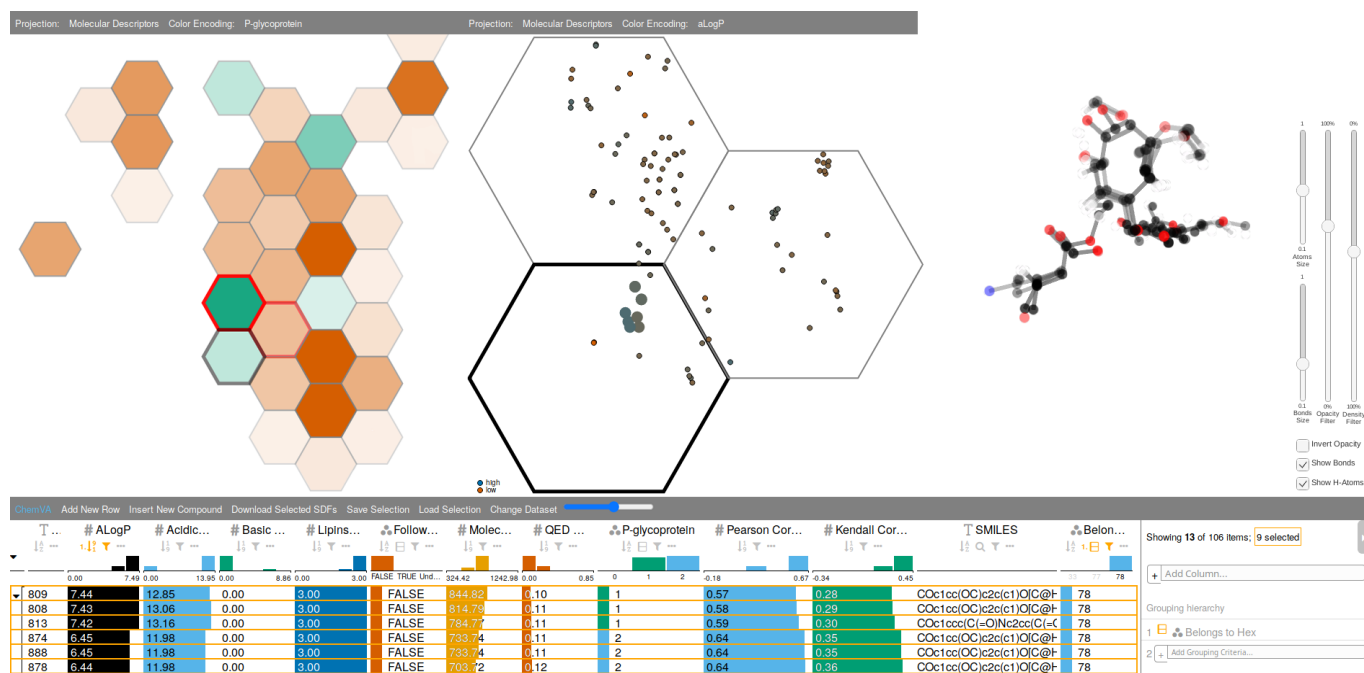


Figura 7.10: Ilustración del proceso exploratorio en el conjunto de datos de inhibidores de la P-glicoproteína, mostrando solo un subconjunto de moléculas de interés en la vista Detallada y seleccionando aquellas con valores altos de  $\log P$ .

seleccionaron celdas hexagonales que contuvieran compuestos activos, las cuales se exploraron utilizando la vista Detallada, en la cual se utilizó el valor de  $\log P$  como codificación de colores. Finalmente, lx expertx en el dominio utilizó la vista en 3D para encontrar subestructuras comunes entre los compuestos seleccionados, lo que le permitió identificar con éxito determinantes químicos estructurales del perfil de bioactividad del conjunto de compuestos seleccionado. Este flujo de trabajo se ilustra en la figura 7.10.

### 7.7.2. Segunda etapa: Evaluación cualitativa de la herramienta por expertxs externxs

En la segunda etapa de evaluación, presentamos *ChemVA* a dos expertxs en el dominio de los *Laboratorios Loschmidt* de la Universidad Masaryk, quienes evaluaron nuestra herramienta en términos de funcionalidad y experiencia de usuario. También presentamos nuestra nueva vista de Contraste a unx expertx en visualización especialista en técnicas de reducción de dimensionalidad. Ningunx de lxs expertxs que participaron de la segunda etapa de evaluación formó parte del proceso de diseño y desarrollo de *ChemVA*.



En primer lugar, realizamos una breve introducción a la herramienta, sus vistas y las estrategias de visualización propuestas. Esta etapa duró aproximadamente treinta minutos con cada unx de ellxs, después de lo cual pudieron utilizar la herramienta sin necesidad de asistencia adicional por parte de nuestro equipo. Si bien ambxs expertxs destacaron que las vistas hexagonal y 3D resultaron intuitivas y fáciles de utilizar, también señalaron la necesidad de una documentación adecuada que describa otras vistas, como nuestra nueva vista de contraste, para respaldar su usabilidad.

Después de la introducción a la herramienta, lxs expertxs nos presentaron casos de estudio de su propia investigación en los que creen que *ChemVA* podría ser una herramienta valiosa. Destacaron la funcionalidad de las vistas tabular y la posibilidad de integrar múltiples vistas 2D, lo que permite comparar fácilmente numerosas propiedades, lo cual es una característica poco común pero muy útil en las herramientas para el cribado virtual. Ambxs destacaron la utilidad de tener diferentes proyecciones y la capacidad de enfocarse en una fuente específica de datos utilizando los puntajes de correlación y comparándolos mediante la vista de Contraste.

Además, lxs expertxs identificaron un uso potencial de *ChemVA* en un trabajo en desarrollo en su laboratorio, que consistía en el análisis de alrededor de 4.300 medicamentos aprobados por la Agencia de Drogas y Medicamentos de Estados Unidos (*FDA*) para el tratamiento de COVID-19. Al momento de la evaluación, lxs expertxs analizaban este conjunto de datos mediante programas desarrollados en Python para filtrar y agrupar los datos en función de un número limitado de propiedades manualmente seleccionadas y, en este sentido, resaltaron que *ChemVA* les permitiría explorar más características moleculares en simultáneo y sus relaciones, así como también elaborar nuevas hipótesis de investigación en torno al caso de estudio.

Según lx expertx en visualización, la vista de Contraste cumple su función al permitir a lx usarix visualizar cómo se correlacionan las dos representaciones moleculares basadas en vectores, es decir, los dos espacios de características, en términos de agrupar el mismo conjunto de compuestos químicos. Mencionó que la vista de Contraste podría ser útil para encontrar determinantes estructurales de similitud al observar los cambios en los vecindarios o agrupamientos de compuestos al seleccionar diferentes proyecciones. También podría facilitar la identificación de moléculas individuales que se desplazan entre vecindarios de una proyección a otra, lo que ayudaría a comprender qué características moleculares son más importantes para un perfil de bioactividad específico.

## 7.8. Resultados y discusión

En esta sección del capítulo resumimos algunos comentarios y observaciones finales proporcionados por lxs expertxs que llevaron a cabo ambas etapas de la evaluación de *ChemVA*. Discutimos los resultados de ambos casos de estudio, así como las fortalezas y limitaciones de nuestra herramienta y potenciales expansiones de la misma.

Como se muestra en la sección 7.7.1, los objetivos de ambos casos de estudio se cumplieron de manera efectiva utilizando *ChemVA*, lo que sugiere que la herramienta tiene el potencial de ser ampliamente adoptada y utilizada por expertxs en química medicinal. Esta afirmación también es respaldada por los resultados de la evaluación cualitativa realizada por expertxs externxs, descrita en la sección 7.7.2. Nuestra herramienta permitió una comparación intuitiva de conjuntos de compuestos químicamente similares. A diferencia de otras herramientas, que solamente permiten ordenar y filtrar compuestos por un conjunto de propiedades acotado, *ChemVA* permitió un análisis integral de los datos al tener en cuenta múltiples representaciones moleculares y niveles de granularidad de los datos químicos, a través de funcionalidades como la agrupación de compuestos en celdas hexagonales y la posibilidad de ajustar su tamaño.

En el primer caso de estudio, la vista Detallada ayudó a lx expertx a distinguir compuestos similares basados en su proximidad en las cuatro proyecciones (R1 y R2). Además, los mapeos de color y formas presentados en las vistas Hexagonal y detallada permitieron a lx expertx encontrar rápidamente una celda hexagonal específica (R1).

La tarea de analizar la similitud de compuestos pudo ser probada exhaustivamente mediante diferentes proyecciones proporcionadas por *ChemVA* (R2). En ambos casos de estudio fue posible examinar estas proyecciones en términos de su confiabilidad, utilizando los puntajes de correlación proporcionados (R3), y compararlos mediante la vista de Contraste (R2). Estas dos características proporcionadas por *ChemVA* ayudaron a lx especialista en el primer caso de estudio para evaluar si las proyecciones eran confiables y, por lo tanto, si las celdas hexagonales agrupaban efectivamente compuestos similares.

La evaluación también mostró la utilidad de la vista 3D para descubrir y analizar estructuras 3D similares y subestructuras comunes entre los compuestos seleccionados (R5). Examinar los determinantes químicos de la bioactividad en ambos casos de estudio fue sencillo mediante la función de alineación molecular proporcionada por *ChemVA*, según lx expertx a cargo de la evaluación. Cada subconjunto de compuestos químicos pudo ser alineado rápidamente a demanda, a diferencia de otras herramientas de software químico, donde esta tarea suele ser un proceso arduo y a menudo está

restringida a moléculas por sobre cierto umbral de similitud estructural [142]. Además, el filtro de opacidad y la opción de invertir opacidad de dicha vista, por medio de los cuales se pueden resaltar las partes comunes y diferentes de los compuestos, permitieron una exploración profunda de los patrones estructurales comunes.

La coordinación de las distintas vistas con la vista Tabular también resultó útil para lxs expertxs (R4), al ofrecer un acceso rápido a la información resumida en los encabezados de las columnas. La vista Tabular demostró ser efectiva para explorar un conjunto de datos numeroso como lo es el conjunto de inhibidores de la *P-glicoproteína*, permitiendo la búsqueda de compuestos exhibiendo singularidades en sus valores de características moleculares a través de las operaciones de filtrado y agrupación jerárquica durante el segundo caso de estudio. Otras funcionalidades, como mostrar fórmulas SMILES para copiar, la opción de exportar estructuras 3D o buscar compuestos por fórmula SMILES, convirtieron a *ChemVA* en una herramienta flexible y fácilmente interoperable con otras herramientas de química informática.

*ChemVA* demostró ser útil tanto como herramienta de exploración y visualización, como en el diseño y evaluación de nuevos compuestos durante el primer caso de estudio (R6). Como se discutió en la sección 7.7.2, *ChemVA* también fue elogiado por lxs expertxs como una herramienta útil e innovadora durante la evaluación externa.

Una limitación identificada en los casos de estudio es que los nuevos compuestos no siempre se proyectaron cerca de compuestos estructuralmente similares. Esta funcionalidad operó en algunas representaciones moleculares de manera más consistente que en otras, como se puede observar en la figura 7.9. Esto podría estar relacionado con el hecho de que el conjunto de datos de serotonina-dopamina utilizado en el experimento correspondiente al primer caso de estudio es relativamente pequeño (118 compuestos), por lo que el modelo paramétrico basado en redes neuronales entrenado para proyectar nuevos compuestos no fue capaz de generalizar para cada representación molecular.

También identificamos posibles direcciones para futuras extensiones de la herramienta, teniendo en cuenta tanto la experiencia de usarix como aspectos de funcionalidad. Lxs expertxs externxs sugirieron ampliar la funcionalidad actual de la herramienta, permitiendo a lx usarix cargar y personalizar sus propios datos. Unx de lxs expertxs también sugirió habilitar la adición de nuevos compuestos dibujándolos en un editor gráfico similar a la herramienta *ChemDoodle* [381] en lugar de usar únicamente fórmulas SMILES. Otras posibles direcciones incluyen proporcionar opciones para la búsqueda de subestructuras y la posibilidad de guardar capturas de análisis. Estas sugerencias indican un interés en una mayor personalización y flexibilidad en la forma en que lxs usarixs pueden interactuar con los datos y las visualizaciones en *ChemVA*.

En resumen, la evaluación de *ChemVA* a través de casos de estudio y la retroalimentación de especialistas independientes en informática molecular y visualización respaldan su utilidad para tareas de diseño y exploración de fármacos en el contexto de cribado virtual. La herramienta ha demostrado ser efectiva en la comparación de compuestos, la exploración de proyecciones y la identificación de determinantes químicos de la actividad biológica. Sin embargo, también se señaló la necesidad de una documentación más detallada para algunas características y la posibilidad de mejorar las opciones de personalización y flexibilidad en la interacción con la herramienta.

Por último, cabe señalar que aunque nuestra herramienta está diseñada para un dominio específico, *ChemVA* ha sido desarrollada utilizando técnicas y estrategias que podrían ser aplicadas a otros tipos de datos, incluyendo aquellos que no sean de naturaleza química. Por ejemplo, la vista de Contraste propuesta podría ayudar en el análisis visual de la preservación de grupos y vecindarios en varios métodos de agrupamiento de datos de diferentes tipos. Del mismo modo, el análisis de la confiabilidad de las proyecciones 2D podría ser aplicable a cualquier tipo de datos sometidos a un proceso de reducción de dimensionalidad. El concepto detrás de la vista 3D, que facilita la visualización de subestructuras comunes superpuestas, podría ser adaptado a otros tipos de datos basados en grafos o incluso a datos de texto. Esto demuestra el potencial de versatilidad de los métodos y conceptos detrás del diseño funcional de *ChemVA*.

## 7.9. Síntesis y conclusiones

En este capítulo de la tesis abordamos algunas de las problemáticas existentes en el contexto del cribado virtual asistido por herramientas de analítica visual. La exploración detallada y, a la vez, intuitiva, de grandes conjuntos de compuestos químicos, haciendo foco especialmente en el hallazgo de patrones de similitud entre moléculas, requiere de la aplicación de técnicas de reducción dimensional que permitan visualizar los compuestos y su relación espacial. Independientemente de ello, a la hora de analizar los espacios moleculares en sus proyecciones de baja dimensionalidad, resulta difícil establecer criterios que permitan determinar la confiabilidad de dichas proyecciones, aprovechar de forma óptima la información complementaria de diferentes representaciones moleculares, y combinar de forma estratégica diferentes fuentes de información molecular, como pueden serlo descriptores moleculares específicos o información sobre el perfil farmacológico de los compuestos.

En este sentido presentamos *ChemVA*, una herramienta novedosa para el análisis visual de compuestos químicos, especialmente enfocada en la evaluación de la similitud molecular para el cribado virtual. Nuestra herramienta propone un conjunto de vistas coordinadas que brindan

soporte para la exploración visual y la comparación de proyecciones 2D, obtenidas mediante la aplicación de técnicas de reducción de dimensionalidad en diferentes representaciones moleculares complementarias. Nuestra herramienta incorpora un enfoque novedoso, que hace uso de diferentes fuentes de información química para elaborar un perfil químico, físico y biológico completo de cada compuesto en estudio. Para ello, sacamos provecho de la complementariedad entre diferentes representaciones moleculares a base de vectores, algunas de ellas tradicionales y otras obtenidas a base de modelos de aprendizaje profundo.

Nuestra herramienta fue validada en dos casos de estudio y en una evaluación cualitativa independiente. Nuestros casos de estudio, realizados por especialistas en química medicinal, confirmaron que *ChemVA* cumple con los requisitos funcionales y brinda el apoyo adecuado para tareas específicas del proceso de cribado virtual de fármacos en dos conjuntos de datos diferentes. La evaluación cualitativa realizada por tres expertos externos reveló el potencial de nuestra herramienta para su adopción en el campo, así como la utilidad de nuestra nueva vista de Contraste, la cual constituye un aporte novedoso en el área de visualización de grandes volúmenes de datos. El proceso de evaluación nos permitió identificar posibles extensiones para *ChemVA*, que guiarán nuestro trabajo futuro en la herramienta.

El desarrollo de *ChemVA* nos permitió profundizar nuestro conocimiento sobre la complementariedad de diferentes representaciones moleculares y la riqueza de información detrás de ellas, la cual requiere de herramientas y técnicas flexibles y versátiles para ser explotada. Este desarrollo nos dio la oportunidad de abordar múltiples tópicos de investigación profundamente vinculados a la tesis, como lo es el modelado predictivo, estrategias de minería y visualización de grandes volúmenes de datos no estructurados, representaciones moleculares, analítica visual y su uso para diseño *de novo* de fármacos, cubriendo un amplio tramo del camino crítico del proceso moderno de descubrimiento y desarrollo de fármacos.

# Capítulo 8

## Conclusiones y trabajo futuro

En este capítulo presentamos las conclusiones generales de este trabajo de tesis. Describimos los aportes realizados en las temáticas abordadas, brindando un breve resumen de las principales contribuciones de la tesis. Finalmente, proponemos algunas posibles líneas de trabajo futuro derivadas de las investigaciones desarrolladas en el marco de esta tesis.

---

El objetivo general de esta tesis doctoral consistió en el diseño, implementación y validación de estrategias computacionales novedosas basadas en aprendizaje automático y profundo para contribuir en diferentes fases del proceso de desarrollo de medicamentos. Más específicamente, propusimos y desarrollamos un conjunto de enfoques, técnicas y herramientas computacionales a distintos niveles de granularidad, orientados a mejorar el estado del arte en materia de modelado predictivo de perfiles de bioactividad y la definición de dominios de aplicabilidad de dichos modelos, así como también en lo que respecta al diseño, obtención y aplicación de representaciones moleculares basadas en vectores y analítica visual para cribado virtual de fármacos.

El proceso de desarrollo de un nuevo fármaco requiere de múltiples etapas que insumen tiempo e inversiones millonarias, con el agravante de que la probabilidad de lograr un fármaco aprobado es muy baja. En este contexto, la incorporación de estrategias de aprendizaje automático e inteligencia artificial en etapas tempranas del proceso y su integración con conocimiento experto del dominio de química medicinal y farmacia han revolucionado la disciplina, mejorando su eficacia, precisión y reduciendo los costos asociados. Entre las múltiples tareas en el proceso de desarrollo de medicamentos en las que se integran estrategias computacionales, destacamos al modelado QSAR, a las técnicas de cribado virtual de fármacos y al aprendizaje de representaciones moleculares como pilares

fundamentales en la identificación temprana de compuestos candidatos. Estas tres tareas son en las que hemos centrado los desarrollos propuestos en el marco de la presente tesis.

Durante los últimos años han proliferado las publicaciones científicas que abordan la creación de algoritmos novedosos basados en aprendizaje profundo para obtener representaciones moleculares, denominadas *embeddings*. El proceso de selección de la representación molecular a emplear en una tarea dada no resulta trivial, puesto que requiere un estricto y detallado diseño experimental que contemple una variedad de posibles escenarios de modelado y que tenga en consideración las diferencias en el aprendizaje de las representaciones dadas por las características del algoritmo empleado para obtener dichos *embeddings*. No obstante, la correcta elección de las representaciones moleculares empleadas resulta determinante en la confiabilidad y precisión de los resultados, por lo que resulta fundamental proporcionar estrategias que permitan analizar distintas alternativas de representación molecular y ponderar sus virtudes y limitaciones en modelado QSAR y en otras tareas involucradas en el desarrollo de fármacos.

En el contexto del modelado QSAR, la constante ampliación y publicación de bases de datos de compuestos químicos etiquetados y el desarrollo de nuevas representaciones moleculares proponen interesantes desafíos en términos de rendimiento predictivo e interpretabilidad de modelos. Los algoritmos de aprendizaje automático tradicionalmente empleados en el modelado QSAR han exhibido históricamente dificultades para lidiar con representaciones moleculares de alta dimensionalidad, por lo que siempre ha resultado necesario recurrir a un paso previo de selección de características, con la consecuente simplificación del problema a modelar y la potencial pérdida de información. Los modelos basados en aprendizaje profundo resultan especialmente propicios para mitigar esta problemática, aunque a expensas de una fase de selección de modelos donde se analicen diferentes parametrizaciones y de una baja interpretabilidad. En este contexto, resulta crucial desarrollar estrategias de modelado integrales, que permitan aprender patrones significativos de datos de alta dimensionalidad de forma precisa, brindando al mismo tiempo herramientas para la determinación efectiva del dominio de aplicabilidad de los modelos generados y la interpretación fiable de sus predicciones.

Una vez entrenado y validado, un modelo QSAR se emplea para predecir el perfil de bioactividad de un nuevo compuesto en proceso de evaluación como candidato a fármaco. Como hemos discutido en capítulos previos de la presente tesis, los mecanismos de interacción y propiedades farmacodinámicas de los compuestos químicos son complejos y es muy común encontrar múltiples receptores biológicos con los que un mismo compuesto interactúa de formas diferentes. La incorporación de información de múltiples blancos biológicos complementarios en el proceso de modelado puede resultar determinante

en el desempeño predictivo de un modelo dado y representa un nicho a explorar, con el potencial de mejorar drásticamente los resultados obtenidos para propiedades conocidas o incluso de permitir el modelado de propiedades complejas y sub-exploradas.

En el área de cribado virtual de fármacos, donde se exploran enormes espacios químicos de gran dimensionalidad, un aspecto fundamental es el análisis preciso y confiable de similitud entre compuestos químicos, tanto en términos de su estructura como de sus propiedades farmacodinámicas y farmacocinéticas. La identificación de determinantes de similitud entre dos compuestos candidatos no resulta trivial, sobre todo considerando que la exploración de dichos espacios moleculares requiere la proyección en baja dimensionalidad de los conjuntos de compuestos químicos, con todos los desafíos que tal proceso conlleva. En este contexto, las estrategias de analítica visual son aliadas indispensables para asistir a expertos en el dominio, resultando fundamental la integración y coordinación de representaciones moleculares y fuentes de información química complementarias.

El desarrollo de los trabajos de investigación presentados en esta tesis fue delineado por los objetivos y preguntas de investigación planteadas inicialmente en el capítulo 1. Dichos objetivos tuvieron origen en el estudio permanente del área de aprendizaje profundo e informática molecular para identificar direcciones y tendencias salientes. Las contribuciones de esta tesis abordaron desafíos y limitaciones no resueltos por las propuestas anteriores, mejorando el estado del arte en múltiples tareas involucradas en el proceso de descubrimiento de nuevos fármacos. El abordaje del diseño y desarrollo experimental de nuestros trabajos se basó en el estudio e implementación de estrategias y conceptos de vanguardia del aprendizaje profundo, algunos de ellos vinculados a áreas no relacionadas con la química medicinal. Por esta razón, si bien las contribuciones de la presente tesis se presentan en el área de la informática molecular, se tratan de desarrollos extensibles y aplicables a otras áreas de investigación de las ciencias de la computación.

## 8.1. Resumen de las contribuciones realizadas

En primera instancia, nuestras contribuciones se orientaron al modelado predictivo de propiedades trascendentales en la química medicinal, empleando representaciones moleculares de alta dimensionalidad y proponiendo enfoques novedosos basados en aprendizaje profundo que no requieren del paso previo de selección de características, empleado por defecto en la mayoría de los trabajos de investigación publicados previamente. Tal y como describimos en el capítulo 4, propusimos un enfoque integral de modelado QSAR para la predicción de biodegradabilidad y de la interacción entre compuestos candidatos y los *citocromos P450* en las isoformas CYP2C9 y CYP3A4,



propiedades trascendentales en el área de la química medicinal, que integra además una estrategia de definición de dominio de aplicabilidad y de interpretabilidad *a posteriori* [322]. Nuestra estrategia de modelado logró un rendimiento predictivo ampliamente superior al de las técnicas establecidas como estado del arte, confirmando nuestra hipótesis de que la selección de características conlleva pérdida de información crucial para el rendimiento predictivo de los modelos. Nuestra estrategia de definición del dominio de aplicabilidad se basó en la estimación de confianza del modelo entrenado, computada a partir de las probabilidades de salida de la red neuronal, resultando eficaz por cuanto demostró tener una fuerte correlación con los resultados satisfactorios de dicho modelo. Además, en dicho trabajo se proporcionó una técnica novedosa para brindar interpretabilidad *a posteriori* del modelo, la cual se basa en la agregación de los parámetros entrenables de la red neuronal que permitió identificar correcta y consistentemente los descriptores moleculares y características de mayor incidencia en el proceso de modelado. Esta técnica constituye un avance en materia de interpretabilidad de modelos basados en redes neuronales, cuya adopción en el dominio de informática molecular se vio históricamente desacelerada precisamente por su escasa explicabilidad.

También enmarcado en la subdisciplina de modelado QSAR, implementamos una estrategia de modelado novedosa para la predicción de mutagenicidad de Ames basada en redes neuronales profundas y empleando un enfoque de aprendizaje multi-tarea (MTL) [248]. La mutagenicidad de Ames es una propiedad que normalmente se modela por medio de aprendizaje de tarea única, lo cual implica una simplificación del análisis de los resultados de mutagenicidad para diferentes cepas bacterianas. La experimentación *in vitro*, posterior al modelado QSAR, suele dar resultados contradictorios con estos modelos simples. Nuestro modelo, descrito en detalle en el capítulo 5, superó el estado del arte en la predicción de mutagenicidad de Ames al combinar información experimental de cinco cepas de *Salmonella typhimurium*, permitiendo no solamente predecir la mutagenicidad de un compuesto candidato para cada uno de los blancos biológicos por separado, sino además modelando exitosamente la mutagenicidad como propiedad global por medio del aprendizaje conjunto de los determinantes farmacodinámicos asociados a cada una de las cepas. Nuestro enfoque constituyó la primer propuesta de aprendizaje multi-tarea publicada para esta propiedad, y los resultados obtenidos sustentan nuestra hipótesis de que el aprendizaje multi-tarea permite el modelado de propiedades complejas, en entornos de fuerte desbalance de clases, y permite aprovechar al máximo la información complementaria brindada por múltiples receptores biológicos y sus complejas interacciones en organismos vivos. Por último, nuestro modelo resulta de especial utilidad para expertxs en el área, por cuanto permite modelar incluso en ausencia de información experimental para algunas cepas, y favorece a la interpretabilidad de las predicciones de mutagenicidad finales al indicar la predicción realizada para cada cepa en particular.

De la mano con los avances vertiginosos en el desarrollo de nuevas arquitecturas de aprendizaje profundo, durante los últimos años se experimentó una gran proliferación de algoritmos para la obtención de *embeddings* moleculares. Estas representaciones, ricas en información estructural y química, comenzaron a ser utilizadas en diversas tareas del desarrollo de fármacos, entre ellas el modelado QSAR. No obstante, a pesar del visible compromiso de la comunidad científica con este tópico de investigación, no encontramos publicaciones previas a esta tesis orientadas a constituir un marco de trabajo o un diseño experimental riguroso para la evaluación de *embeddings* moleculares en el contexto de modelado QSAR. En lugar de ello, en la mayor parte de las publicaciones científicas en las que se propusieron modelos QSAR entrenados a partir de *embeddings* moleculares no se explicitó el hallazgo de evidencia de que el desempeño de dichas representaciones impactara significativamente en el modelado, en contraste al impacto de otras representaciones tradicionalmente utilizadas en el área. En el capítulo 6 presentamos un minucioso trabajo de investigación y análisis comparativo de diferentes estrategias de representación molecular, comprendiendo tanto representaciones tradicionales como *embeddings* obtenidos a partir de arquitecturas neuronales profundas [324]. Nuestro diseño experimental contrastó cinco estrategias diferentes de *embedding* y tres representaciones tradicionales, empleando múltiples variantes de canonicalización de fórmulas SMILES y considerando varias dimensionalidades y complejidades de representación. Nuestros experimentos devinieron en el entrenamiento y validación de más de 25.000 modelos QSAR para cinco tareas de clasificación y tres de regresión. Nuestros resultados, sometidos a rigurosas evaluaciones de significancia estadística, no arrojaron evidencia que sustente la hipótesis de que el uso de *embeddings* moleculares implique una mejora significativa en el desempeño de los modelos QSAR con respecto al uso de representaciones tradicionales. Estos resultados, que *a priori* parecen refutar las direcciones de investigación actuales, ponen de manifiesto la imperiosa necesidad de conducir un análisis y selección rigurosa de la representación molecular empleada en las diversas tareas de desarrollo de fármacos, sirviendo además nuestro diseño experimental como marco de referencia para la conducción de análisis comparativos de similar tenor.

Por último, ya en el contexto del cribado virtual de fármacos, en el capítulo 7 presentamos una herramienta integral e interactiva de analítica visual denominada *ChemVA* [323]. Esta herramienta integró conceptos y técnicas asociados al análisis de similitud estructural a partir de diversas representaciones moleculares y estrategias de reducción dimensional, e involucró el estudio y aplicación de conceptos fundamentales de esta tesis, tales como el modelado predictivo, el desarrollo de representaciones moleculares, el diseño de estrategias de minería y la visualización de grandes volúmenes de datos. La integración de tales enfoques derivó en la creación de una herramienta de analítica visual para cribado virtual y diseño *de novo* de fármacos, abarcando un segmento

significativo del proceso actual de descubrimiento de medicamentos. El desarrollo de *ChemVA* se fundó en el uso de diferentes fuentes de información química y estructural que permitiera la elaboración de un perfil farmacológico completo de los compuestos candidatos, basándose en la complementariedad de diferentes representaciones moleculares, tanto tradicionales como *embeddings*. *ChemVA* consta de una distribución en cuadrícula de vistas coordinadas diseñadas para permitir la inspección visual e interactiva de conjuntos de datos por medio de proyecciones 2D y 3D. Además, permite contrastar las proyecciones obtenidas a partir de diferentes representaciones moleculares y sus agrupamientos, brindando soporte a expertos en el delicado proceso de cribado virtual. Por último, proporciona facilidades para el diseño *de novo* de compuestos químicos, permitiendo la incorporación de estructuras moleculares nuevas a las visualizaciones de los conjuntos de datos en estudio. El proceso de validación de *ChemVA* constó de su uso en dos casos de estudio por parte de un experto en farmacia y en una evaluación cualitativa por parte de expertos en visualización y computación gráfica. Según tales evaluaciones independientes, la herramienta cumplió con los requisitos funcionales necesarios para el desarrollo de tareas vinculadas al cribado virtual de fármacos y diseño *de novo*, constituyendo un aporte significativo y novedoso en el área.

## 8.2. Trabajo futuro

Como líneas de trabajo futuro, además de la mejora continua de los métodos y propuestas ya desarrollados en la presente tesis, proyectamos trabajar en tres tópicos íntimamente relacionados con los explorados en este trabajo. En primer lugar, evaluamos la posibilidad de emplear modelos basados en aprendizaje profundo para la obtención de *embeddings contextuales*, orientados al reposicionamiento de fármacos. Nuestra idea consiste en combinar estos modelos con estrategias de entrenamiento multimodal, que empleen fuentes diversas de información físico-química, tales como estructuras moleculares, texto, estructuras de receptores biológicos, etc., a fin de confeccionar *embeddings* moleculares integrales del perfil farmacológico de los compuestos. Sostenemos la hipótesis de que dichas representaciones permitirían evaluar compuestos ya aprobados para ciertos blancos farmacológicos como potenciales ligandos a nuevos blancos con una mayor efectividad. Hemos realizado experimentos vinculados a esta área de investigación en el marco de un proyecto galardonado

por *Google Latin America Research Awards* en 2020<sup>1</sup> y 2021<sup>2,3</sup>.

En segundo lugar, evaluamos el desarrollo de estrategias de exploración guiadas del espacio latente de un modelo generativo basadas en algoritmos evolutivos multi-objetivo. Una vez que se ha entrenado un modelo generativo de compuestos químicos, el espacio latente de baja dimensionalidad de dicho modelo permite el muestreo y obtención de nuevos compuestos, tal y como se explicó en el capítulo 2. La exploración guiada de dicho espacio latente permitiría obtener compuestos novedosos para propiedades farmacodinámicas y estructurales deseadas. Actualmente, nos encontramos desarrollando una estrategia novedosa de tales características, basada en un algoritmo evolutivo nuevo que integra componentes de algoritmos evolutivos multi-objetivo clásicos.

Por último, una extensión interesante a los enfoques presentados como líneas de trabajo futuro y a lo largo de esta tesis consiste en la incorporación de funcionalidades para integrar conocimiento experto de forma interactiva en los procesos de aprendizaje automáticos. Estas estrategias, conocidas por su nombre en inglés como *human in the loop*, tienen el potencial de mejorar significativamente los procesos de aprendizaje y exploración. Vislumbramos su aplicación exitosa en tareas como el cómputo de representaciones moleculares, el desarrollo de modelos predictivos para bioactividad y la exploración de espacios moleculares latentes.

---

<sup>1</sup>Anuncio Google LARA 2020: <https://blog.google/intl/es-419/noticias-de-la-empresa/de-google/estos-son-los-ganadores-de-la-edicion/>

<sup>2</sup>Anuncio Google LARA 2021 (AR): <https://blog.google/intl/es-419/noticias-de-la-empresa/de-google/premios-lara-novena-edicion-estos-son-los-24-ganadores/>

<sup>3</sup>Anuncio Google LARA 2021 (BR): <https://blog.google/intl/pt-br/novidades/iniciativas/conheca-os-vencedores-do-premio-lara-2021-o-programa-de-bolsas-de-pesquisa-do-google/>

# Apéndice A

## Hiperparámetros evaluados durante las etapas de selección de modelos en la evaluación de representaciones moleculares (capítulo 6)

### Búsqueda exploratoria de hiperparámetros para modelos de *embedding* molecular supervisados

- *PaccMann*: En la capa de *Self-Attention*, variamos el número de nodos en la capa oculta (16, 50 y 100) y el valor de profundidad de atención (20, 50, 100 y 256). En la red neuronal de alimentación hacia adelante, utilizada para la predicción de propiedades, variamos el número de nodos en las capas ocultas ([512, 256, 64, 16], [100, 50, 20, 5], [150, 50, 10] y [100, 20, 5]) y las funciones de activación (*ReLU* y *sigmoidea*) [347] de las capas. Evaluamos el desempeño del modelo de *embedding* con y sin una función de codificación posicional sinusoidal aplicada a las entradas [397], y con y sin una función de pérdida ponderada basada en el desequilibrio en cada clase [91]. Además, variamos los hiperparámetros de regularización; entre ellos, la tasa de *dropout* por capa (0,5, 0,25 y 0,15) y el coeficiente *lambda* de regularización *L2* (0,0001, 0,005 y 0,001) [262]. El *learning rate* se fijó en 0,001, y empleamos el optimizador *Adam* [198]. Utilizamos un tamaño de *minibatch* de 512 para los conjuntos de datos pequeños (*SR-ARE*, *SR-MMP*, *SR-ATAD5*, *ESOL*, *FreeSolv* y *Lipophilicity*) y un tamaño de *minibatch* de 2048

para los conjuntos de datos grandes (*HIV*, *PCBA-686978*). Como resultado de este proceso de búsqueda exploratoria, evaluamos un total de 175 combinaciones diferentes de hiperparámetros para cada conjunto de datos etiquetado.

- *SA-BiLSTM*: Considerando que cada ciclo de entrenamiento de este método conlleva un tiempo significativo, decidimos centrarnos en los hiperparámetros probados por los autores en el artículo original [449]. Variamos el número de cabezales de atención (5, 10, 15 y 18) y el valor de profundidad de atención (10, 20, 50 y 100) en la capa de *Self-Attention*, así como el número de unidades ocultas en la capa recurrente bidireccional *Bi-LSTM* (64 y 128). Siguiendo los valores indicados en el artículo de referencia, utilizamos un tamaño de *minibatch* de 64 para todos los conjuntos de datos y un coeficiente de *gradient clipping* de 0,3. La tasa de *dropout* por capa se fijó en 0,2 y el valor del coeficiente  $\lambda$  para la regularización L2 se fijó en 0,01 [262]. Los modelos fueron entrenados durante un máximo de 1.000 épocas, según lo indicado en el artículo de referencia. Entrenamos con y sin la aplicación de una función de pérdida ponderada para compensar el desequilibrio de clases. Como resultado de esta etapa de búsqueda exploratoria, evaluamos un total de 64 combinaciones diferentes de hiperparámetros para cada conjunto de datos etiquetado.

## Búsqueda exploratoria de hiperparámetros para clasificadores y modelos de regresión

Para cada método de clasificación, realizamos una búsqueda exploratoria de hiperparámetros utilizando un particionado fijo y estratificado de los datos. En el caso de las *SVM*, variamos el coeficiente de regularización  $c \in \{0,01, 0,05, 0,1, 0,25, 0,5, 1, 5, 10, 20\}$ , y para los modelos basados en *RF* variamos la profundidad máxima del árbol  $md \in \{2, 3, 5, 8, 10, 20\}$ . Para las *FFNN*, variamos el número de nodos en cada capa ( $[150, 50, 10]$ ,  $[100, 50, 10]$  y  $[100, 20, 5]$ ), el coeficiente de regularización L2  $\lambda \in \{0,0001, 0,005, 0,001\}$ , el tamaño del *minibatch*  $b \in \{64, 128, 256, 512\}$ , la función de activación utilizada en las capas ocultas (*ReLU* y *tanh*) y el coeficiente de *paciencia* para *early stopping*  $p \in \{70, 100, 200, 500\}$ . También probamos diferentes valores de *learning rate*  $\alpha \in \{0,00001, 0,0001, 0,001\}$ .

En el caso de los modelos de regresión, para la regresión *Ridge* variamos el coeficiente de regularización  $\alpha \in \{0,001, 0,05, 0,1, 0,5, 1, 10\}$ , la tolerancia  $tol \in \{0,00001, 0,0001, 0,001\}$  y establecimos el número máximo de iteraciones para la convergencia en  $max\_iter = 10.000$ . Para

el modelo basado en *GBR*, variamos el número de estimadores  $n\_estimators \in \{50, 100, 200, 500\}$ , la profundidad máxima del árbol  $max\_depth \in \{2, 3, 5, 8, 10, 20\}$  y la cantidad mínima de muestras requeridas para dividir un nodo interno  $min\_samples\_split \in \{2, 3\}$ . Finalmente, para las *FFNN*, variamos el número de nodos en cada capa ( $[200, 70, 15]$ ,  $[150, 50, 10]$ ,  $[100, 50, 10]$  y  $[100, 20, 5]$ ), el coeficiente de regularización L2  $\lambda \in \{0,0001, 0,005, 0,001, 0,01, 0,1\}$ , la función de activación utilizada en las capas ocultas (*ReLU* y *tanh*), el coeficiente de *paciencia* para *early stopping*  $p \in \{100, 200, 500\}$  y el valor de *learning rate*  $\alpha \in \{0,00001, 0,0001, 0,001\}$ . Establecimos el tamaño del *minibatch* en  $b = 200$  y también probamos dos funciones de pérdida: *RMSE* y *MSE*.

# Apéndice B

## Listado de abreviaciones

- **Acc:** del inglés *Accuracy*, Exactitud.
- **ADME-Tox:** del inglés *Absorption, Distribution, Metabolism, Excretion, Toxicity*, Absorción, Distribución, Metabolismo, Excreción, Toxicidad.
- **ADN/ARN:** Ácido Desoxirribonucleico / Ácido Ribonucleico.
- **ANOVA:** del inglés *ANalysis Of VAriance*, Análisis de la Varianza.
- **ASCII:** del inglés *American Standard Code for Information Interchange*, Código Estándar estadounidense para el Intercambio de Información.
- **AUC, AUC-ROC:** del inglés *Area Under the Curve - Receiver Operating Characteristic*, Área Bajo la Curva de Característica Operativa del Receptor.
- **AV:** Analítica Visual.
- **BAcc:** del inglés *Balanced Accuracy*, Exactitud Balanceada.
- **BCE:** del inglés *Binary Cross-Entropy*, Entropía Cruzada Binaria.
- **BGD:** del inglés *Batch Gradient Descent*, Descenso por el Gradiente por Lotes.
- **Bi-LSTM:** del inglés *Bidirectional Long-Short Term Memory*.
- **BN:** del inglés *Batch Normalization*, Normalización por Lotes.
- **COVID-19:** del inglés *COronaVIRus Disease 19*, Enfermedad del Coronavirus 19.



- **CNN**: del inglés *Convolutional Neural Network*, Red Neuronal Convolutacional.
- **DNN**: del inglés *Deep Neural Network*, Red Neuronal Profunda.
- **DBSCAN**: del inglés *Density-Based Spatial Clustering of Applications with Noise*, Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido.
- **ECFP**: del inglés *Extended Connectivity FingerPrint*, *Fingerprint* de Conectividad Extendida.
- **EC50**: del inglés *Effective Concentration 50*, Concentración Efectiva 50.
- **EMA**: del inglés *European Medicines Agency*, Agencia Europea de Medicina.
- **F1**: del inglés *F1 Score*, Puntaje F1.
- **FDA**: del inglés *Food and Drug Administration*, agencia de Administración de Alimentos y Medicamentos.
- **FFNN**: del inglés *Feed-Forward Neural Network*, Red Neuronal de Alimentación hacia Adelante.
- **FN**: del inglés *False Negatives*, Falsos Negativos.
- **FP**: del inglés *False Positives*, Falsos Positivos.
- **GAN**: del inglés *Generative Adversarial Networks*, Redes Generativas Adversarias.
- **GBR**: del inglés *Gradient Boosting Regressor*, Regresor de Potenciación del Gradiente.
- **GNN**: del inglés *Graph Neural Networks*, Redes Neuronales de Grafos.
- **GN**: del inglés *Group Normalization*, Normalización por Grupos.
- **GRU**: del inglés *Gated Recurrent Unit*.
- **H1**: del inglés *Harmonic 1 Score*, Puntaje H1.
- **HBA**: del inglés *Hydrogen Bond Acceptors*, Aceptadores de Puentes de Hidrógeno.
- **HBD**: del inglés *Hydrogen Bond Donors*, Donantes de Puentes de Hidrógeno.
- **IC50**: del inglés *Inhibitory Concentration 50*, Concentración Inhibitoria 50.

- **IUPAC**: del inglés *International Union of Pure and Applied Chemistry*, Unión Internacional de Química Pura y Aplicada.
- **IN**: del inglés *Instance Normalization*, Normalización por Instancia.
- **InChI**: del inglés *International Chemical Identifier*, Identificador Químico Internacional.
- **KL Divergence**: del inglés *Kullback-Leibler Divergence*, Divergencia de Kullback-Leibler.
- **k-NN**: del inglés *k-Nearest Neighbors*, k Vecinos más Cercanos.
- **Lipinski's RO5**: del inglés *Lipinski's Rule Of Five*, Regla de los Cinco de Lipinski.
- **LDA**: del inglés *Linear Discriminant Analysis*, Análisis del Discriminante Lineal.
- **LN**: del inglés *Layer Normalization*, Normalización por Capas.
- **LogP**: Logaritmo del coeficiente de partición octanol-agua.
- **MACCS**: del inglés *Molecular ACCess System*.
- **MAP**: del inglés *Mean Average Precision*.
- **MCS**: del inglés *Maximum Common Substructure*, Subestructura Común Máxima.
- **MDS**: del inglés *MultiDimensional Scaling*, Escalado Multidimensional.
- **MAE**: del inglés *Mean Absolute Error*, Error Medio Absoluto.
- **MCC**: del inglés *Matthew's Correlation Coefficient*, Coeficiente de Correlación de Matthews.
- **MTL**: del inglés *Multi-Task Learning*, Aprendizaje Multi-Tarea.
- **MSR**: del inglés *Multivariate Subspace Regression*, Regresión Multivariada de Subespacios.
- **MW**: del inglés *Molecular Weight*, Peso Molecular.
- **NER**: del inglés *Non-Error Rate*, Tasa de No-Error.
- **NLP**: del inglés *Natural Language Processing*, Procesamiento del Lenguaje Natural.
- **NB**: del inglés *Naïve Bayes*.

- **OECD**: del inglés *Organization for Economic Cooperation and Development*, Organización para la Cooperación Económica y Desarrollo.
- **PCA**: del inglés *Principal Component Analysis*, Análisis de Componentes Principales.
- **QED**: del inglés *Quantitative Estimate of Drug-likeness*, Estimación Cuantitativa de Afinidad a Fármacos.
- **QSAR**: del inglés *Quantitative Structure-Activity Relationship*, Relación Cuantitativa de Estructura-Actividad.
- $R^2$ : Coeficiente de Determinación.
- **RBF**: del inglés *Radial Basis Function*, Función Base Radial.
- **ReLU**: del inglés *Rectified Linear Unit*, Unidad Rectificada Lineal.
- **RF**: del inglés *Random Forests*, Bosques Aleatorios.
- **RNN**: del inglés *Recurrent Neural Network*, Red Neuronal Recurrente.
- **RMSE**: del inglés *Root Mean Squared Error*, Raíz del Error Medio Cuadrático.
- **RMSProp**: del inglés *Root Mean Squared Propagation*, Propagación de la Raíz del Error Medio Cuadrático.
- **RB**: del inglés *Ready Biodegradability*, Biodegradabilidad Simple.
- **SA**: del inglés *Synthetic Accessibility*, Accesibilidad Sintética.
- **SGD**: del inglés *Stochastic Gradient Descent*, Descenso por el Gradiente Estocástico.
- **SDF**: del inglés *Structure-Data File*, Archivo Estructura-Dato.
- **SELFIES**: del inglés *SELF-referencing Embedded Strings*, Cadenas Embebidas Auto-Referenciales.
- **SMARTS**: del inglés *SMILES Arbitrary Target Specification*, Especificación Arbitraria de SMILES Objetivo.
- **Sn**: del inglés *Sensitivity*, Sensibilidad.
- **SOM**: del inglés *Self-Organizing Maps*, Mapas Auto-Organizados.

- **SMILES**: del inglés *Simplified Molecular Input Line Entry System*, Sistema Simplificado Lineal de Entrada Molecular.
- **Sp**: del inglés *Specificity*, Especificidad.
- **STL**: del inglés *Single-Task Learning*, Aprendizaje de Tarea Única.
- **SVM**: del inglés *Support Vector Machine*, Máquina de Vectores de Soporte.
- **TP**: del inglés *True Positives*, Verdaderos Positivos.
- **TN**: del inglés *True Negatives*, Verdaderos Negativos.
- **t-SNE**, **pt-SNE**: del inglés *(parametric) t-Distributed Stochastic Neighbor Embedding*, Incrustación (paramétrica) de Vecinos Estocásticos t-Distribuidos.
- **VAE**: del inglés *Variational Auto-Encoder*, *Auto-Encoder* Variacional.
- **XAI**: del inglés *eXplainable Artificial Intelligence*, Inteligencia Artificial Explicable.

# Bibliografía

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- [2] C. Abate, S. Decherchi, and A. Cavalli. Graph neural networks for conditional de novo drug design. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, page e1651, 2023.
- [3] H. Abdi and L. J. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.
- [4] N. Adams and P. Murray-Rust. Engineering polymer informatics: towards the computer-aided design of polymers. *Macromolecular rapid communications*, 29(8):615–632, 2008.
- [5] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional space. In *Database Theory—ICDT 2001: 8th International Conference London, UK, January 4–6, 2001 Proceedings 8*, pages 420–434. Springer, 2001.
- [6] C. Agoni, F. A. Olotu, P. Ramharack, and M. E. Soliman. Druggability and drug-likeness concepts in drug design: are biomodelling and predictive tools having their say? *Journal of molecular modeling*, 26:1–11, 2020.
- [7] M. Ahmed, R. Seraj, and S. M. S. Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.

- [8] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf. A systematic review on supervised and unsupervised machine learning algorithms for data science. *Supervised and unsupervised learning for data science*, pages 3–21, 2020.
- [9] S. Alsenan, I. Al-Turaiki, and A. Hafez. Autoencoder-based dimensionality reduction for qsar modeling. In *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, pages 1–4. IEEE, 2020.
- [10] V. M. Alves, A. Golbraikh, S. J. Capuzzi, K. Liu, W. I. Lam, D. R. Korn, D. Pozefsky, C. H. Andrade, E. N. Muratov, and A. Tropsha. Multi-Descriptor Read Across (MuDRA): A Simple and Transparent Approach for Developing Accurate Quantitative Structure–Activity Relationship Models. *Journal of Chemical Information and Modeling*, 58(6):1214–1223, Junio 2018. doi: 10.1021/acs.jcim.8b00124.
- [11] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan. Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8:1–74, 2021.
- [12] B. N. Ames, F. D. Lee, and W. E. Durston. An improved bacterial test system for the detection and classification of mutagens and carcinogens. *Proc. Natl. Acad. Sci.*, 70(3):782–786, 1973. doi: 10.1073/pnas.70.3.782.
- [13] A. d. Amo, J. Montero, G. Biging, and V. Cutello. Fuzzy classification systems. *European Journal of Operational Research*, 156(2):495–507, 2004.
- [14] P. Angelov and E. Soares. Towards explainable deep neural networks (xdnn). *Neural Networks*, 130:185–194, 2020.
- [15] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete. A survey on modern trainable activation functions. *Neural Networks*, 138:14–32, 2021.
- [16] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.
- [17] H. Askr, E. Elgeldawi, H. Aboul Ella, Y. A. Elshaiar, M. M. Gomaa, and A. E. Hassanien. Deep learning in drug discovery: an integrative review and future challenges. *Artificial Intelligence Review*, 56(7):5975–6037, 2023.

- [18] S. Athmaja, M. Hanumanthappa, and V. Kavitha. A survey of machine learning algorithms for big data analytics. In *2017 International conference on innovations in information, embedded and communication systems (ICIIECS)*, pages 1–4. IEEE, 2017.
- [19] J. L. Atwood and L. J. Barbour. Molecular graphics: from science to art. *Crystal growth & design*, 3(1):3–8, 2003.
- [20] K. Atz, F. Grisoni, and G. Schneider. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.
- [21] M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, 70(7-9):1304–1330, 2007. doi: 10.1016/j.neucom.2006.11.018.
- [22] S. Avram, A. Bora, L. Halip, and R. Curpăn. Modeling Kinase Inhibition Using Highly Confident Data Sets. *Journal of Chemical Information and Modeling*, 58(5):957–967, Mayo 2018. doi: 10.1021/acs.jcim.7b00729.
- [23] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- [24] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLOS ONE*, 10(7):e0130140, Julio 2015. doi: 10.1371/journal.pone.0130140.
- [25] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [26] B. Baillif, J. Cole, P. McCabe, and A. Bender. Deep generative models for 3d molecular structure. *Current Opinion in Structural Biology*, 80:102566, 2023.
- [27] P. Baldi and P. J. Sadowski. Understanding dropout. *Advances in neural information processing systems*, 26, 2013.
- [28] D. Ballabio, F. Biganzoli, R. Todeschini, and V. Consonni. Qualitative consensus of QSAR ready biodegradability predictions. *Toxicological & Environmental Chemistry*, pages 1–24, Diciembre 2016. doi: 10.1080/02772248.2016.1260133.
- [29] D. Bank, N. Koenigstein, and R. Giryes. Autoencoders. *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*, pages 353–374, 2023.

- [30] D. Baptista, J. Correia, B. Pereira, and M. Rocha. Evaluating molecular representations in machine learning models for drug response prediction and interpretability. *Journal of Integrative Bioinformatics*, 19(3):20220006, 2022.
- [31] F. Barbosa and D. Horvath. Molecular similarity and property similarity. *Current Topics in Medicinal Chemistry*, 4(6):589–600, 2004. doi: 10.2174/1568026043451186.
- [32] I. I. Baskin, V. A. Palyulin, and N. S. Zefirov. Neural Networks in Building QSAR Models. In *Artificial Neural Networks*, pages 133–154. Humana Press, 2006. doi: 10.1007/978-1-60327-101-1\_8.
- [33] I. I. Baskin, D. Winkler, and I. V. Tetko. A renaissance of neural networks in drug discovery. *Expert Opinion on Drug Discovery*, 11(8):785–795, Agosto 2016. doi: 10.1080/17460441.2016.1201262.
- [34] D. Baumann and K. Baumann. Reliable estimation of prediction errors for qsar models under model uncertainty using double cross-validation. *Journal of cheminformatics*, 6(1):1–19, 2014.
- [35] A. Bender and R. C. Glen. Molecular similarity: a key technique in molecular informatics. *Organic & biomolecular chemistry*, 2(22):3204–3218, 2004.
- [36] E. Benfenati, A. Golbamaki, G. Raitano, A. Roncaglioni, S. Manganelli, F. Lemke, U. Norinder, E. Lo Piparo, M. Honma, A. Manganaro, et al. A large comparison of integrated sar/qsar models of the ames test for mutagenicity. *SAR QSAR Environ. Res.*, 29(8):591–611, 2018.
- [37] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.
- [38] R. Benigni. Towards quantitative read across: Prediction of ames mutagenicity in a large database. *Regul. Toxicol. Pharmacol.*, 108:104434, 2019.
- [39] R. Benigni, C. L. Battistelli, C. Bossa, O. Tcheremenskaia, and P. Crettaz. New perspectives in toxicological information management, and the role of isstox databases in assessing chemical mutagenicity and carcinogenicity. *Mutagenesis*, 28(4):401–409, 2013.
- [40] R. Benigni, A. Bassan, and M. Pavan. In silico models for genotoxicity and drug regulation. *Expert Opin. Drug Metab. Toxicol.*, 16(8):651–662, 2020.



- [41] A. P. Bento, A. Gaulton, A. Hersey, L. J. Bellis, J. Chambers, M. Davies, F. A. Krüger, Y. Light, L. Mak, S. McGlinchey, et al. The chembl bioactivity database: an update. *Nucleic acids research*, 42(D1):D1083–D1090, 2014.
- [42] F. Berenger and Y. Yamanishi. A Distance-Based Boolean Applicability Domain for Classification of High Throughput Screening Data. *Journal of Chemical Information and Modeling*, page acs.jcim.8b00499, Enero 2019. doi: 10.1021/acs.jcim.8b00499.
- [43] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, P. Bourne, P. Rose, A. Prlic, et al. RCSB protein data bank: Structural biology views for basic and applied research. *Nucleic Acids Research*, 28:235–242, 2000.
- [44] G. R. Bickerton, G. V. Paolini, J. Besnard, S. Muresan, and A. L. Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.
- [45] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- [46] G. Book. Compendium of chemical terminology. *International Union of Pure and Applied Chemistry*, 528, 2014.
- [47] M. Bostock, V. Ogievetsky, and J. Heer. D<sup>3</sup> data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. doi: 10.1109/TVCG.2011.185.
- [48] A. Botchkarev. Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. *arXiv preprint arXiv:1809.03006*, 2018.
- [49] L. Bottou and O. Bousquet. Large-scale machine learning with stochastic gradient descent. *Proceedings of COMPSTAT'2010*, 2010.
- [50] K. Bouhedjar, A. Boukelia, A. Khorief Nacereddine, A. Boucheham, A. Belaidi, and A. Djerourou. A natural language processing approach based on embedding deep learning from heterogeneous compounds for quantitative structure–activity relationship modeling. *Chemical Biology & Drug Design*, 96(3):961–972, 2020.
- [51] N. Brown, M. Fiscato, M. H. Segler, and A. C. Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

- [52] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.
- [53] I. L. Buxton. Pharmacokinetics and pharmacodynamics. *Goodman and Gilman's the pharmacologic basis of therapeutics, 11th Ed. New York: McGraw-Hill*, pages 1–52, 2006.
- [54] O. Calin. *Deep learning architectures*. Springer, 2020.
- [55] J. Cardoso-Silva, L. G. Papageorgiou, and S. Tsoka. Network-based piecewise linear regression for qsar modelling. *Journal of Computer-Aided Molecular Design*, 33:831–844, 2019.
- [56] E. Carnesecchi, G. Raitano, A. Gamba, E. Benfenati, and A. Roncaglioni. Evaluation of non-commercial models for genotoxicity and carcinogenicity in the assessment of efsa's databases. *SAR QSAR Environ. Res.*, 31(1):33–48, 2020.
- [57] D. B. Carr, R. J. Littlefield, W. Nicholson, and J. Littlefield. Scatterplot matrix techniques for large n. *Journal of the American Statistical Association*, 82(398):424–436, 1987. doi: 10.2307/2289444.
- [58] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8):832, 2019.
- [59] A. Cassano, G. Raitano, E. Mombelli, A. Fernández, J. Cester, A. Roncaglioni, and E. Benfenati. Evaluation of qsar models for the prediction of ames genotoxicity: a retrospective exercise on the chemical substances registered under the eu reach regulation. *J. Environ. Sci. Health, Part C: Toxicol. Carcinog.*, 32(3):273–298, 2014.
- [60] L. Castelo-Soccio, H. Kim, M. Gadina, P. L. Schwartzberg, A. Laurence, and J. J. O'Shea. Protein kinases: drug targets for immunological disorders. *Nature Reviews Immunology*, pages 1–20, 2023.
- [61] A. Cayot, D. Laroche, A. Disson-Dautriche, A. Arbault, J.-F. Maillefert, and P. Ornetti. Cytochrome P450 interactions and clinical implication in rheumatology. *Clinical Rheumatology*, 33(9):1231–1238, Septiembre 2014. doi: 10.1007/s10067-014-2710-3.
- [62] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, and G. Pujadas. Molecular fingerprint similarity search in virtual screening. *Methods*, 71:58–63, 2015. doi: 10.1016/j.ymeth.2014.08.005.

- [63] L. Ceriani, E. Papa, S. Kovarich, R. Boethling, and P. Gramatica. Modeling ready biodegradability of fragrance materials. *Environmental Toxicology and Chemistry*, 34(6):1224–1231, Junio 2015. doi: 10.1002/etc.2926.
- [64] H. S. Chan, H. Shan, T. Dahoun, H. Vogel, and S. Yuan. Advancing drug discovery via artificial intelligence. *Trends in pharmacological sciences*, 40(8):592–604, 2019.
- [65] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
- [66] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke. The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6):1241–1250, Junio 2018. doi: 10.1016/J.DRUDIS.2018.01.039.
- [67] F. Cheng, Y. Yu, J. Shen, L. Yang, W. Li, G. Liu, P. W. Lee, and Y. Tang. Classification of Cytochrome P450 Inhibitors and Noninhibitors Using Combined Classifiers. *Journal of Chemical Information and Modeling*, 51(5):996–1011, Mayo 2011. doi: 10.1021/ci200028n.
- [68] A. Cherkasov, E. N. Muratov, D. Fourches, A. Varnek, I. I. Baskin, M. Cronin, J. Dearden, P. Gramatica, Y. C. Martin, R. Todeschini, et al. Qsar modeling: where have you been? where are you going to? *Journal of Medicinal Chemistry*, 57(12):4977–5010, 2014.
- [69] S. Chithrananda, G. Grand, and B. Ramsundar. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. *arXiv e-prints*, art. arXiv:2010.09885, Octubre 2020.
- [70] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv e-prints*, art. arXiv:1406.1078, Junio 2014.
- [71] F. Chollet et al. Keras, 2015. URL <https://github.com/fchollet/keras>. Accessed 2022-04-25.
- [72] C. S. Chu, J. D. Simpson, P. M. O’Neill, and N. G. Berry. Machine learning—predicting ames mutagenicity of small molecules. *J. Mol. Graphics Modell.*, 109:108011, 2021.
- [73] K. V. Chuang, L. M. Gunsalus, and M. J. Keiser. Learning molecular representations for medicinal chemistry: Miniperspective. *Journal of Medicinal Chemistry*, 63(16):8705–8722, 2020.

- [74] A. A. Ciociola, L. B. Cohen, P. Kulkarni, C. Kefalas, A. Buchman, C. Burke, T. Cain, J. Connor, E. D. Ehrenpreis, J. Fang, et al. How drugs are developed and approved by the fda: current process and future directions. *Official journal of the American College of Gastroenterology—ACG*, 109(5):620–623, 2014.
- [75] D. E. Clark and D. R. Westhead. Evolutionary algorithms in computer-aided molecular design. *Journal of Computer-Aided Molecular Design*, 10:337–358, 1996.
- [76] R. Concu and M. N. D. S. Cordeiro. On the relevance of feature selection algorithms while developing non-linear qsars. *Ecotoxicological QSARs*, pages 177–194, 2020.
- [77] M. A. Cox and T. F. Cox. Multidimensional scaling. In *Handbook of Data Visualization*, pages 315–347. Springer, 2008. doi: 10.1007/978-3-540-33037-0\_14.
- [78] W. Cui. Visual analytics: A comprehensive overview. *IEEE access*, 7:81555–81573, 2019.
- [79] A. Cutler, D. R. Cutler, and J. R. Stevens. Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175, 2012.
- [80] R. Cutura, S. Holzer, M. Aupetit, and M. Sedlmair. VisCoDeR: A tool for visually comparing dimensionality reduction algorithms. In *Esann*, 2018.
- [81] A. Dalke. Deepsmiles: An adaptation of smiles for use in machine-learning of chemical structures, 2018.
- [82] A. Das and P. Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.
- [83] Dassault Systèmes. Biovia Bioactivity Databases Datasheet, 2020. URL [https://www.simulations-plus.com/?s=MUT\\_Risk](https://www.simulations-plus.com/?s=MUT_Risk). Accessed 2022-04-08.
- [84] L. David, A. Thakkar, R. Mercado, and O. Engkvist. Molecular representations in ai-driven drug discovery: a review and practical guide. *Journal of Cheminformatics*, 12(1):1–22, 2020.
- [85] I. Daylight Chemical Information Systems. Smarts-a language for describing molecular patterns, 2007.
- [86] Daylight Chemical Information Systems, Inc. Daylight fingerprints, 2019. <http://www.daylight.com/>, online April 2020.

- [87] J. S. Delaney. Esol: estimating aqueous solubility directly from molecular structure. *Journal of chemical information and computer sciences*, 44(3):1000–1005, 2004.
- [88] R. V. Devi, S. S. Sathya, and M. S. Coumar. Evolutionary algorithms for de novo drug design – A survey. *Applied Soft Computing*, 27:543–552, Febrero 2015. doi: 10.1016/J.ASOC.2014.09.042.
- [89] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv e-prints*, art. arXiv:1810.04805, Octubre 2018.
- [90] J. A. DiMasi. Research and development costs of new drugs. *JAMA*, 324(5):517–517, 2020.
- [91] P. Domingos. Metacost: A general method for making classifiers cost-sensitive. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 155–164, 1999.
- [92] A. R. T. Donders, G. J. Van Der Heijden, T. Stijnen, and K. G. Moons. A gentle introduction to imputation of missing values. *Journal of clinical epidemiology*, 59(10):1087–1091, 2006.
- [93] S. Dong, P. Wang, and K. Abbas. A survey on deep learning and its applications. *Computer Science Review*, 40:100379, 2021.
- [94] F. K. Došilović, M. Brčić, and N. Hlupić. Explainable artificial intelligence: A survey. In *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 0210–0215. IEEE, 2018.
- [95] F. Drobnič, A. Kos, and M. Pustišek. On the interpretability of machine learning models and experimental feature selection in case of multicollinear data. *Electronics*, 9(5):761, 2020.
- [96] A. K. Dubey and V. Jain. Comparative study of convolution neural network’s relu and leaky-relu activation functions. In *Applications of Computing, Automation and Wireless Systems in Electrical Engineering: Proceedings of MARC 2018*, pages 873–880. Springer, 2019.
- [97] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri. Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 2022.
- [98] J. Dunn, L. Mingardi, and Y. D. Zhuo. Comparing interpretability and explainability for feature selection. *arXiv preprint arXiv:2105.05328*, 2021.

- [99] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of Chemical Information and Computer Sciences*, 42(6):1273–1280, 2002.
- [100] S. Ekins. The next era: deep learning in pharmaceutical research. *Pharmaceutical research*, 33(11):2594–2603, 2016.
- [101] S. Ekins, T. R. Lane, F. Urbina, and A. C. Puhl. In silico adme/tox comes of age: twenty years later. *Xenobiotica*, pages 1–7, 2023.
- [102] M. Eklund, U. Norinder, S. Boyer, and L. Carlsson. Choosing Feature Selection and Learning Algorithms in QSAR. *Journal of Chemical Information and Modeling*, 54(3):837–843, Marzo 2014. doi: 10.1021/ci400573c.
- [103] D. C. Elton, Z. Boukouvalas, M. D. Fuge, and P. W. Chung. Deep learning for molecular design—a review of the state of the art. *Molecular Systems Design & Engineering*, 4(4):828–849, 2019.
- [104] F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi, and M. Dehmer. An introductory review of deep learning for prediction models with big data. *Frontiers in Artificial Intelligence*, 3:4, 2020.
- [105] P. Ertl and A. Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1: 1–11, 2009.
- [106] J. Fang, R. Yang, L. Gao, S. Yang, X. Pang, C. Li, Y. He, A.-L. Liu, and G.-H. Du. Consensus models for CDK5 inhibitors in silico and their application to inhibitor discovery. *Molecular Diversity*, 19(1):149–162, Febrero 2015. doi: 10.1007/s11030-014-9561-3.
- [107] A. Fernández, R. Rallo, and F. Giralt. Prioritization of in silico models and molecular descriptors for the assessment of ready biodegradability. *Environmental Research*, 142:161–168, Octubre 2015. doi: 10.1016/J.ENVRES.2015.06.031.
- [108] J. Figueroa Barraza, E. López Droguett, and M. R. Martins. Towards interpretable deep learning: a feature selection framework for prognostics and health management using deep neural networks. *Sensors*, 21(17):5888, 2021.

- [109] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936. doi: 10.1111/j.1469-1809.1936.tb02137.x.
- [110] D. R. Flower. *Molecular informatics: sharpening drug design's cutting edge*. Royal Society of Chemistry, Cambridge, UK, 2002.
- [111] A. H. Foss, M. Markatou, and B. Ray. Distance metrics and clustering methods for mixed-type data. *International Statistical Review*, 87(1):80–109, 2019.
- [112] K. Furmanová, S. Gratzl, H. Stitz, T. Zichner, M. Jarešová, A. Lex, and M. Streit. Taggle: Combining overview and details in tabular data visualizations. *Information Visualization*, 19(2):114–136, 2020. doi: 10.1177/1473871619878085.
- [113] K. Gao, D. D. Nguyen, V. Sresht, A. M. Mathiowetz, M. Tu, and G.-W. Wei. Are 2d fingerprints still valuable for drug discovery? *Physical Chemistry Chemical Physics*, 22(16):8373–8390, 2020.
- [114] Z. Gao, X. Ji, G. Zhao, H. Wang, H. Zheng, G. Ke, and L. Zhang. Uni-qsar: an auto-ml tool for molecular property prediction. *arXiv preprint arXiv:2304.12239*, 2023.
- [115] S. García, J. Luengo, and F. Herrera. *Data preprocessing in data mining*, volume 72. Springer, 2015.
- [116] A. Gaulton, A. Hersey, M. Nowotka, A. P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. J. Bellis, E. Cibrián-Uhalte, et al. The ChEMBL database in 2017. *Nucleic Acids Research*, 45(D1):D945–d954, 2017. doi: 10.1093/nar/gkw1074.
- [117] N. Gehlenborg and B. Wong. Heat maps: heat maps are useful for visualizing multivariate data but must be applied properly. *Nature Methods*, 9(3):213–214, 2012.
- [118] F. Ghasemi, A. Mehridehnavi, A. Perez-Garrido, and H. Perez-Sanchez. Neural network and deep-learning algorithms used in qsar studies: merits and drawbacks. *Drug discovery today*, 23(10):1784–1790, 2018.
- [119] G. Gini, F. Zanoli, A. Gamba, G. Raitano, and E. Benfenati. Could deep learning in neural networks improve the qsar models? *SAR QSAR Environ. Res.*, 30(9):617–642, 2019.
- [120] A. Gisbrecht, A. Schulz, and B. Hammer. Parametric nonlinear dimensionality reduction using kernel t-SNE. *Neurocomputing*, 147:71–82, 2015. doi: 10.1016/j.neucom.2013.11.045.

- [121] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.
- [122] G. B. Goh, C. Siegel, A. Vishnu, N. O. Hodas, and N. Baker. Chemception: A Deep Neural Network with Minimal Chemistry Knowledge Matches the Performance of Expert-developed QSAR/QSPR Models. *arXiv e-prints*, art. arXiv:1706.06689, Junio 2017.
- [123] G. B. Goh, K. Sakloth, C. Siegel, A. Vishnu, and J. Pfandtner. Multimodal Deep Neural Networks using Both Engineered and Learned Representations for Biodegradability Prediction. *arXiv preprint arXiv:1808.04456*, Agosto 2018. doi: arXiv:1808.04456v2.
- [124] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018.
- [125] E. Gonzalez, S. Jain, P. Shah, N. Torimoto-Katori, A. Zakharov, D.-T. Nguyen, S. Sakamuru, R. Huang, M. Xia, R. S. Obach, et al. Development of robust quantitative structure-activity relationship models for cyp2c9, cyp2d6, and cyp3a4 catalysis and inhibition. *Drug Metabolism and Disposition*, 49(9):822–832, 2021.
- [126] M. Goodarzi, B. Dejaegher, and Y. Vander Heyden. Feature selection methods in QSAR studies. *Journal of AOAC International*, 95(3):636–51, 2012.
- [127] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.
- [128] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.
- [129] I. Goodfellow, Y. Bengio, and A. Courville. Regularization for deep learning. *Deep learning*, pages 216–261, 2016.
- [130] P. Gramatica. On the development and validation of qsar models. *Computational Toxicology: Volume II*, pages 499–526, 2013.
- [131] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. LineUp: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '13)*, 19(12):2277–2286, 2013. doi: 10.1109/tvcg.2013.173.



- [132] Greg Landrum. RDKit: Open-Source Cheminformatics Software, 2021. URL <http://www.rdkit.org>. Accessed 2021-08-05.
- [133] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. Towards conceptual compression. *arXiv preprint arXiv:1604.08772*, 2016.
- [134] F. Grisoni, V. Consonni, and R. Todeschini. *Impact of Molecular Descriptors on Computational Models*, pages 171–209. Springer New York, New York, NY, 2018. ISBN 978-1-4939-8639-2. doi: 10.1007/978-1-4939-8639-2\_5. URL [https://doi.org/10.1007/978-1-4939-8639-2\\_5](https://doi.org/10.1007/978-1-4939-8639-2_5).
- [135] F. Grisoni, M. Moret, R. Lingwood, and G. Schneider. Bidirectional molecule generation with recurrent neural networks. *Journal of chemical information and modeling*, 60(3):1175–1183, 2020.
- [136] R. Grosse. Lecture 15: Exploding and vanishing gradients. *University of Toronto Computer Science*, 2017.
- [137] W. Gu, L. Zhou, Z. Wang, C. Lin, J. Liu, H. Ge, and L. Shi. Ready biodegradability ring testing of 4-isopropylphenol in different laboratories for critical evaluation of a biodegradable reference substance. *Integrated Environmental Assessment and Management*, 17(3):562–572, 2021.
- [138] D. Guidolin, M. Marcoli, C. Tortorella, G. Maura, and L. F. Agnati. Receptor-receptor interactions as a widespread phenomenon: novel targets for drug development? *Frontiers in endocrinology*, 10:53, 2019.
- [139] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang. Xai—explainable artificial intelligence. *Science robotics*, 4(37):eaay7120, 2019.
- [140] A. Gupta, A. T. Müller, B. J. Huisman, J. A. Fuchs, P. Schneider, and G. Schneider. Generative recurrent networks for de novo drug design. *Molecular informatics*, 37(1-2):1700111, 2018.
- [141] P. A. Gutiérrez, M. Perez-Ortiz, J. Sanchez-Monedero, F. Fernandez-Navarro, and C. Hervas-Martinez. Ordinal regression methods: survey and experimental study. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):127–146, 2015.
- [142] M. Gütlein, A. Karwath, and S. Kramer. CheS-Mapper 2.0 for visual validation of (Q)SAR models. *Journal of Cheminformatics*, 6(1), 2014. ISSN 1758-2946. doi: 10.1186/s13321-014-0041-7.

- [143] M. Hahsler, M. Piekenbrock, and D. Doran. dbscan: Fast density-based clustering with r. *Journal of Statistical Software*, 91:1–30, 2019.
- [144] M. Hall and G. Holmes. Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1437–1447, Noviembre 2003. doi: 10.1109/TKDE.2003.1245283.
- [145] C. Hansch and T. Fujita.  $p$ - $\sigma$ - $\pi$  analysis. a method for the correlation of biological activity and chemical structure. *Journal of the American Chemical Society*, 86(8):1616–1626, 1964.
- [146] R. M. Hanson, J. Prilusky, Z. Renjian, T. Nakane, and J. L. Sussman. JSmol and the next-generation web-based representation of 3D molecular structure as applied to proteopedia. *Israel Journal of Chemistry*, 53(3-4):207–216, 2013. doi: 10.1002/ijch.201300024.
- [147] M. D. Hanwell, D. E. Curtis, D. C. Lonie, T. Vandermeersch, E. Zurek, and G. R. Hutchison. Avogadro: an advanced semantic chemical editor, visualization, and analysis platform. *Journal of Cheminformatics*, 4(1):17, 2012.
- [148] G. Harshvardhan, M. K. Gourisaria, M. Pandey, and S. S. Rautaray. A comprehensive survey and analysis of generative models in machine learning. *Computer Science Review*, 38:100285, 2020.
- [149] M. Hartenfeller and G. Schneider. De novo drug design. *Chemoinformatics and computational chemical biology*, pages 299–323, 2011.
- [150] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.
- [151] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034. IEEE, 2015.
- [152] S. Heller, A. McNaught, S. Stein, D. Tchekhovskoi, and I. Pletnev. Inchi-the worldwide chemical structure identifier standard. *Journal of cheminformatics*, 5(1):1–9, 2013.
- [153] J. Hemmerich and G. F. Ecker. In silico toxicology: From structure–activity relationships towards deep learning and adverse outcome pathways. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.*, 10(4):e1475, 2020.

- [154] S. Heo, U. Safder, and C. Yoo. Deep learning driven qsar model for environmental toxicology: effects of endocrine disrupting chemicals on human health. *Environmental Pollution*, 253:29–38, 2019.
- [155] K. Herrmann, A. Holzwarth, S. Rime, B. C. Fischer, and C. Kneuer. (q) sar tools for the prediction of mutagenic properties: Are they ready for application in pesticide regulation? *Pest Manage. Sci.*, 76(10):3316–3325, 2020.
- [156] R. P. Hertzberg and A. J. Pope. High-throughput screening: new technology for the 21st century. *Current Opinion in Chemical Biology*, 4(4):445–451, 2000. ISSN 1367-5931. doi: 10.1016/S1367-5931(00)00110-1.
- [157] G. Hessler, K.-H. Baringhaus, G. Hessler, and K.-H. Baringhaus. Artificial Intelligence in Drug Design. *Molecules*, 23(10):2520, Octubre 2018. doi: 10.3390/molecules23102520.
- [158] A. Heuillet, F. Couthouis, and N. Díaz-Rodríguez. Explainability in deep reinforcement learning. *Knowledge-Based Systems*, 214:106685, 2021.
- [159] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [160] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, Febrero 1970. doi: 10.1080/00401706.1970.10488634.
- [161] T. Hoffmann and M. Gastreich. The next level in chemical space navigation: going far beyond enumerable compound libraries. *Drug Discovery Today*, 24(5):1148–1156, 2019. ISSN 1359-6446. doi: 10.1016/j.drudis.2019.02.013.
- [162] M. Honma. An assessment of mutagenicity of chemical substances by (quantitative) structure–activity relationship. *Genes Environ.*, 42(1):1–13, 2020.
- [163] M. Honma, A. Kitazawa, A. Cayley, R. V. Williams, C. Barber, T. Hanser, R. Saiakhov, S. Chakravarti, G. J. Myatt, K. P. Cross, et al. Improvement of quantitative structure–activity relationship (qsar) tools for predicting ames mutagenicity: outcomes of the ames/qsar international challenge project. *Mutagenesis*, 34(1):3–16, 2019.
- [164] M.-P. Hosseini, S. Lu, K. Kamaraj, A. Slowikowski, and H. C. Venkatesh. Deep learning architectures. *Deep learning: concepts and architectures*, pages 1–24, 2020.

- [165] M. Hossin and M. N. Sulaiman. A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, 5(2):1, 2015.
- [166] S. Hu, P. Chen, P. Gu, and B. Wang. A deep learning-based chemical system for qsar prediction. *IEEE J. Biomed. Health Inform.*, 24(10):3020–3028, 2020.
- [167] B. Huang and O. A. Von Lilienfeld. Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity, 2016.
- [168] R. Huang, M. Xia, D. Nguyen, et al. Editorial: Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental toxicants and drugs. *front environ sci* 5: 3. *Front. Environ. Sci.*, 5(3):5, 2017.
- [169] J. Hughes, S. Rees, S. Kalindjian, and K. Philpott. Principles of early drug discovery. *British Journal of Pharmacology*, 162(6):1239–1249, Marzo 2011. doi: 10.1111/j.1476-5381.2010.01127.x.
- [170] T. B. Hughes, N. L. Dang, G. P. Miller, and S. J. Swamidass. Modeling reactivity to biological macromolecules with a deep multitask network. *ACS Cent. Sci.*, 2(8):529–537, 2016.
- [171] W. Humphrey, A. Dalke, and K. Schulten. VMD: Visual molecular dynamics. *Journal of Molecular Graphics*, 14(1):33–38, 1996. doi: 10.1016/0263-7855(96)00018-5.
- [172] Y. G. Ichihara, M. Okabe, K. Iga, Y. Tanaka, K. Musha, and K. Ito. Color universal design: the selection of four easily distinguishable colors for all color vision types. In *Color Imaging XIII: Processing, Hardcopy, and Applications*, volume 6807, page 680700. International Society for Optics and Photonics, 2008. doi: 10.1117/12.765420.
- [173] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015.
- [174] Istituto Superiore di Sanità. ISSTOX Chemical Toxicity Databases, 2019. URL <https://www.iss.it/isstox>. Accessed 2022-03-22.
- [175] E. Jacoby, A. Schuffenhauer, M. Popov, K. Azzaoui, E. Vangrevelinghe, J. Priestle, P. Ferrara, B. Faller, and P. Acklin. Molecular informatics as an enabling in silico technology platform for drug discovery. *Chimia*, 58(9):577–577, 2004.

- [176] S. Jaeger, S. Fulle, and S. Turk. Mol2vec: Unsupervised machine learning approach with chemical intuition. *Journal of Chemical Information and Modeling*, 58(1):27–35, 2018. doi: 10.1021/acs.jcim.7b00616.
- [177] W. James and C. Stein. Estimation with quadratic loss. In *Breakthroughs in statistics: Foundations and basic theory*, pages 443–460. Springer, 1992.
- [178] C. Janiesch, P. Zschech, and K. Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, 2021.
- [179] K. Janocha and W. M. Czarnecki. On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*, 2017.
- [180] A. P. Janssen, S. H. Grimm, R. H. Wijdeven, E. B. Lenselink, J. Neefjes, C. A. van Boeckel, G. J. van Westen, and M. van der Stelt. Drug discovery maps, a machine learning model that visualizes and predicts kinome-inhibitor interaction landscapes. *Journal of Chemical Information and Modeling*, 59(3):1221–1229, 2018. doi: 10.1021/acs.jcim.8b00640.
- [181] K. Jayasundara, A. Hollis, M. Krahn, M. Mamdani, J. S. Hoch, and P. Grootendorst. Estimating the clinical cost of drug development for orphan versus non-orphan drugs. *Orphanet journal of rare diseases*, 14:1–10, 2019.
- [182] D. Jiang, Z. Wu, C.-Y. Hsieh, G. Chen, B. Liao, Z. Wang, C. Shen, D. Cao, J. Wu, and T. Hou. Could graph neural networks learn better molecular representation for drug discovery? a comparison study of descriptor-based and graph-based models. *Journal of cheminformatics*, 13(1):1–23, 2021.
- [183] J. Jiménez-Luna, F. Grisoni, and G. Schneider. Drug discovery with explainable artificial intelligence. *Nature Machine Intelligence*, 2(10):573–584, 2020.
- [184] Jmol. Jmol: an open-source Java viewer for chemical structures in 3D, 2009. [www.jmol.org](http://www.jmol.org), online January 2020.
- [185] M. A. Johnson and G. M. Maggiora. *Concepts and applications of molecular similarity*. Wiley, 1990. doi: 10.1002/jcc.540130415.
- [186] G. Joshi, R. Walambe, and K. Kotecha. A review on explainability in multimodal deep neural nets. *IEEE Access*, 9:59800–59821, 2021.

- [187] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.
- [188] U. Kamath and J. Liu. *Explainable artificial intelligence: An introduction to interpretable machine learning*. Springer, 2021.
- [189] S. Kar, K. Roy, and J. Leszczynski. Applicability Domain: A Step Toward Confident Predictions and Decidability for QSAR Modeling. In *Methods in molecular biology (Clifton, N.J.)*, volume 1800, pages 141–169. Springer, 2018. doi: 10.1007/978-1-4939-7899-1\_6.
- [190] B. Karlik and A. V. Olgac. Performance analysis of various activation functions in generalized mlp architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*, 1(4):111–122, 2011.
- [191] T. Kasamatsu, A. Kitazawa, S. Tajima, M. Kaneko, K.-i. Sugiyama, M. Yamada, M. Yasui, K. Masumura, K. Horibata, and M. Honma. Development of a new quantitative structure–activity relationship model for predicting ames mutagenicity of food flavor chemicals using stardrop™ auto-modeller™. *Genes Environ.*, 43(1):1–17, 2021.
- [192] J. Kazius, R. McGuire, and R. Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *J. Med. Chem.*, 48(1):312–320, 2005.
- [193] D. B. Kell, S. Samanta, and N. Swainston. Deep learning and generative methods in cheminformatics and chemical biology: navigating small molecule space intelligently. *Biochemical Journal*, 477(23):4559–4580, 2020.
- [194] S. Khademi, X. Shi, T. Mager, R. Siebes, C. Hein, V. de Boer, and J. van Gemert. Sight-Seeing in the Eyes of Deep Neural Networks. In *2018 IEEE 14th International Conference on e-Science (e-Science)*, pages 407–408. IEEE, Octubre 2018. ISBN 978-1-5386-9156-4. doi: 10.1109/eScience.2018.00125.
- [195] Khronos. WebGL: OpenGL ES for the Web. <https://www.khronos.org/webgl/>, online April 2020, 2011.
- [196] S. Kim, P. A. Thiessen, E. E. Bolton, J. Chen, G. Fu, A. Gindulyte, L. Han, J. He, S. He, B. A. Shoemaker, et al. Pubchem substance and compound databases. *Nucleic Acids Research*, 44 (D1):D1202–d1213, 2016. doi: 10.1093/nar/gkv951.

- [197] T. B. Kimber, Y. Chen, and A. Volkamer. Deep learning in virtual screening: recent applications and developments. *International journal of molecular sciences*, 22(9):4435, 2021.
- [198] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, art. arXiv:1412.6980, Diciembre 2014.
- [199] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [200] W. Klingspohn, M. Mathea, A. ter Laak, N. Heinrich, and K. Baumann. Efficiency of different measures for defining the applicability domain of classification models. *Journal of Cheminformatics*, 9(1):44, Diciembre 2017. doi: 10.1186/s13321-017-0230-2.
- [201] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990. doi: 10.1109/5.58325.
- [202] S. Korkmaz. Deep learning-based imbalanced data classification for drug discovery. *Journal of chemical information and modeling*, 60(9):4180–4190, 2020.
- [203] A. Koutsoukas, K. J. Monaghan, X. Li, and J. Huan. Deep-learning: investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data. *Journal of Cheminformatics*, 9(1):42, Diciembre 2017. doi: 10.1186/s13321-017-0226-y.
- [204] B. Kozlíková, M. Krone, M. Falk, N. Lindow, M. Baaden, D. Baum, I. Viola, J. Parulek, and H.-C. Hege. Visualization of biomolecular structures: State of the art revisited. *Computer Graphics Forum*, 36(8):178–204, 2016. doi: 10.1111/cgf.13072.
- [205] M. A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37(2):233–243, 1991. doi: 10.1002/aic.690370209.
- [206] O. Kramer and O. Kramer. K-nearest neighbors. *Dimensionality reduction with unsupervised nearest neighbors*, pages 13–23, 2013.
- [207] M. Krenn, F. Häse, A. Nigam, P. Friederich, and A. Aspuru-Guzik. Selfies: a robust representation of semantically constrained graphs with an example application in chemistry. *arXiv preprint arXiv:1905.13741*, 1(3), 2019.
- [208] E. Krieger and G. Vriend. YASARA View—molecular graphics for all devices—from smartphones to workstations. *Bioinformatics*, 30(20):2981–2982, 2014. doi: 10.1093/bioinformatics/btu426.

- [209] M. Kuhn, K. Johnson, et al. *Applied predictive modeling*, volume 26. Springer, 2013.
- [210] D. Kuzminykh, D. Polykovskiy, A. Kadurin, A. Zhebrak, I. Baskov, S. Nikolenko, R. Shayakhmetov, and A. Zhavoronkov. 3d molecular representations based on the wave transform for convolutional neural networks. *Molecular Pharmaceutics*, 15(10):4378–4385, 2018.
- [211] G. Landrum. Rdkit: open-source cheminformatics <http://www.rdkit.org>, 2016.
- [212] R. A. Laskowski and M. B. Swindells. LigPlot+: Multiple ligand-protein interaction diagrams for drug discovery. *Journal of Chemical Information and Modeling*, 51(10):2778–2786, 2011. doi: 10.1021/ci200227u.
- [213] A. Lavecchia. Machine-learning approaches in drug discovery: methods and applications. *Drug Discovery Today*, 20(3):318–331, Marzo 2015. doi: 10.1016/J.DRUDIS.2014.10.012.
- [214] A. Lavecchia. Deep learning in drug discovery: opportunities, challenges and future prospects. *Drug discovery today*, 24(10):2017–2032, 2019.
- [215] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. In *Neural Information Processing Systems (NIPS)*, 1989.
- [216] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [217] J. A. Lee and M. Verleysen. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing*, 72(7-9):1431–1443, 2009.
- [218] T. Lei, Y. Li, Y. Song, D. Li, H. Sun, and T. Hou. ADMET evaluation in drug discovery: 15. Accurate prediction of rat oral acute toxicity using relevance vector machine and consensus modeling. *Journal of Cheminformatics*, 8(1):6, Diciembre 2016. doi: 10.1186/s13321-016-0117-7.
- [219] E. B. Lenselink, N. ten Dijke, B. Bongers, G. Papadatos, H. W. T. van Vlijmen, W. Kowalczyk, A. P. IJzerman, and G. J. P. van Westen. Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set. *Journal of Cheminformatics*, 9(1):45, Diciembre 2017. doi: 10.1186/s13321-017-0232-0.
- [220] R. Lewis, R. Guha, T. Korcsmaros, and A. Bender. Synergy maps: exploring compound combinations using network-based visualization. *Journal of Cheminformatics*, 7(1), 2015. doi: 10.1186/s13321-015-0090-6.



- [221] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45, 2017.
- [222] J. Li, D. Luo, T. Wen, Q. Liu, and Z. Mo. Representative feature selection of molecular descriptors in qsar modeling. *Journal of Molecular Structure*, 1244:131249, 2021.
- [223] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.
- [224] H. Lin, H. Ding, F.-B. Guo, and J. Huang. Prediction of subcellular location of mycobacterial protein using feature selection techniques. *Molecular Diversity*, 14(4):667–671, Noviembre 2010. doi: 10.1007/s11030-009-9205-1.
- [225] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18, 2020.
- [226] E. Lionta, G. Spyrou, D. K. Vassilatis, and Z. Cournia. Structure-based virtual screening for drug discovery: Principles, applications and recent advances. *Current Topics in Medicinal Chemistry*, 14(16):1923–1938, 2014. doi: 10.2174/1568026614666140929124445.
- [227] C. A. Lipinski. Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discovery Today: Technologies*, 1(4):337–341, 2004. doi: 10.1016/j.ddtec.2004.11.007.
- [228] R. Liu, H. Wang, K. P. Glover, M. G. Feasel, and A. Wallqvist. Dissecting Machine-Learning Prediction of Molecular Activity: Is an Applicability Domain Needed for Quantitative Structure–Activity Relationship Models Based on Deep Neural Networks? *Journal of Chemical Information and Modeling*, page acs.jcim.8b00348, Noviembre 2018. doi: 10.1021/acs.jcim.8b00348.
- [229] S. Liu, M. Furkan Demirel, and Y. Liang. N-Gram Graph: Simple Unsupervised Representation for Graphs, with Applications to Molecules. *arXiv e-prints*, art. arXiv:1806.09206, Junio 2018.
- [230] E. S. Lubana, R. Dick, and H. Tanaka. Beyond batchnorm: towards a unified understanding of normalization in deep learning. *Advances in Neural Information Processing Systems*, 34: 4778–4791, 2021.
- [231] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik. Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships. *Journal of Chemical Information and Modeling*, 55(2):263–274, Febrero 2015. doi: 10.1021/ci500747n.

- [232] S. J. Y. Macalino, V. Gosu, S. Hong, and S. Choi. Role of computer-aided drug design in modern drug discovery. *Archives of Pharmacal Research*, 38(9):1686–1701, Septiembre 2015. doi: 10.1007/s12272-015-0640-5.
- [233] R. Macarron, M. N. Banks, D. Bojanic, D. J. Burns, D. A. Cirovic, T. Garyantes, D. V. Green, R. P. Hertzberg, W. P. Janzen, J. W. Paslay, et al. Impact of high-throughput screening in biomedical research. *Nature Reviews Drug Discovery*, 10(3):188–195, 2011. doi: 10.1038/nrd3368.
- [234] G. M. Maggiora and V. Shanmugasundaram. Molecular similarity measures. In *Chemoinformatics*, pages 1–50. Springer, 2004. doi: 10.1385/1-59259-802-1:001.
- [235] M. S. Mahmud, J. Z. Huang, S. Salloum, T. Z. Emara, and K. Sadatdiyev. A survey of data partitioning and sampling methods to support big data analysis. *Big Data Mining and Analytics*, 3(2):85–101, 2020.
- [236] E. H. B. Maia, L. C. Assis, T. A. De Oliveira, A. M. Da Silva, and A. G. Taranto. Structure-based virtual screening: from classical to artificial intelligence. *Frontiers in chemistry*, 8:343, 2020.
- [237] K.-K. Mak and M. R. Pichika. Artificial intelligence in drug development: present status and future prospects. *Drug discovery today*, 24(3):773–780, 2019.
- [238] M. A. Maloof. On machine learning, roc analysis, and statistical tests of significance. In *2002 international conference on pattern recognition*, volume 2, pages 204–207. IEEE, 2002.
- [239] A. Mammone, M. Turchi, and N. Cristianini. Support vector machines. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(3):283–289, 2009.
- [240] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008. ISBN 9780521865715.
- [241] K. Mansouri, T. Ringsted, D. Ballabio, R. Todeschini, and V. Consonni. Quantitative Structure–Activity Relationship Models for Ready Biodegradability of Chemicals. *Journal of Chemical Information and Modeling*, 53(4):867–878, Abril 2013. doi: 10.1021/ci4000213.
- [242] R. L. Marchese Robinson, A. Palczewska, J. Palczewski, and N. Kidley. Comparison of the Predictive Performance and Interpretability of Random Forest and Linear Models on

- Benchmark Data Sets. *Journal of Chemical Information and Modeling*, 57(8):1773–1792, Agosto 2017. doi: 10.1021/acs.jcim.6b00753.
- [243] R. Marcinkevičs and J. E. Vogt. Interpretability and explainability: A machine learning zoo mini-tour. *arXiv preprint arXiv:2012.01805*, 2020.
- [244] E. MarÉchal. Measuring bioactivity: Ki, ic50 and ec50. *Chemogenomics and Chemical Genetics: A User's Introduction for Biologists, Chemists and Informaticians*, pages 55–65, 2011.
- [245] D. M. Maron and B. N. Ames. Revised methods for the salmonella mutagenicity test. *Mutat. Res., Environ. Mutagen. Relat. Subj.*, 113(3-4):173–215, 1983.
- [246] A. E. Márquez-Chamorro, G. Asencio-Cortés, C. E. Santiesteban-Toca, and J. S. Aguilar-Ruiz. Soft computing methods for the prediction of protein tertiary structures: A survey. *Applied Soft Computing*, 35:398–410, Octubre 2015. doi: 10.1016/J.ASOC.2015.06.024.
- [247] M. J. Martínez, I. Ponzoni, M. F. Díaz, G. E. Vazquez, and A. J. Soto. Visual analytics in cheminformatics: user-supervised descriptor selection for QSAR methods. *Journal of Cheminformatics*, 7(1):39, Diciembre 2015. doi: 10.1186/s13321-015-0092-4.
- [248] M. J. Martínez, M. V. Sabando, A. J. Soto, C. Roca, C. Requena-Triguero, N. E. Campillo, J. A. Páez, and I. Ponzoni. Multitask deep neural networks for ames mutagenicity prediction. *Journal of Chemical Information and Modeling*, 62(24):6342–6351, 2022. doi: 10.1021/acs.jcim.2c00532. URL <https://doi.org/10.1021/acs.jcim.2c00532>. PMID: 36066065.
- [249] K. Martinez-Mayorga, A. Madariaga-Mazon, J. L. Medina-Franco, and G. Maggiora. The impact of chemoinformatics on drug discovery in the pharmaceutical industry. *Expert opinion on drug discovery*, 15(3):293–306, 2020.
- [250] A. Mathew, P. Amudha, and S. Sivakumari. Deep learning techniques: an overview. *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pages 599–608, 2021.
- [251] M. Matveieva and P. Polishchuk. Benchmarks for interpretation of qsar models. *Journal of cheminformatics*, 13(1):41, 2021.
- [252] A. Mauri, V. Consonni, R. Todeschini, et al. Molecular descriptors. In *Handbook of computational chemistry*, pages 2065–2093. Springer International Publishing, 2017.

- [253] A. Mayr, G. Klambauer, T. Unterthiner, and S. Hochreiter. Deeptox: toxicity prediction using deep learning. *Front. Environ. Sci.*, 3:80, 2016.
- [254] W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5(4):115–133, 1943.
- [255] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- [256] A. K. Menon, A. S. Rawat, S. J. Reddi, and S. Kumar. Can gradient clipping mitigate label noise? In *International Conference on Learning Representations*, 2019.
- [257] A. Micheli and M. Podda. Deep learning in cheminformatics. In *Deep Learning in Biology and Medicine*, pages 157–195. World Scientific, 2022.
- [258] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv e-prints*, art. arXiv:1301.3781, Enero 2013.
- [259] M. R. Min, H. Guo, and D. Shen. Parametric t-distributed stochastic exemplar-centered embedding. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 477–493. Springer, 2018. doi: 10.1007/978-3-030-10925-7\_29.
- [260] D. L. Mobley and J. P. Guthrie. Freesolv: a database of experimental and calculated hydration free energies, with input files. *Journal of computer-aided molecular design*, 28(7):711–720, 2014.
- [261] G. Montavon, W. Samek, and K.-R. Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, Febrero 2018. doi: 10.1016/J.DSP.2017.10.011.
- [262] J. Moody, S. Hanson, A. Krogh, and J. A. Hertz. A simple weight decay can improve generalization. *Advances in Neural Information Processing Systems*, 4(1995):950–957, 1995.
- [263] R. Moradi, R. Berangi, and B. Minaei. A survey of regularization strategies for deep models. *Artificial Intelligence Review*, 53:3947–3986, 2020.
- [264] E. F. Morales and H. J. Escalante. A brief introduction to supervised, unsupervised, and reinforcement learning. In *Biosignal processing and classification using computational learning and intelligence*, pages 111–129. Elsevier, 2022.

- [265] A. Mordvintsev, C. Olah, and M. Tyka. Inceptionism: Going deeper into neural networks. *Google Research*, 2015. <https://blog.research.google/2015/06/inceptionism-going-deeper-into-neural.html>.
- [266] L. F. Moreno. Understanding fischer projection and angular line representation conversion. *Journal of Chemical Education*, 89(1):175–176, 2012.
- [267] H. L. Morgan. The generation of a unique machine description for chemical structures—a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.
- [268] H. Moriwaki, Y.-S. Tian, N. Kawashita, and T. Takagi. Mordred: a molecular descriptor calculator. *Journal of Cheminformatics*, 10(1):4, 2018.
- [269] G. M. Morris, R. Huey, W. Lindstrom, M. F. Sanner, R. K. Belew, D. S. Goodsell, and A. J. Olson. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry*, 30(16):2785–2791, 2009.
- [270] K. Mortelmans and E. Zeiger. The ames salmonella/microsome mutagenicity assay. *Mutation research/fundamental and molecular mechanisms of mutagenesis*, 455(1-2):29–60, 2000.
- [271] V. D. Mouchlis, A. Afantitis, A. Serra, M. Fratello, A. G. Papadiamantis, V. Aidinis, I. Lynch, D. Greco, and G. Melagraki. Advances in de novo drug design: from conventional to machine learning methods. *International journal of molecular sciences*, 22(4):1676, 2021.
- [272] S. S. Mousavi, M. Schukat, and E. Howley. Deep reinforcement learning: an overview. In *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016: Volume 2*, pages 426–440. Springer, 2018.
- [273] E. N. Muratov, J. Bajorath, R. P. Sheridan, I. V. Tetko, D. Filimonov, V. Poroikov, T. I. Oprea, I. I. Baskin, A. Varnek, A. Roitberg, et al. Qsar without borders. *Chemical Society Reviews*, 49(11):3525–3564, 2020.
- [274] F. Murtagh and P. Contreras. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97, 2012.
- [275] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10)*, pages 807–814. Omnipress, 2010.

- [276] M. V. Narkhede, P. P. Bartakke, and M. S. Sutaone. A review on weight initialization strategies for neural networks. *Artificial intelligence review*, 55(1):291–322, 2022.
- [277] H. Nasrallah. Atypical antipsychotic-induced metabolic side effects: insights from receptor-binding profiles. *Molecular Psychiatry*, 13(1):27–35, 2008. doi: 10.1038/sj.mp.4002066.
- [278] J. J. Naveja and J. L. Medina-Franco. Finding constellations in chemical space through core analysis. *Frontiers in Chemistry*, 7, 2019. doi: 10.3389/fchem.2019.00510.
- [279] S. Nembri, F. Grisoni, V. Consonni, and R. Todeschini. In Silico Prediction of Cytochrome P450-Drug Interaction: QSARs for CYP3A4 and CYP2C9. *International Journal of Molecular Sciences*, 17(6):914, Junio 2016. doi: 10.3390/ijms17060914.
- [280] I. E. Nielsen, D. Dera, G. Rasool, R. P. Ramachandran, and N. C. Bouaynaya. Robust explainability: A tutorial on gradient-based attribution methods for deep neural networks. *IEEE Signal Processing Magazine*, 39(4):73–84, 2022.
- [281] D. Numeroso and D. Bacciu. Meg: Generating molecular counterfactual explanations for deep graph networks. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2021.
- [282] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018.
- [283] N. M. O’Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison. Open babel: An open chemical toolbox. *Journal of Cheminformatics*, 3(1), 2011. doi: 10.1186/1758-2946-3-33.
- [284] T. N. OECD. 471: bacterial reverse mutation test. *OECD Guidelines for the Testing of Chemicals, Section*, 4, 1997.
- [285] M. G. Omran, A. P. Engelbrecht, and A. Salman. An overview of clustering methods. *Intelligent Data Analysis*, 11(6):583–605, 2007.
- [286] R. OpenAI. Gpt-4 technical report. *arXiv*, pages 2303–08774, 2023.
- [287] A. Oskooei, J. Born, M. Manica, V. Subramanian, J. Sáez-Rodríguez, and M. Rodríguez Martínez. Paccmann: Prediction of anticancer compound sensitivity with multi-modal attention-based neural networks. arxiv doi: <https://arxiv.org/abs/1811.06802>, 14 july 2019. In *Workshop on Machine Learning for Molecules and Materials in NeurIPS*, 2018.

- [288] A. Oussidi and A. Elhassouny. Deep generative models: Survey. In *2018 International conference on intelligent systems and computer vision (ISCV)*, pages 1–8. IEEE, 2018.
- [289] R. Özçelik, H. Öztürk, A. Özgür, and E. Ozkirimli. Chemboost: A chemical language based approach for protein–ligand binding affinity prediction. *Molecular Informatics*, 40(5):2000212, 2021.
- [290] H. Öztürk, E. Ozkirimli, and A. Özgür. A novel methodology on distributed representations of proteins using their interacting ligands. *Bioinformatics*, 34(13):i295–i303, 2018.
- [291] W. Pan, L. Feng, C.-P. Sun, X.-G. Tian, C. Shi, C. Wang, X. Lv, Y. Wang, S.-S. Huang, B.-J. Zhang, et al. Tetracyclic triterpenoids as inhibitors of cytochrome p450 3a4 and their quantitative structure activity relationship analysis. *Arabian Journal of Chemistry*, 16(10):105156, 2023.
- [292] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning (ICML)*, 2013.
- [293] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [294] J. Peltonen and Z. Lin. Information retrieval approach to meta-visualization. *Machine Learning*, 99(2):189–229, 2015. doi: 10.1007/s10994-014-5464-x.
- [295] E. Pérez Santín, R. Rodríguez Solana, M. González García, M. D. M. García Suárez, G. D. Blanco Díaz, M. D. Cima Cabal, J. M. Moreno Rojas, and J. I. López Sánchez. Toxicity prediction based on artificial intelligence: A multidisciplinary overview. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.*, 11(5):e1516, 2021.
- [296] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng, and T. E. Ferrin. UCSF Chimera—A visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25(13):1605–1612, 2004. doi: 10.1002/jcc.20084.
- [297] W. H. L. Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli. Autoencoders. In *Machine learning*, pages 193–208. Elsevier, 2020.

- [298] P. Polishchuk. Interpretation of Quantitative Structure–Activity Relationship Models: Past, Present, and Future. *Journal of Chemical Information and Modeling*, 57(11):2618–2639, Noviembre 2017. doi: 10.1021/acs.jcim.7b00274.
- [299] D. Polykovskiy, A. Zhebrak, B. Sanchez-Lengeling, S. Golovanov, O. Tatanov, S. Belyaev, R. Kurbanov, A. Artamonov, V. Aladinskiy, M. Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.
- [300] I. Ponzoni, V. Sebastián-Pérez, C. Requena-Triguero, C. Roca, M. J. Martínez, F. Cravero, M. F. Díaz, J. A. Páez, R. G. Arrayás, J. Adrio, and N. E. Campillo. Hybridizing Feature Selection and Feature Learning Approaches in QSAR Modeling for Drug Discovery. *Scientific reports*, 7(1):2403, 2017. doi: 10.1038/s41598-017-02114-3.
- [301] M. Popova, O. Isayev, and A. Tropsha. Deep reinforcement learning for de novo drug design. *Science Advances*, 4(7):eaap7885, 2018.
- [302] D. Probst and J.-L. Reymond. Exploring drugbank in virtual reality chemical space. *Journal of Chemical Information and Modeling*, 58(9):1731–1735, 2018. doi: 10.1021/acs.jcim.8b00402.
- [303] D. Probst and J.-L. Reymond. Visualization of very large high-dimensional data sets as minimum spanning trees. *Journal of Cheminformatics*, 12(1):1–13, 2020. doi: 10.1186/s13321-020-0416-x.
- [304] J. Qi, J. Du, S. M. Siniscalchi, X. Ma, and C.-H. Lee. On mean absolute error for deep neural network based vector-to-vector regression. *IEEE Signal Processing Letters*, 27:1485–1489, 2020.
- [305] A. Rácz, D. Bajusz, and K. Héberger. Multi-level comparison of machine learning classifiers and their performance metrics. *Molecules*, 24(15):2811, 2019.
- [306] A. Rakhimbekova, T. I. Madzhidov, R. I. Nugmanov, T. R. Gimadiev, I. I. Baskin, and A. Varnek. Comprehensive analysis of applicability domains of qspr models for chemical reactions. *International Journal of Molecular Sciences*, 21(15):5542, 2020.
- [307] P. E. Rauber, A. X. Falcão, and A. C. Telea. Visualizing time-dependent data using dynamic t-SNE. In E. Bertini, N. Elmqvist, and T. Wischgoll, editors, *EuroVis 2016 - Short Papers*. The Eurographics Association, 2016. ISBN 978-3-03868-014-7. doi: 10.2312/eurovisshort.20161164.



- [308] N. Rego and D. Koes. 3Dmol.js: molecular visualization with WebGL. *Bioinformatics*, 31(8): 1322–1324, 2015. doi: 10.1093/bioinformatics/btu829.
- [309] Z. Reitermanova et al. Data splitting. In *WDS*, volume 10, pages 31–36. Matfyzpress Prague, 2010.
- [310] M. T. Ribeiro, S. Singh, and C. Guestrin. "why should i trust you?". In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 1135–1144, New York, New York, USA, 2016. ACM Press. ISBN 9781450342322. doi: 10.1145/2939672.2939778.
- [311] D. Rogers and M. Hahn. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, Mayo 2010. doi: 10.1021/ci100050t.
- [312] L. Rokach and O. Maimon. Clustering methods. *Data mining and knowledge discovery handbook*, pages 321–352, 2005.
- [313] L. Rokach and O. Maimon. Decision trees. *Data mining and knowledge discovery handbook*, pages 165–192, 2005.
- [314] A. Ronacher. Flask: a lightweight WSGI web application framework, 2009. <https://palletsprojects.com/p/flask/>, online April 2020.
- [315] D. Rosner and G. Markowitz. Persistent pollutants: A brief history of the discovery of the widespread toxicity of chlorinated hydrocarbons. *Environmental Research*, 120:126–133, Enero 2013. doi: 10.1016/J.ENVRES.2012.08.011.
- [316] K. Roy. Advances in qsar modeling. *Applications in Pharmaceutical, Chemical, Food, Agricultural and Environmental Sciences; Springer: Cham, Switzerland*, 555:39, 2017.
- [317] K. Roy, S. Kar, and P. Ambure. On a simple approach for determining applicability domain of QSAR models. *Chemometrics and Intelligent Laboratory Systems*, 145:22–29, Julio 2015. doi: 10.1016/j.chemolab.2015.04.013.
- [318] K. Roy, P. Ambure, and R. B. Aher. How important is to detect systematic error in predictions and understand statistical applicability domain of QSAR models? *Chemometrics and Intelligent Laboratory Systems*, 162:44–54, Marzo 2017. doi: 10.1016/J.CHEMOLAB.2017.01.010.
- [319] S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

- [320] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.
- [321] D. P. Russo, K. M. Zorn, A. M. Clark, H. Zhu, and S. Ekins. Comparing Multiple Machine Learning Algorithms and Metrics for Estrogen Receptor Binding Prediction. *Molecular Pharmaceutics*, 15(10):4361–4370, Octubre 2018. doi: 10.1021/acs.molpharmaceut.8b00546.
- [322] M. V. Sabando, I. Ponzoni, and A. J. Soto. Neural-based approaches to overcome feature selection and applicability domain in drug-related property prediction. *Applied Soft Computing*, 85:105777, 2019. ISSN 1568-4946. doi: <https://doi.org/10.1016/j.asoc.2019.105777>. URL <https://www.sciencedirect.com/science/article/pii/S1568494619305587>.
- [323] M. V. Sabando, P. Ulbrich, M. Selzer, J. Byška, J. Mičan, I. Ponzoni, A. J. Soto, M. L. Ganuza, and B. Kozlíková. Chemva: Interactive visual analysis of chemical compound similarity in virtual screening. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):891–901, 2021. doi: 10.1109/TVCG.2020.3030438.
- [324] M. V. Sabando, I. Ponzoni, E. E. Milios, and A. J. Soto. Using molecular embeddings in qsar modeling: does it make a difference? *Briefings in bioinformatics*, 23(1):bbab365, 2022. ISSN 1477-4054. doi: 10.1093/bib/bbab365. URL <https://doi.org/10.1093/bib/bbab365>.
- [325] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *Advances in neural information processing systems*, 29, 2016.
- [326] T. Sander, J. Freyss, M. von Korff, and C. Rufener. DataWarrior: An open-source program for chemistry aware data visualization and analysis. *Journal of Chemical Information and Modeling*, 55(2):460–473, 2015. doi: 10.1021/ci500588j.
- [327] K. Sankaran and S. P. Holmes. Generative models: An interdisciplinary perspective. *Annual Review of Statistics and Its Application*, 10:325–352, 2023.
- [328] M. F. Sanner, A. J. Olson, and J.-C. Spohner. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305–320, 1996.
- [329] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

- [330] I. H. Sarker. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, 2(6):420, 2021.
- [331] R. Sayle. Reading and writing molecular file formats for data exchange of small molecules, biopolymers and reactions. In *ABSTRACTS OF PAPERS OF THE AMERICAN CHEMICAL SOCIETY*, volume 246. AMER CHEMICAL SOC 1155 16TH ST, NW, WASHINGTON, DC 20036 USA, 2013.
- [332] G. Schneider. Virtual screening: an endless staircase? *Nature Reviews Drug Discovery*, 9(4):273–276, 2010.
- [333] B. Schölkopf, K. Tsuda, and J.-P. Vert. *Kernel methods in computational biology*. MIT press, 2004.
- [334] L. Schrödinger and W. DeLano. Pymol, 2020. URL <http://www.pymol.org/pymol>.
- [335] Schrödinger, LLC. LigPrep, 2022. Schrödinger Release 2022-1. Accessed 2022-04-22.
- [336] Schrödinger, LLC. Maestro, 2022. Schrödinger Release 2022-1. Accessed 2022-04-22.
- [337] T. W. Schultz, M. T. Cronin, J. D. Walker, and A. O. Aptula. Quantitative structure–activity relationships (qsars) in toxicology: a historical perspective. *Journal of Molecular structure: THEOCHEM*, 622(1-2):1–22, 2003.
- [338] P. Schwaller, T. Gaudin, D. Lanyi, C. Bekas, and T. Laino. “found in translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chemical Science*, 9(28):6091–6098, 2018.
- [339] P. Sedgwick. Pearson’s correlation coefficient. *Bmj*, 345, 2012.
- [340] M. Sedlmair, T. Munzner, and M. Tory. Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2634–2643, 2013. doi: 10.1109/tvcg.2013.153.
- [341] M. H. Segler, T. Kogej, C. Tyrchan, and M. P. Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Central Science*, 4(1):120–131, 2018.
- [342] C. Selassie, S. Mekapati, and R. Verma. Qsar: then and now. *Current Topics in Medicinal Chemistry*, 2(12):1357–1379, 2002.

- [343] A. Seth and K. Roy. Qsar modeling of algal low level toxicity values of different phenol and aniline derivatives using 2d descriptors. *Aquatic Toxicology*, 228:105627, 2020.
- [344] P. Shah, A. Zakharov, R. S. Obach, A. Simeonov, C. Hop, D.-T. Guyen, E. Gonzalez, H. Sun, and X. Xu. Development of a multitask deep learning QSAR model using data from individual cytochrome P450 isozymes. *Drug Metabolism and Pharmacokinetics*, 33(1):S35–S36, Enero 2018. doi: 10.1016/J.DMPK.2017.11.131.
- [345] B. Shaker, S. Ahmad, J. Lee, C. Jung, and D. Na. In silico methods and tools for drug discovery. *Computers in biology and medicine*, 137:104851, 2021.
- [346] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [347] S. Sharma. Activation functions in neural networks. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>, 2017.
- [348] T. Shi, Y. Yang, S. Huang, L. Chen, Z. Kuang, Y. Heng, and H. Mei. Molecular image-based convolutional neural network for the prediction of admet properties. *Chemometrics and Intelligent Laboratory Systems*, 194:103853, 2019.
- [349] B. Shin, S. Park, K. Kang, and J. C. Ho. Self-attention based molecule representation for predicting drug-target interaction. In *Machine Learning for Healthcare Conference*, pages 230–248. PMLR, 2019.
- [350] R. Shwartz-Ziv and N. Tishby. Opening the Black Box of Deep Neural Networks via Information. *arXiv preprint arXiv:1703.00810*, Marzo 2017.
- [351] S. Siafis, D. Tzachanis, M. Samara, and G. Papazisis. Antipsychotic drugs: from receptor-binding profiles to metabolic side effects. *Current neuropharmacology*, 16(8):1210–1223, 2018. doi: 10.2174/1570159x15666170630163616.
- [352] R. S. Simoes, V. G. Maltarollo, P. R. Oliveira, and K. M. Honorio. Transfer and multi-task learning in qsar modeling: advances and challenges. *Frontiers in pharmacology*, 9:74, 2018.
- [353] Simulations Plus, Inc. MUT\_Risk-8.5, 2018. URL [https://www.simulations-plus.com/?s=MUT\\_Risk](https://www.simulations-plus.com/?s=MUT_Risk). Accessed 2022-03-20.
- [354] P. Smialowski, D. Frishman, and S. Kramer. Pitfalls of supervised feature selection. *Bioinformatics*, 26(3):440–443, 2010.

- [355] M. Sokolova, N. Japkowicz, and S. Szpakowicz. Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In *Australasian Joint Conference on Artificial Intelligence*, pages 1015–1021. Springer, 2006.
- [356] K. M. R. Srivalli and P. Lakshmi. Overview of p-glycoprotein inhibitors: a rational outlook. *Brazilian Journal of Pharmaceutical Sciences*, 48(3):353–367, 2012. doi: 10.1590/s1984-82502012000300002.
- [357] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15: 1929–1958, 2014.
- [358] K. srl. Dragon (software for molecular descriptor calculation). Pisa, Italy, 2016. URL [https://chm.kode-solutions.net/products\\_dragon.php](https://chm.kode-solutions.net/products_dragon.php). Version 7.0.
- [359] F. Stanzione, I. Giangreco, and J. C. Cole. Use of molecular docking computational tools in drug discovery. *Progress in Medicinal Chemistry*, 60:273–343, 2021.
- [360] T. Sterling and J. J. Irwin. ZINC 15—ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11):2324–2337, 2015. doi: 10.1021/acs.jcim.5b00559.
- [361] M. Strickert, A. J. Soto, and G. E. Vazquez. Adaptive matrix distances aiming at optimum regression subspaces. In *ESANN 2010, European Symposium on Artificial Neural Networks*, pages 93–98, 2010.
- [362] D. Sun, W. Gao, H. Hu, and S. Zhou. Why 90 *Acta Pharmaceutica Sinica B*, 12(7):3049–3062, 2022. ISSN 2211-3835. doi: <https://doi.org/10.1016/j.apsb.2022.02.002>. URL <https://www.sciencedirect.com/science/article/pii/S2211383522000521>.
- [363] J. Sun, X. Cao, H. Liang, W. Huang, Z. Chen, and Z. Li. New interpretations of normalization methods in deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5875–5882, 2020.
- [364] R.-Y. Sun. Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 8(2):249–294, 2020.
- [365] E. Swann, B. Sun, D. Cleland, and A. Barnard. Representing molecular and materials data for unsupervised machine learning. *Molecular Simulation*, 44(11):905–920, 2018.

- [366] J. Tang, J. Liu, M. Zhang, and Q. Mei. Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th International Conference on World Wide Web*, pages 287–297, 2016. doi: 10.1145/2872427.2883041.
- [367] W. Tang, J. Chen, Z. Wang, H. Xie, and H. Hong. Deep learning for predicting toxicity of chemicals: A mini review. *J. Environ. Sci. Health, Part C: Toxicol. Carcinog.*, 36(4):252–271, 2018.
- [368] Y. Tanrikulu, B. Krüger, and E. Proschak. The holistic integration of virtual screening in drug discovery. *Drug Discovery Today*, 18(7-8):358–364, 2013. doi: 10.1016/j.drudis.2013.01.007.
- [369] U. Technologies. Unity Engine, 2005. <https://unity.com>, online March 2020.
- [370] A. Thakur and A. Konde. Fundamentals of neural networks. *International Journal for Research in Applied Science and Engineering Technology*, 9:407–26, 2021.
- [371] D. Thorne, J. Kilford, M. Hollings, A. Dalrymple, M. Ballantyne, C. Meredith, and D. Dillon. The mutagenic assessment of mainstream cigarette smoke using the ames assay: A multi-strain approach. *Mutat. Res., Genet. Toxicol. Environ. Mutagen.*, 782:9–17, 2015. ISSN 1383-5718. doi: <https://doi.org/10.1016/j.mrgentox.2015.03.006>. URL <https://www.sciencedirect.com/science/article/pii/S1383571815000443>.
- [372] S. Tian, J. Wang, Y. Li, D. Li, and L. Xu. The application of in silico drug-likeness predictions in pharmaceutical research. *Advanced Drug Delivery Reviews*, 86:2–10, Junio 2015. doi: 10.1016/J.ADDR.2015.01.009.
- [373] Y. Tian and Y. Zhang. A comprehensive survey on regularization strategies in machine learning. *Information Fusion*, 80:146–166, 2022.
- [374] Y. Tian, S. Zhang, H. Yin, and A. Yan. Quantitative structure-activity relationship (qsar) models and their applicability domain analysis on hiv-1 protease inhibitors by machine learning methods. *Chemometrics and Intelligent Laboratory Systems*, 196:103888, 2020.
- [375] T. Tieleman and G. Hinton. Lecture 6.5—rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2):26, 2012.
- [376] S. Tilkov and S. Vinoski. Node.js: Using javascript to build high-performance network programs. *IEEE Internet Computing*, 14(6):80–83, 2010. doi: 10.1109/MIC.2010.145.

- [377] A. Tintó-Moliner and M. Martín. Quantitative weight of evidence method for combining predictions of quantitative structure-activity relationship models. *SAR QSAR Environ. Res.*, 31(4):261–279, 2020.
- [378] R. Todeschini and V. Consonni. *Molecular descriptors for chemoinformatics: volume I: alphabetical listing/volume II: appendices, references*, volume 41. John Wiley & Sons, 2009.
- [379] R. Todeschini and V. Consonni. *Handbook of molecular descriptors*, volume 11. John Wiley & Sons, 2010. doi: 10.1002/9783527613106.
- [380] R. Todeschini, D. Ballabio, M. Cassotti, and V. Consonni. N3 and BNN: Two New Similarity Based Classification Methods in Comparison with Other Classifiers. *Journal of Chemical Information and Modeling*, 55(11):2365–2374, Noviembre 2015. doi: 10.1021/acs.jcim.5b00326.
- [381] W. L. Todsén. Chemdoodle 6.0. *Journal of chemical information and modeling*, 54(8):2391–2393, 2014.
- [382] X. Tong, X. Liu, X. Tan, X. Li, J. Jiang, Z. Xiong, T. Xu, H. Jiang, N. Qiao, and M. Zheng. Generative models for de novo drug design. *Journal of Medicinal Chemistry*, 64(19):14011–14027, 2021.
- [383] P. H. Torres, A. C. Soderó, P. Jofily, and F. P. Silva-Jr. Key topics in molecular docking for drug design. *International journal of molecular sciences*, 20(18):4574, 2019.
- [384] P.-L. Toutain and A. Bousquet-mélou. Bioavailability and its assessment. *Journal of veterinary pharmacology and therapeutics*, 27(6):455–466, 2004.
- [385] G. Tripepi, K. Jager, F. Dekker, and C. Zoccali. Linear and logistic regression analysis. *Kidney international*, 73(7):806–810, 2008.
- [386] A. Tropsha. Best practices for qsar model development, validation, and exploitation. *Molecular informatics*, 29(6-7):476–488, 2010.
- [387] O. Trott and A. J. Olson. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010.
- [388] M. Tsang, D. Cheng, and Y. Liu. Detecting Statistical Interactions from Neural Network Weights. *arXiv preprint arXiv:1705.04977*, Mayo 2017.

- [389] J. W. Tukey. *Exploratory data analysis*, volume 2. Addison-Wesley Publishing Company, 1977.
- [390] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.
- [391] O. Ursu, A. Rayan, A. Goldblum, and T. I. Oprea. Understanding drug-likeness. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 1(5):760–781, 2011.
- [392] L. Van Der Maaten. Learning a parametric embedding by preserving local structure. In *Artificial Intelligence and Statistics*, pages 384–391, 2009.
- [393] L. Van Der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- [394] L. Van Der Maaten, E. O. Postma, H. J. van den Herik, et al. Dimensionality reduction: A comparative review. *Journal of Machine Learning Research*, 10(66-71):13, 2009.
- [395] T. Van Erven and P. Harremoës. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.
- [396] M. VaseemAkram, D. Nagarjuna, M. Ramaiah, M. Nagabhusanam, and B. Venkateswarlu. Regulatory requirements of drug master files by food and drug administration (usa), european medicines agency (europe) and health canada (canada) and their comparison. *Journal of Global Trends in Pharmaceutical Sciences*, 5(4):2220–224, 2014.
- [397] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.
- [398] A. Vellido, J. D. Martín-Guerrero, and P. J. G. Lisboa. Making machine learning models interpretable. *ESANN*, 2012.
- [399] B. Venkatesh and J. Anuradha. A review of feature selection and its methods. *Cybernetics and information technologies*, 19(1):3–26, 2019.
- [400] P. A. Vikhar. Evolutionary algorithms: A critical review and its future prospects. In *2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC)*, pages 261–265. IEEE, 2016.



- [401] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.
- [402] S. Vishveshwara, K. Brinda, and N. Kannan. Protein structure: insights from graph theory. *Journal of Theoretical and Computational Chemistry*, 1(01):187–211, 2002.
- [403] G. Vistoli, A. Pedretti, and B. Testa. Assessing drug-likeness—what are we missing? *Drug discovery today*, 13(7-8):285–294, 2008.
- [404] S. Vollert, M. Atzmueller, and A. Theissler. Interpretable machine learning: A brief survey from the predictive maintenance perspective. In *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*, pages 01–08. IEEE, 2021.
- [405] A. K. Waljee, P. D. Higgins, and A. G. Singal. A primer on predictive models. *Clinical and translational gastroenterology*, 5(1):e44, 2014.
- [406] W. P. Walters and M. A. Murcko. Prediction of ‘drug-likeness’. *Advanced drug delivery reviews*, 54(3):255–271, 2002.
- [407] M. Wang, Z. Wang, H. Sun, J. Wang, C. Shen, G. Weng, X. Chai, H. Li, D. Cao, and T. Hou. Deep learning approaches for de novo drug design: An overview. *Current Opinion in Structural Biology*, 72:135–144, 2022.
- [408] Q. Wang, Y. Ma, K. Zhao, and Y. Tian. A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, pages 1–26, 2020.
- [409] S. Wang, T. Che, A. Levit, B. K. Shoichet, D. Wacker, and B. L. Roth. Structure of the d2 dopamine receptor bound to the atypical antipsychotic drug risperidone. *Nature*, 555(7695):269–273, 2018.
- [410] S. Wang, Y. Guo, Y. Wang, H. Sun, and J. Huang. Smiles-bert: large scale unsupervised pre-training for molecular property prediction. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 429–436, 2019.
- [411] X. Wang, Y. Zhao, and F. Pourpanah. Recent advances in deep learning. *International Journal of Machine Learning and Cybernetics*, 11:747–750, 2020.

- [412] Z. Wang and J. Chen. Applicability domain characterization for machine learning qsar models. In *Machine Learning and Deep Learning in Computational Toxicology*, pages 323–353. Springer, 2023.
- [413] Z. Wang, J. Chen, and H. Hong. Developing qsar models with defined applicability domains on ppar $\gamma$  binding affinity using large data sets and machine learning algorithms. *Environmental Science & Technology*, 55(10):6857–6866, 2021.
- [414] M. A. Wani, F. A. Bhat, S. Afzal, and A. I. Khan. *Advances in deep learning*. Springer, 2020.
- [415] R. H. Waring. Cytochrome p450: genotype to phenotype. *Xenobiotica*, 50(1):9–18, 2020.
- [416] G. I. Webb, E. Keogh, and R. Mäikkulainen. Naïve bayes. *Encyclopedia of machine learning*, 15(1):713–714, 2010.
- [417] K. Weicker. *Evolutionäre algorithmen*. Springer-Verlag, 2015.
- [418] D. Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.
- [419] D. Weininger. Smiles. 3. depict. graphical depiction of chemical structures. *J. Chem. Inf. Comput. Sci.*, 30(3):237–243, 1990. ISSN 0095-2338. doi: 10.1021/ci00067a005.
- [420] G. P. Wellawatte, A. Seshadri, and A. D. White. Model agnostic generation of counterfactual explanations for molecules. *Chemical science*, 13(13):3697–3705, 2022.
- [421] D. S. Wigh, J. M. Goodman, and A. A. Lapkin. A review of molecular representation in the age of machine learning. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1603, 2022.
- [422] P. Willett. From chemical documentation to chemoinformatics: 50 years of chemical information science. *Journal of Information Science*, 34(4):477–499, 2008.
- [423] R. V. Williams, D. M. DeMarini, L. F. Stankowski Jr, P. A. Escobar, E. Zeiger, J. Howe, R. Elespuru, and K. P. Cross. Are all bacterial strains required by oecd mutagenicity test guideline tg471 needed? *Mutat. Res., Genet. Toxicol. Environ. Mutagen.*, 848:503081, 2019.
- [424] D. A. Winkler. Neural Networks as Robust Tools in Drug Lead Discovery and Development. *Molecular Biotechnology*, 27(2):139–168, Junio 2004. doi: 10.1385/MB:27:2:139.

- [425] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987. doi: 10.1016/0169-7439(87)80084-9.
- [426] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.
- [427] J. Wu, J. Wang, H. Xiao, and J. Ling. Visualization of high dimensional turbulence simulation data using t-SNE. In *19th AIAA Non-Deterministic Approaches Conference*, 2017. doi: 10.2514/6.2017-1770.
- [428] K. Wu and G.-W. Wei. Quantitative toxicity prediction using topology based multitask deep neural networks. *J. Chem. Inf. Model.*, 58(2):520–531, 2018.
- [429] Y. Wu and K. He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.
- [430] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical Science*, 9(2): 513–530, 2018.
- [431] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32:950–957, 2020.
- [432] Z. Wu, M. Zhu, Y. Kang, E. L.-H. Leung, T. Lei, C. Shen, D. Jiang, Z. Wang, D. Cao, and T. Hou. Do we need different machine learning algorithms for qsar modeling? a comprehensive assessment of 16 machine learning algorithms on 14 qsar data sets. *Briefings in Bioinformatics*, 2020.
- [433] Z. Wu, D. Jiang, C.-Y. Hsieh, G. Chen, B. Liao, D. Cao, and T. Hou. Hyperbolic relational graph convolution networks plus: a simple but highly efficient qsar-modeling method. *Briefings in Bioinformatics*, 2021.
- [434] X. Xu, T. Liang, J. Zhu, D. Zheng, and T. Sun. Review of classical dimensionality reduction and sample selection methods for large-scale data processing. *Neurocomputing*, 328:5–15, 2019. doi: 10.1016/j.neucom.2018.02.100.

- [435] Y. Xu. Deep neural networks for qsar. In *Artificial Intelligence in Drug Design*, pages 233–260. Springer, 2022.
- [436] Z. Xu, S. Wang, F. Zhu, and J. Huang. Seq2seq fingerprint: An unsupervised deep molecular embedding for drug discovery. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 285–294, 2017.
- [437] J. Xue, H. Zhang, and K. Dana. Deep texture manifold for ground terrain recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2018. doi: 10.1109/CVPR.2018.00065.
- [438] J. Yang, R. Yan, A. Roy, D. Xu, J. Poisson, and Y. Zhang. The i-tasser suite: protein structure and function prediction. *Nature Methods*, 12(1):7, 2015.
- [439] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea, A. Palmer, V. Settels, T. Jaakkola, K. Jensen, and R. Barzilay. Correction to analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59(12):5304–5305, 2019. doi: 10.1021/acs.jcim.9b01076. URL <https://doi.org/10.1021/acs.jcim.9b01076>. PMID: 31814400.
- [440] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59(8):3370–3388, 2019.
- [441] L. Yang, Y. Wang, J. Chang, Y. Pan, R. Wei, J. Li, and H. Wang. Qsar modeling the toxicity of pesticides against americamysis bahia. *Chemosphere*, 258:127217, 2020.
- [442] A. Yoshimori, T. Tanoue, and J. Bajorath. Integrating the structure–activity relationship matrix method with molecular grid maps and activity landscape models for medicinal chemistry applications. *ACS Omega*, 4(4):7061–7069, 2019. doi: 10.1021/acsomega.9b00595.
- [443] W. Yu and A. D. MacKerell. *Computer-Aided Drug Design Methods*, pages 85–106. Springer New York, New York, NY, 2017. ISBN 978-1-4939-6634-9. doi: 10.1007/978-1-4939-6634-9\_5. URL [https://doi.org/10.1007/978-1-4939-6634-9\\_5](https://doi.org/10.1007/978-1-4939-6634-9_5).
- [444] Y. Yu, X. Si, C. Hu, and J. Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

- [445] J. Yuan, C. Chen, W. Yang, M. Liu, J. Xia, and S. Liu. A survey of visual analytics techniques for machine learning. *Computational Visual Media*, 7:3–36, 2021.
- [446] J. Zhang, T. He, S. Sra, and A. Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. *arXiv preprint arXiv:1905.11881*, 2019.
- [447] L. Zhang, J. Tan, D. Han, and H. Zhu. From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug discovery today*, 22(11):1680–1685, 2017.
- [448] Y. Zhang and Q. Yang. An overview of multi-task learning. *Natl. Sci. Rev.*, 5(1):30–43, 2018.
- [449] S. Zheng, X. Yan, Y. Yang, and J. Xu. Identifying structure–property relationships through smiles syntax analysis with self-attention mechanism. *Journal of Chemical Information and Modeling*, 59(2):914–923, 2019.
- [450] W. Zhu, Z. Webb, X. Han, K. Mao, W. Sun, and J. Romagnoli. Generic process visualization using parametric t-SNE. *IFAC-PapersOnLine*, 51(18):803–808, 2018. doi: 10.1016/j.ifacol.2018.09.262.