UNIVERSIDAD NACIONAL DEL SUR

TESIS DE MAGISTER EN MATEMÁTICA

Análisis Estadístico de Datos usando S-PLUS y ROBETH

Marta Susana Malla

BAHÍA BLANCA **ARGENTINA**

2005

Prefacio

Esta tesis es presentada como parte de los requisitos para optar al grado Académico de Magister en Matemática, de la Universidad Nacional del Sur, y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otras. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el Departamento de Matemática de la Universidad Nacional del Sur, durante el período comprendido entre los meses de diciembre de 1997 y mayo del 2005, bajo la dirección del Dr. Oscar Humberto Bustos, Profesor Titular de la Facultad de Matemática, Astronomía y Física de la Universidad Nacional de Córdoba y la codirección del Dr. Eduardo de Weerth, ex-Profesor Titular del Departamento de Matemática de la Universidad Nacional del Sur.

Agradecimientos

Deseo expresar mi agradecimiento al Dr. Oscar Humberto Bustos por su confianza e invalorable apoyo en la realización de este trabajo.

Al Dr. Eduardo de Weerth por su contribución a mi formación académica.

Y muy especialmente agradezco a mi familia y amigos por el apoyo constante para que alcanzara mi meta.

31 de Mayo del 2005

Departamento de Matemática

UNIVERSIDAD NACIONAL DEL SUR

Resumen

Se analizan tres aspectos relacionados con los procedimientos robustos, especialmente en el modelo de regresión lineal simple y multivariada. El primer aspecto se refiere a los fundamentos teóricos de los métodos robustos presentándose un resumen de la notación adoptada en los problemas a tratar. El segundo aspecto trata sobre la implementación del "software" ROBETH sobre plataforma S-PLUS con aplicaciones a conjuntos de datos. El tercer aspecto trata de la realización de una pequeña guía explicativa de los principales rasgos de programación y convenciones del ROBETH. Esta tesis está organizada de la siguiente manera: el Capítulo 1 trata sobre el análisis de los problemas de locación de una y dos muestras; el Capítulo 2 se refiere a soluciones robustas de los problemas de regresión lineal basadas en M-estimadores; el Capítulo 3 se refiere al cálculo de la estimación de la matriz de covarianza de los coeficientes estimados según el Capítulo 2. Y finalmente el Capítulo 4 se refiere al cálculo de estimadores con alto punto de ruptura en regresión lineal. Se incluye, además, apéndices que contienen rasgos de programación y convenciones del ROBETH. Se adjunta un CD, en donde se detalla el desarrollo de cada una de las subrutinas tratadas en los capítulos anteriores y algunas subrutinas de servicio mencionadas en el apéndice A.

Índice General

Introducción

En análisis de regresión, una de las herramientas más usadas, para ajustar una tendencia lineal es el método de mínimos cuadrados, que ante la presencia de datos outliers (verticales y horizontales) no es recomendable, dado que un "outlier" cuanto más exagerado sea, hará que el ajuste lineal tienda a pasar cerca de él y alejarse del grueso de los datos. Ante la presencia de datos influenciales una de las alternativas a mínimos cuadrados es el uso de métodos de regresión robustos, que tienen como objetivo ajustar un modelo lineal que resista la influencia de los outliers (ver Yohai [37]). Existe una variedad de métodos robustos de regresión que son agrupados principalmente en tres clases:

1. M-regresión (M, por máxima verosimilitud)
2. R-regresión (R, por rangos)
3. L-Regresión (L, por combinación lineal de estadísticos de orden)

En los siguientes ejemplos se muestra la presencia de outliers reflejando la influencia que tienen sobre los parámetros de la regresión lineal simple. Los datos de la Figura 1, forman una nube de puntos de 77 observaciones de una industria, recolectados para un estudio de conta minación (Guillemin et al. [6]). En dicha figura se observan tres puntos especiales, denotados con 1, 2 y 3. A la totalidad de los puntos se le ajustó una recta por el método de mínimos cuadrados (por medio de la subrutina RICLLS, descripta más adelante) y se encontró que $\widehat{\sigma}$, la clásica estimación del desvío estándar del error, fue de 0.59. Se observaron los residuales más grandes que tres veces σ con especto a la recta de mínimos cuadrados, y a pesar de que este límite fue arbitrario, se obtuvo que de acuerdo con el modelo gaussiano a lo sumo el 0.3% de los residuales (es decir, menos de 1 en 77) fueran mayores, en valor absoluto que 3σ. El residual de la observación 1 fue de -2.57 y $2.57 > 3(\widehat{\sigma}) = (3)(0.59) = 1.77$, pero mirando los datos originales se encontró que el valor de y era incorrecto. Corrigiendo este dato se obtuvo un $\widehat{\sigma} = 0.53$. El residual de la observación 2 dio $1.61 > 3(\widehat{\sigma}) = (3)(0.53) = 1.59$. Luego se quito el punto 2 ya que parecía conveniente su extracción. Al sacar el punto 2, $\widehat{\sigma}$ se transformó en 0.49. El residual de la observación 3 fue de -1.5, y $1.5 > (3)(0.49)$. No existía ninguna explicación con respecto a este outlier, era efectivamente un dato real, no obstante se quitó obteniéndose la reducción del $\widehat{\sigma}$ al valor 0.47.

Resumiendo, se observa que corrigiendo la observación 1, y quitando las observaciones 2 y 3 no existe ninguna diferencia respecto de la estimación de la pendiente y de la intersección con el eje y, pero sí existe un cambio en la estimación clásica de σ. Esto tiene un efecto significativo en la región de tolerancia como se ve en la Figura 1.

Outliers

Las observaciones 1, 2 y 3 son ejemplos de outliers. En general entendemos como outliers aquellas observaciones individuales alejadas del patrón establecido por la mayoría de los datos. Ocurren principalmente por errores de registro (como la observación 1), o constituyen observaciones que accidentalmente son miembros de otra población (como la 2), o por legítimas observaciones extremas (como la 3). Los outliers 1, 2 y 3 se desvían del grueso de los datos dado que tienen valores discordantes respecto de la variable dependiente; llamaremos a ellos: *outliers en la dirección y.* Este tipo de outliers se observan frecuentemente cuando los valores de la variable explicativa son fijos, de acuerdo a un diseño de experimento. Si bien los tres outliers expuestos en el ejemplo anterior aumentaron la estimación de σ, no han obtenido sustancial efecto en la estimación de los coeficientes. Es fácil imaginar outliers con efectos mucho más dramáticos. La Figura 2 muestra 13 observaciones con un outlier P en la dirección y. Los datos, tomados de Hampel et al., p. 310 [8] muestran qué tan importante es la influencia que ejerce P en la estimación de los coeficientes de la recta de regresión, como también sobre la estimación del error estándar.

Observaciones con un outlier en la dirección x

La Figura 3 corresponde al mismo conjunto de datos que la Figura 2. En ella, el punto P(19.7,44.9) fué sustituído por P(77.6,44.9) que tiene la componente x muy distante aunque el punto P está bien alineado respecto de los otros puntos. La recta de regresión por mínimos cuadrados calculada con todos los datos (línea punteada) está próxima a la recta de regresión por mínimos cuadrados sin el punto P (línea punteada suave). En contraste, si el punto P se sustituye por Q(77.6,15.7) el resultado esta fuertemente sesgado por la presencia del punto Q, ya que la recta estimada con Q (línea sólida) es un resumen engañoso de la relación entre la variable respuesta y la dependiente. Este tipo de outlier en la dirección x es llamado *punto palanca.*

Contaminación abundante

El porcentaje de puntos que pertenece a otras poblaciones, también llamada proporción de contaminación, puede afectar la elección de las técnicas estadísticas. En el primer ejemplo tratado el porcentaje de contaminación, fue sólo del 4 %. Pero Rousseeuw y Leroy [27], en el ejemplo de las llamadas telefónicas internacionales desde Bélgica (en decenas de millones) dan datos con alta contaminación, de aproximadamente el 30 %.Ver Figura 4, y Venables and Ripley, Cap. 8 [36], donde se puede observar que entre los años 1964 y 1967 se anotaron erróneamente la cantidad de minutos, en vez de la cantidad de llamadas realizadas. A este conjunto de datos se le ajustaron rectas robustas: least trimmed squares (LTS) y least median of squares (LMS).

No todos los outliers tienen el mismo impacto sobre los resultados del análisis de regresión, y en función de su ubicación y su frecuencia podemos distinguir tres situaciones:

i) outliers en la dirección de y,

ii) outliers en la dirección de x o *puntos palanca,*

iii) alta frecuencia (hasta un 50%) de outliers, con respecto tanto a x como a y.

Tipos de problemas y métodos robustos para regresión lineal

El principal propósito de los procedimientos robustos es dar un resultado resistente, ante la presencia y ausencia de outliers. Con la idea de alcanzar esta estabilidad se darán diferentes procedimientos robustos de acuerdo a tres tipos de problemas:

- Problemas con outliers en la dirección y, como se ve en las Figuras 1 y 2. Los métodos robustos comunmente usados para esta clase de problemas están basados en los estimadores del tipo Huber.

- Problemas con moderado porcentaje de outliers multivariado en el espacio de covarianzas (es decir: outliers en el espacio x o puntos palancas), como se ve en las Figuras 3 y 4. Para este tipo de problemas, usaremos estimadores del tipo Mallows y estimadores del tipo Schweppe.

- Problemas donde la frecuencia de outliers, con respecto a ambas componentes: x e y puede ser muy alta. Para esta clase de problemas usaremos estimadores con alto punto de ruptura.

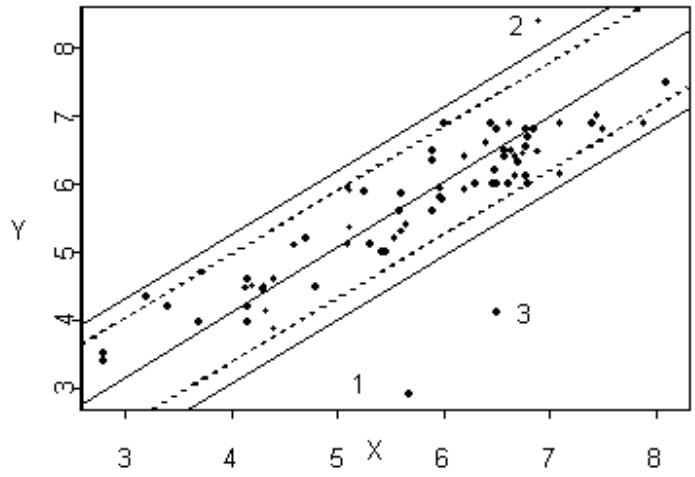

Figura 1: Nube de puntos con 77 elementos. Ajuste por mínimos cuadrados (LS) y límites de tolerancia del 95% con todos los datos (línea continua) y sin los puntos 1, 2 y 3 (línea punteada).

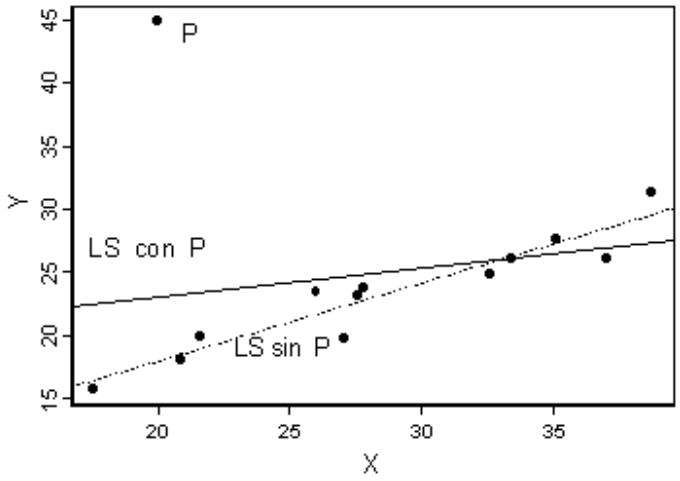

Figura 2: Outlier P en la dirección y. Ajuste por mínimos cuadrados con P (línea continua) y sin P (línea punteada).

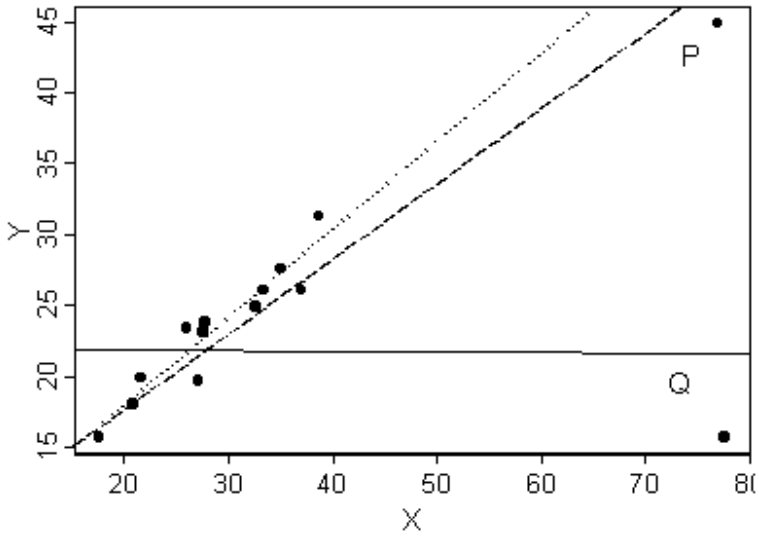

Figura 3: LS con todos los datos (línea punteada), LS sin el punto P (línea punteada suave), LS con el punto Q que reemplaza a P (línea continua).

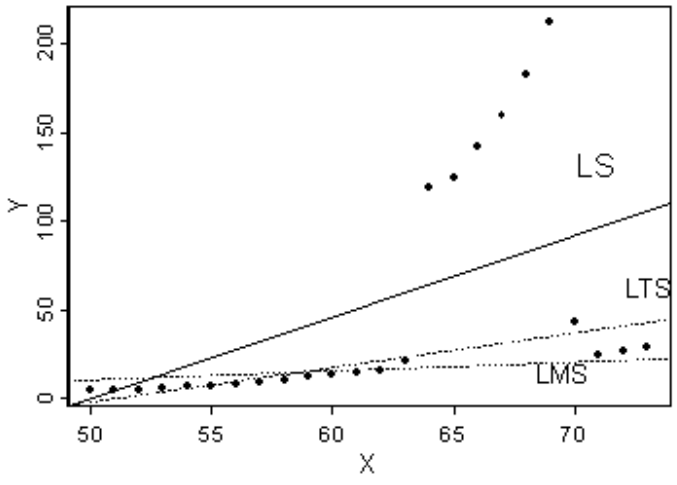

Figura 4: Contaminación abundante, rectas LS, LTS y LMS.

Capítulo 1

Problemas de Locación

Con el objetivo de implementar el programa de computación ROBETH, según Marazzi and Randriamiharisoa [18], sobre plataforma S-PLUS [31] [32] [33] se explicarán, en este capítulo las subrutinas agrupadas con el nombre de LCMAIN. Ellas se refieren a los procedimientos incluidos en el programa ROBETH para el análisis de los problemas de locación de una y dos muestras.

1.1 Introducción

Si las observaciones siguen una distribución normal, los procedimientos clásicos basados en medias aritméticas, desvíos estándar, t-tests, etc., son los más eficientes. Desafortunadamente, no son robustos cuando la distribución normal es justamente un modelo paramétrico aproximado. En este caso, existen procedimientos resistentes y eficientes basados en M-estimadores. Más aún, en los problemas de una y dos muestras, procedimientos robustos adecuados y simples se pueden derivar de los populares tests no paramétricos. Los casos más importantes se pueden encontrar en ROBETH.

1.1.1 El problema de una muestra: aproximación paramétrica

Siguiendo la notación de Huber [10] y Hampel et al. [8], sea $y_1, y_2, \cdots, y_n$ una muestra de n observaciones. Supongamos que:

A_1: $y_i = \theta + e_i$, para i=1, $\cdots$,n.

A_2: Los errores e_i, para i=1, $\cdots$,n, independientes e idénticamente distribuidos.

donde θ es el *parámetro de locación* desconocido. Sea $L(\cdot/\sigma)$ la distribución del error donde σ es un *parámetro de escala*. Usualmente $L(\cdot/\sigma) \approx \Phi(\cdot)$, la distribución normal estándar con densidad $\phi(s) = (1/\sqrt{2\pi})\exp(-s^2/2)$, con σ desconocido.

1.1.2 Procedimiento clásico

Los clásicos estimadores de mínimos cuadrados de θ y σ son la *media aritmética* y el *desvío estándar empírico*:

$$\widehat{\theta}_{LS} = \frac{1}{n}\sum_{i=1}^{n} y_i, \tag{1.1}$$

$$\widehat{\sigma}_{LS}=\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \widehat{\theta}_{LS})^2}\ . \tag{1.2}$$

El límite inferior y superior t_L, t_U del intervalo de confianza que incluye a θ con probabilidad $1-2\alpha$ estan dados por:

$$t_L = \widehat{\theta}_{LS} - t_{\alpha,n-1}\left(\frac{\widehat{\sigma}_{LS}}{\sqrt{n}}\right) \qquad \text{y} \qquad t_U = \widehat{\theta}_{LS} + t_{\alpha,n-1}\left(\frac{\widehat{\sigma}_{LS}}{\sqrt{n}}\right), \tag{1.3}$$

donde $t_{\alpha,n-1}$ es el α-cuantil superior de la distribución t con (n-1) grados de libertad.

1.1.3 Mediana y desviación absoluta mediana

Un simple estimador resistente de θ está dado por la *mediana* de las observaciones

$$\widehat{\theta}_{MED} = \underset{i}{med}\ (yi), \tag{1.4}$$

donde

$$\underset{i}{med}\ (yi) = \begin{cases} y_{[(n+1)/2]} & \text{si n es impar,} \\ (y_{[n/2]} + y_{[(n/2)+1]})/2 & \text{si n es par;} \end{cases} \tag{1.5}$$

donde $y_{[1]} <= y_{[2]} <= \cdots <= y_{[n]}$ constituye la muestra ordenada. Un simple estimador resistente de σ es:

$$\widehat{\sigma}_{MAD} = MAD/\beta_0\ \ , \tag{1.6}$$

donde

$$MAD = \underset{i}{med}\ (\mid y_i - \widehat{\theta}_{MED} \mid) \tag{1.7}$$

es la *desviación absoluta mediana (MAD)*, y $(y_i - \widehat{\theta}_{MED})$ denota el i-ésimo residual (ver Seber, p. 160 [29]). Aquí β_0 es una constante adecuada. La elección clásica de β_0 es

$$\beta_0 = \Phi^{-1}(3/4) \approx 0.6745 \tag{1.8}$$

que hace al estimador $\widehat{\sigma}_{MAD}$ asintóticamente insesgado cuando $L(\cdot/\sigma) = \Phi(\cdot)$. La varianza asintótica de $\widehat{\theta}_{MED}$ es

$$A(\widehat{\theta}_{MED}, L) = \frac{1}{4l(0)^2} \cdot \sigma^2, \tag{1.9}$$

donde l es la densidad de L según Hampel et al. [8]. Luego, usando $l(0) = \frac{1}{\sqrt{2\pi}}$, la varianza de $\widehat{\theta}_{MED}$ se puede estimar por

$$\widehat{V}(\widehat{\theta}_{MED}) = \frac{\pi}{2} \cdot (\frac{\widehat{\sigma}^2_{MAD}}{n}), \tag{1.10}$$

y un intervalo de confianza aproximado que incluye a θ con probabilidad 1-2α es

$$(\widehat{\theta}_{MED} - z_\alpha \cdot \sqrt{\widehat{V}(\widehat{\theta}_{MED})}, \qquad \widehat{\theta}_{MED} + z_\alpha \cdot \sqrt{\widehat{V}(\widehat{\theta}_{MED})}\ ), \tag{1.11}$$

donde z_α es el α-cuantil superior de la distribución normal estándar.

1.1.4 M-estimadores de locación y de escala

La mediana y la desviación absoluta mediana no son muy eficientes cuando $L(\cdot/\sigma) = \Phi(\cdot)$, pero un buen compromiso entre resistencia y eficiencia en el modelo gaussiano se puede obtener con los M-estimadores. Si σ es conocido, un *M-estimador* $\widehat{\theta}$ de θ está definido implícitamente por la solución de

$$\sum_{i=1}^{n} \Psi(\frac{y_i - \theta}{\sigma}) = 0, \tag{1.12}$$

donde Ψ es una función adecuada. En esta ecuación σ se supone conocido. Pero el parámetro de escala σ es usualmente desconocido. El programa ROBETH permite

estimarlo resolviendo la ecuación

$$\sum_{i=1}^{n} \chi(\frac{y_i - \theta}{\sigma}) = (n-1)\beta \tag{1.13}$$

para σ simultáneamente con (1.12). Aquí, χ es otra función adecuada y β es una constante adecuada. Alternativamente, σ puede estimarse resolviendo

$$\sigma = \underset{i}{med}\ (\mid y_i - \theta \mid)/\beta_0 \tag{1.14}$$

simultáneamente con (1.12), donde β_0 está definida en (1.8).

Usualmente, las funciones Ψ y χ tienen que cumplir con ciertas condiciones con la finalidad de asegurar existencia y unicidad de la solución y convergencia del algoritmo iterativo usado para sus cálculos según Huber [10]. Por ejemplo, podemos asumir $\psi(x) = \rho'(x)$ y $\chi(x) = x.\psi(x) - \rho(x)$ para alguna función convexa y positiva ρ tal que $\rho(0) = 0$. En este caso ψ es monótona creciente. Un caso importante especial es la *función de Huber* $\psi(x) = \psi_c(x) = máx\,[-c, mín(c, x)]$ donde c es una constante que el usuario especifica teniendo en cuenta la pérdida de eficiencia que está dispuesto a aceptar ante el modelo gaussiano en intercambio de la resistencia a los outliers. Son interesantes las $\psi-$funciones tales que $\psi(s) \rightarrow 0$ para $\mid s \mid \rightarrow \infty$ (ver Hampel et al., p. 149 [8]). Un ejemplo es la *función de Tukey doblemente pesada*

$$\psi(s) = \begin{cases} s.(1-s^2)^2 & \text{con -1<=s <= 1,} \\ 0 & \text{en el resto.} \end{cases}$$

El programa ROBETH proporciona una serie de funciones FORTRAN de las cuales se puede seleccionar un par particular ψ y χ .

Observación: La media aritmética, y el desvío estándar se pueden obtener de (1.12) y (1.13) usando $\psi(s) = s$, $\chi(s) = s^2/2$ y $\beta = 1/2$. Similarmente, la mediana y la desviación absoluta mediana son casos especiales de (1.12) y (1.13) con $\psi(s) = sign(s)$. Por problemas computacionales, el programa ROBETH trata estos casos por separado.

ROBETH proporciona subrutinas para la determinación de la constante β de tal manera que la estimación σ obtenida de (1.13) sea asintóticamente insesgada para errores

normales. Con este fin tomamos

$$\beta = \int \chi(s) \cdot d\Phi(s). \tag{1.15}$$

La varianza asintótica de $\widehat{\theta}$ es

$$A(\widehat{\theta}, L) = \frac{E(\Psi^2)}{E(\Psi')^2} \cdot \sigma^2, \tag{1.16}$$

donde

$$E(\Psi^2) = \int \Psi^2(s) \cdot dL(s/\sigma) \quad \text{y} \quad E(\Psi') = \int \Psi^{'}(s) \cdot dL(s/\sigma). \tag{1.17}$$

Para el cálculo de $E(\Psi^2)$ *y* $E(\Psi')$, en el caso $L(\cdot/\sigma) \equiv \Phi(\cdot)$, se pueden encontrar subrutinas en LCMAIN. Finalmente, la varianza de $\widehat{\theta}$ se puede estimar por

$$\widehat{V}(\widehat{\theta}) = \frac{\widehat{\sigma}^2}{n} \cdot \left[\frac{\frac{1}{n-1} \cdot \sum_{i=1}^{n} \Psi^2((y_i - \widehat{\theta})/\widehat{\sigma})}{(\frac{1}{n} \cdot \sum_{i=1}^{n} \Psi'((y_i - \widehat{\theta})/\widehat{\sigma}))^2} \right], \tag{1.18}$$

y un intervalo de confianza aproximado que contiene a θ con probabilidad 1-2α es

$$(\widehat{\theta} - z_\alpha \cdot \sqrt{\widehat{V}(\widehat{\theta})}, \quad \widehat{\theta} + z_\alpha \cdot \sqrt{\widehat{V}(\widehat{\theta})}), \tag{1.19}$$

donde z_α es el α-cuantil superior de la distribución normal estándar.

1.2 El problema de una muestra: aproximación no-paramétrica

Siguiendo la notación de Lehmann [16] y de Sprent [34]: sean $y_1, y_2, \cdots, y_n$ n observaciones independientes con distribución común continua $L(\cdot)$, y sean $y_{[1]} < y_{[2]} < \cdots < y_{[n]}$ las observaciones ordenadas.

1.2.1 Intervalo de confianza para la mediana basado en el test estadístico del signo

Sea μ la mediana de $L(\cdot)$. Se vió que μ se puede estimar por $\underset{i}{med}(y_i)$. Sea S el *test*

estadístico del signo para la hípotesis $H_0 : \mu = 0$:

$$S = \text{número de observaciones } y_i \text{ tal que } y_i > 0. \tag{1.20}$$

Se puede ver que

$$P_\mu(\mu <= y_{[k]}) = P_\mu(\mu \geq y_{[n-k+1]}) = P_0(S <= k-1), \tag{1.21}$$

donde

$$P_0(S <= k-1) = \sum_{i=0}^{k-1} \binom{n}{i} \frac{1}{2^n} \quad \text{para} \quad k = 1, \cdots, n. \tag{1.22}$$

Aqui, P_μ denota la medida de probabilidad definida por $L(\cdot)$, P_0 indica la probabilidad que se obtiene bajo la distribución nula de S (i.e., para $\mu = 0$).

Para $n > 10$ se puede usar la aproximación

$$P_0(S <= k-1) \approx F_S(k) \tag{1.23}$$

donde

$$F_S(k) = \Phi(\frac{2k-n-1}{\sqrt{n}}). \tag{1.24}$$

La subrutina LCMAIN permite obtener el índice k para el cual $F_S(k)$ esta lo más próximo posible a una probabilidad especificada α. El intervalo de confianza para μ se obtiene de

$$P_\mu(y_{[k]} < \mu < y_{[n-k+1]}) \approx 1 - 2\alpha. \tag{1.25}$$

Alternativamente, el índice k se puede obtener usando las tablas de la distribución binomial.

1.2.2 Intervalo de confianza para el centro de simetría basado en el test estadístico de Wilcoxon de una muestra

Sean $y_1, y_2, \cdots, y_n$ n observaciones independientes con distribución común continua y simétrica $L(\cdot - \theta)$ cuyo centro de simetría θ se desea estimar. LCMAIN permite calcular

el *estimador Hodges-Lehmann* de θ definido por

$$\widehat{\theta}_{HL} = \underset{(i,j)}{med}\ (\frac{(y_i + y_j)}{2}, 1 <= i <= j <= n). \tag{1.26}$$

Complementariamente LCMAIN incluye subrutinas para el cálculo de intervalos de confianza para θ de acuerdo al siguiente procedimiento: Sea m=n(n+1)/2 y sea $a_{[1]} < a_{[2]} < \cdots < a_{[m]}$ denota los m promedios ordenados $\frac{(y_i+y_j)}{2}$ para $1 <= i <= j <= n$. Entonces,

$$P_\theta(\theta <= a_{[k]}) = P_0(V_S <= k-1) \ \text{ para } \quad k = 1, \cdots, m, \tag{1.27}$$

donde P_θ denota la medida de probabilidad definida para $L(\cdot - \theta)$, V_s es el test estadístico de Wilcoxon para una muestra, y el subíndice cero en el segundo miembro de la igualdad anterior indica la probabilidad que se calcula bajo la distribución nula de V_s. La siguiente aproximación se puede usar cuando $n > 20$:

$$P_0(V_S <= k-1) \approx F_V(k) \tag{1.28}$$

con

$$F_V(k) = \Phi(\frac{2k - m - 1}{\sqrt{m \cdot (2n+1)/3}}). \tag{1.29}$$

La subrutina LCMAIN incluye una subrutina que permite obtener el índice k para el cual $F_V(k)$ esté lo más próximo posible a una probabilidad especificada α. Otra subrutina permite calcular $a_{[k]}$ para un valor específico de k, como asi también se pueden construir los intervalos de confianza. En realidad,

$$P_\theta(\theta <= a_{[k]}) = P_\theta(\theta \geq a_{[l]}) \tag{1.30}$$

para $l = m + 1 - k$, por lo tanto

$$P_\theta(a_{[k]} < \theta < a_{[l]}) \approx 1 - 2\alpha. \tag{1.31}$$

El índice k se puede obtener usando tablas de la distribución nula de V_S.

Observación: Si m es impar y $k = \frac{m+1}{2}$, entonces $\widehat{\theta}_{HL} = a_{[k]}$.

Si m es par y $k_1 = \frac{m}{2}$ y $k_2 = k_1 + 1$, entonces $\widehat{\theta}_{HL}$=($a_{[k_1]}$+$a_{[k_2]}$)/2. En otras palabras, para calcular el estimador de Hodges-Lehmann, es suficiente calcular $a_{[k]}$ o $a_{[k_1]}$ y $a_{[k_2]}$.

1.3 El problema de dos muestras: aproximación paramétrica

Según Hampel et al. [8], sean $x_1, x_2, \cdots, x_m$ y $y_1, y_2, \cdots, y_n$ dos muestras de m y n observaciones respectivamente. Supongamos

B_1: $x_i = \theta + e_{1i}$ para $i = 1, \cdots, m$,

$y_j = \theta + e_{2j} + \triangle$, para $j = 1, \cdots, n$.

B_2: Los errores $e_{1i},\ i = 1, \cdots, m$ y los errores $e_{2j},\ j = 1, \cdots, n$ son independientes e idénticamente distribuidos.

donde θ es el *parámetro de locación* de la primer muestra, y $\triangle$ es el *parámetro principal* de la segunda muestra. Sea $L(\cdot/\sigma)$ la distribución de los errores, donde σ es un *parámetro de escala* (usualmente desconocido). Comunmente, $L(\cdot/\sigma) \approx \Phi(\cdot)$ es la distribución normal estándar.

1.3.1 t-test e intervalos de confianza asociados al parámetro principal

El procedimiento clásico para testear la hipótesis H_0: $\triangle = 0$ está basado en

$$\widehat{\triangle} = \overline{y} - \overline{x}, \tag{1.32}$$

$$\widehat{\sigma}_1 = \sqrt{\frac{1}{m-1} \cdot \sum (x_i - \overline{x})^2}, \tag{1.33}$$

$$\widehat{\sigma}_2 = \sqrt{\frac{1}{n-1} \cdot \sum (y_i - \overline{y})^2}, \tag{1.34}$$

$$\widehat{\sigma} = \sqrt{\frac{(m-1) \cdot \widehat{\sigma}_1^2 + (n-1) \cdot \widehat{\sigma}_2^2}{(m-1) + (n-1)}}, \tag{1.35}$$

$$t = \frac{\widehat{\triangle}}{\sigma \cdot \sqrt{\frac{1}{m} + \frac{1}{n}}}. \tag{1.36}$$

donde $\overline{x}$ e $\overline{y}$ son las medias aritméticas de las dos muestras, $\widehat{\triangle}$ es un estimador de $\triangle$, $\widehat{\sigma}_1$, $\widehat{\sigma}_2$ son estimadores de σ basados en las dos muestra, $\widehat{\sigma}$ es la *estimación ponderada* de σ , t es el valor observado del test estadístico. La significación del test se puede medir

usando P$[T > t]$, donde T es un variable aleatoria con distribución t con (m+n-2) grados de libertad. Los límites inferior y superior t_L, t_U de un intervalo de confianza que incluye a $\triangle$ con probabilidad 1-2α son:

$$t_L = \widehat{\triangle} - t_{\alpha,m+n-2}\ (\widehat{\sigma}.\sqrt{\frac{1}{m}+\frac{1}{n}}) \tag{1.37a}$$

$$t_U = \widehat{\triangle} + t_{\alpha,m+n-2}\ (\widehat{\sigma}.\sqrt{\frac{1}{m}+\frac{1}{n}}), \tag{1.37b}$$

donde $t_{\alpha,m+n-2}$ es el α-cuantil superior de la distribución t con n+m-2 grados de libertad.

1.3.2 $\tau-$test para el parámetro principal e intervalos de confianza basados en M-estimadores

Supongamos que $L(\cdot/\sigma)$ es simétrica y consideremos el modelo lineal

$$\Omega : z = X_\Omega \vartheta_\Omega + e, \tag{1.38}$$

donde $z = (x_1, x_2, \cdots, x_m, y_1, y_2, \cdots, y_n)^T$, $\quad e = (e_{11}, e_{12}, \cdots, e_{1m}\ , e_{21}, e_{22}, \cdots, e_{2n})^T$

$\vartheta_\Omega = (\vartheta_1, \vartheta_2)^T = (\theta, \triangle)^T$

y $X_\Omega = (a_{hk})$ para $k = 1, 2$ y para $h = 1, \cdots, m+n$ con:

$$\begin{aligned} a_{h1} &= 1 \quad \text{para} \quad h = 1, \cdots, m+n, \\ a_{h2} &= 0 \quad \text{para} \quad h = 1, \cdots, m \\ a_{h2} &= 1 \quad \text{para} \quad h = m+1, \cdots, m+n. \end{aligned}$$

Los estimadores tipo Huber $\widehat{\vartheta} = (\widehat{\vartheta}_1, \widehat{\vartheta}_2)^T$ y $\widehat{\sigma}$ de ϑ_Ω y de σ estan definidos como solución del sistema de ecuaciones:

$$\sum_{h=1}^{m+n} \Psi\left(\left(\frac{z_h - \sum_{k=1}^{2} a_{hk}\vartheta_k}{\sigma}\right) \cdot a_{hl}\right) = 0, \qquad para\ l = 1, 2, \tag{1.39}$$

$$\sum_{h=1}^{m+n} \chi\left(\left(\frac{z_h - \sum_{k=1}^{2} a_{hk}\vartheta_k}{\sigma}\right)\right) = (n-2)\beta, \tag{1.40}$$

donde Ψ y χ son funciones definidas adecuadamente y β es una constante adecuada. Por ejemplo $\psi(x) = \psi_c(x) = máx\,[-c, mín(x, c)]$, donde $\chi(x) = \psi_c^2(x)/2$, y β es la constante definida en (1.15). Opcionalmente, (1.40) se puede reemplazar por

$$\sigma = \underset{h}{med}\ (\mid z_h - \sum_{k=1}^{2} a_{hk}\vartheta_k \mid)/\beta_0, \tag{1.41}$$

donde β_0 es una constante adecuada dada por el experimentador (ver por ejemplo (1.8)) o σ puede ser una constante adecuada. Imponiendo la condición H_0: $\triangle = 0$ en Ω, se obtiene el submodelo

$$\omega\text{: } z_h = \theta + e_h, \quad \text{para } h = 1, \cdots, m+n. \tag{1.42}$$

En este modelo una estimación $\widehat{\theta}$ de θ se calcula resolviendo

$$\sum_{h=1}^{m+n} \Psi(\frac{z_h - \theta}{\widehat{\sigma}}) = 0. \tag{1.43}$$

Notemos que en (1.43) el parámetro de escala está dado por $\widehat{\sigma}$. Denotamos por $r_{\Omega,h}$ los residuales bajo Ω y por $r_{w,h}$ los residuales bajo ω. Entonces el $\tau-$*test estadístico* robusto para testear H_0 está definido como

$$F_r = 2\sum_{h=1}^{m+n}(\rho(\frac{r_{w,h}}{\widehat{\sigma}}) - \rho(\frac{r_{\Omega,h}}{\widehat{\sigma}})), \tag{1.44}$$

donde ρ es una función tal que $\rho' = \psi$. La significación del test se puede evaluar aproximadamente como $P(Z > F_r/f_H)$, donde

$$f_H = \frac{\int \psi^2(s)d\Phi(s)}{\left[\int \psi'(s)d\Phi(s)\right]^2}$$

donde Z es una variable aleatoria Chi-cuadrado con un grado de libertad. La matriz de covarianza asintótica de $\widehat{\vartheta} = (\widehat{\theta}, \widehat{\triangle})^T$ bajo Ω es

$$K(\widehat{\vartheta}, L) = (m+n)\sigma^2 \frac{E(\psi^2)}{E(\psi')^2}(X_\Omega^T X_\Omega)^{-1}, \tag{1.45}$$

donde $E(\psi^2)$ y $E(\psi')$ son constantes definidas en (1.17) y se pueden computar usando la subrutina LCMAIN. Más aún, la matriz de covarianza de $\widehat{\vartheta}$ se puede estimar por

$$\widehat{C}(\widehat{\vartheta}) = \widehat{\sigma}^2 \frac{\left(\frac{1}{m+n-2}\right) \sum \psi^2(r_{\Omega,h}/\widehat{\sigma})}{(\left(\frac{1}{m+n}\right) \sum \psi'(r_{\Omega,h}/\widehat{\sigma}))^2} (X_\Omega^T X_\Omega)^{-1}. \tag{1.46}$$

Por lo tanto, un intervalo de confianza aproximado que incluye a $\triangle$ con probabilidad 1-2α es

$$\left(\widehat{\triangle} - z_\alpha \sqrt{\widehat{c}_{22}}, \quad \widehat{\triangle} + z_\alpha \sqrt{\widehat{c}_{22}}\right), \tag{1.47}$$

donde $\widehat{c}_{22}$ es el segundo elemento diagonal de la matriz de covarianza $\widehat{C}(\widehat{\vartheta})$ y z_α es el $\alpha-$cuartil superior de la distribución normal estándar.

Observación: $(X_\Omega^T X_\Omega)^{-1} = \begin{pmatrix} 1/m & -1/m \\ -1/m & (1/m + 1/n) \end{pmatrix}$.

1.4 El problema de dos muestras: aproximación no-paramétrica

En este parte nos referiremos al Lehmann [16]. Sean $x_1, x_2, \cdots, x_m$ m observaciones independientes con distribución común continua $L(\cdot)$ y sean $y_1, y_2, \cdots, y_n$ n observaciones independientes con distribución común continua $L(\cdot - \Delta)$ donde Δ es el *parámetro principal.*

1.4.1 Intervalo de confianza para el parámetro principal basado en el test estadístico de Mann-Whitney

Sea W_{XY} el *test estadístico de Mann-Whitney.* Es decir,

$$W_{XY}=\text{número de pares } (x_i, y_j) \text{ tales que } x_i < y_j. \tag{1.48}$$

Sea $d_{[1]} < d_{[2]} < \cdots < d_{[mn]}$ las mn diferencias ordenadas $y_j - x_i$. El parámetro principal Δ se puede estimar por medio de

$$\widehat{\Delta}_{XY} = \underset{(i,j)}{med}\ (y_j - x_i). \tag{1.49}$$

Se puede ver que

$$P_\Delta(\Delta <= d_{[k]}) = P_\Delta(\Delta \geq d_{[mn-k+1]}) = P_0(W_{XY} <= k-1), \text{ para } k = 1, \cdots, mn, \quad (1.50)$$

donde P_Δ denota la medida de probabilidad definida por $L(\cdot - \Delta)$, y el subíndice cero indica la probabilidad que se obtiene bajo la distribución nula de W_{XY} (i.e., para $\Delta = 0$).

Para $m > 10$ y $n > 10$ se puede usar la siguiente aproximación

$$P_0(W_{XY} <= k-1) \approx F_W(k), \quad (1.51)$$

donde

$$F_W(k) = \Phi(\frac{2k - mn - 1}{\sqrt{mn(m+n+1)/3}}). \quad (1.52)$$

En LCMAIN se incluye una subrutina que permite obtener el índice k para el cual $F_W(k)$ esté lo más próximo posible a una probabilidad especificada α. Otra subrutina permite el cálculo de $d_{[k]}$ para un valor específico de k. Intervalos de confianza para Δ se obtienen de

$$P_\Delta(d_{[k]} < \Delta < d_{[mn-k+1]}) \approx 1 - 2\alpha. \quad (1.53)$$

Alternativamente, el índice k se puede obtener usando las tablas de la distribución nula de W_{XY}.

Observación: Si mn es par, entonces $\widehat{\Delta}_{XY} = d_{[(mn+1)/2]}$. Si mn es impar, entonces $\widehat{\Delta}_{XY} = \left(d_{[mn/2]} + d_{[(mn)/2+1]}\right) \diagup 2$.

1.5 Descripción de las subrutinas en LCMAIN

El problema de una muestra

LICLLS	Estimación clásica de la media y el desvío estándar.
LILARS	Mediana y desviación absoluta mediana.
LYHALG	M-estimador de locación y de escala.
LYHDLE	Hodges-Lehmann estimador e intervalo de confianza para el centro de simetría basado en el test estadístico de Wilcoxon para una muestra.
LIINDS	Inversa de la aproximación de la distribución nula del test del signo.
LIINDH	Inversa de la aproximación de la distribución nula del test de Wilcoxon para una muestra.

El problema de dos muestras

LITTST	t-test para el parámetro principal.
LYTAU2	τ-test para el parámetro principal.
LYMNWT	Estimación no–paramétrica e intervalo de confianza para el parámetro principal basado en el test estadístico de Mann-Whitney.
LIINDW	Inversa de la aproximación de la distribución nula del test de Mann-Whitney.

Constantes

LIBETH	Cálculo de $\beta = \int \chi(s)d\Phi(s)$, donde $\chi = \psi_d^2/2$ y ψ_d es la función de Huber.
LIBETU	Cálculo de $\beta = \int \chi(s)d\Phi(s)$ donde χ es una función externa dada.
LIBETO	Cálculo de $\beta_0 = \Phi^{-1}(3/4)$.
LIEPSH	Cálculo de $\int \psi^2(s)d\Phi(s)$ y $\int \psi'(s)d\Phi(s)$ cuando ψ es la función de Huber.
LIEPSU	Cálculo de $\int \psi^2(s)d\Phi(s)$ y $\int \psi'(s)d\Phi(s)$ cuando ψ es una función externa.

1.6 Ejemplos con S-PLUS y subrutinas del LCMAIN

1.6.1 Subrutina LICLLS

Estimación clásica de la media y el desvío estándar.

Nombre del archivo en Microsoft Word: T-LICLLS.

Estimación clásica de la media aritmética $\widehat{\theta}_{LS}$ y el desvío estándar empírico $\widehat{\sigma}_{LS}$ de

acuerdo con (1.1) y (1.2). Estimación de la varianza $(\widehat{\sigma}^2_{LS})/n$ para $\widehat{\theta}_{LS}$.

Por medio de la subrutina TQUANT se obtiene un intervalo de confianza

de la forma (1.3) y los residuales $y_i - \widehat{\theta}_{LS}$. Ejemplo: y=(5,2,-1,-2).

library ("robeth",T)

Ingresar las observaciones y cantidad de observaciones.

y <- c(5,2,-1,-2); n <- 4

s <- liclls(y)

media <- s$theta; desvío <- s$sigma; residuales <- s$rs

varpromedio <- (desvío * desvío)/n

La media, el desvío y los residuales son:

media; desvío; residuales

La estimación de la varianza del vector promedio es:

varpromedio

Intervalo de confianza de la forma (1.3) con probabilidad 1-2alfa.

Ingresar p tal P(t<t0)=p con 0<p<1.

p <- 0.975; gl <- n-1; s <- tquant(p,gl); t0 <- s$x; raizn <- sqrt(n)

linferior <- media -t0*(desvio/raizn); lsuperior <- media +t0*(desvio/raizn)

El valor de t0 y los valores de los extremos del I.C. son:

t0; linferior; lsuperior

rm(y,n,media,desvío,varpromedio,residuales,gl,p,t0,raizn,linferior,lsuperior)

Salida en S-PLUS

> # La media, el desvío y los residuales son:

> media; desvío; residuales

[1] 1

> [1] 3.162278

> [1] 4 1 -2 -3

> # La estimación de la varianza del vector promedio es:

> varpromedio

[1] 2.5

> # El valor de t0 y los valores de los extremos del I.C. son:

> t0; linferior; lsuperior

[1] 3.18245

> [1] -4.031896

> [1] 6.031896

1.6.2 Subrutina LILARS (n impar)

Mediana y desviación absoluta mediana.

Nombre del archivo en Microsoft Word: T-LILARS-impar.

Propósito: La subrutina realiza los cálculos de la mediana $\widehat{\theta}_{MED}$ y la

estimación $\widehat{\sigma}_{MAD}$ de σ de acuerdo a (1.4), (1.5), (1.6), (1.7) y (1.8). La

varianza de $\widehat{\theta}_{MED}$ es estimada de acuerdo a (1.10). Usando la subrutina

QUANT se obtiene un intervalo de confianza según (1.11).

Esta subrutina proporciona también la muestra ordenada: $y_{[1]} <= ... <= y_{[n]}$.

Método: Las observaciones son ordenadas en magnitud creciente.

Si n es impar, $\widehat{\theta}_{MED}$ es la observación del medio. Si n es par, $\widehat{\theta}_{MED}$ es la

media de las dos observaciones del medio.

Para el cálculo de $MAD = \underset{i}{med}\ (\mid y_i - \widehat{\theta}_{MED} \mid)$, las diferencias absolutas

$\mid y_i - \widehat{\theta}_{MED} \mid$ se examinan en orden creciente. Esto se expresa observando que,

para n impar, $0 < y_{[\frac{n+1}{2}+1]} - \widehat{\theta}_{MED} <= y_{[\frac{n+1}{2}+2]} - \widehat{\theta}_{MED} < ... <= y_{[n]} - \widehat{\theta}_{MED}$

y $0 < \widehat{\theta}_{MED} - y_{[\frac{n+1}{2}-1]} <= \widehat{\theta}_{MED} - y_{[\frac{n+1}{2}-2]} < ... < \widehat{\theta}_{MED} - y_{[1]}$.

El caso para n par es tratado en forma similar.

Si n es impar, MAD es el valor del medio de las desviaciones absolutas.

Si n es par, MAD es estimado por el más chico de los dos valores medios de las

desviaciones absolutas.

Ejemplo con un número impar de observaciones

library ("robeth",T)

Ingresar las observaciones.

y <- c(-2,-7,-8,-1,1,2,3,5,10)

s <- lilars(y)

```
mediana <- s$theta; desvío <- s$sigma ; MAD <- s$xmad
varianza <- s$var; residuales <- s$rs
# La mediana (1.4), desvío (1.6), MAD (1.7), varianza (1.10) son:
mediana; desvío; MAD; varianza
# Vector y, los residuales correspondientes y el vector ordenado son:
yorden <- s$y;    y; residuales; yorden
# Intervalo de confianza de la forma (1.11) con probabilidad 1-2alfa.
# Ingresar p tal P(Z<z0)=p con 0<p<1.
p <- 0.975; s <- quant(p); z0 <- s$x
inferior <- mediana -z0*(sqrt(varianza));    superior <- mediana +z0*(sqrt(varianza))
# El valor de z0 y los extremos del I.C. son:
z0; inferior; superior
rm(y,yorden,mediana,MAD,desvío,varianza,residuales,p,z0,inferior,superior)
```

Salida en S-PLUS

```
> # La mediana (1.4), desvío (1.6), MAD (1.7), varianza (1.10) son:
> mediana; desvio; MAD; varianza
[1] 1
> [1] 4.447739
> [1] 3
> [1] 3.452677
> # Vector y, los residuales correspondientes y el vector ordenado son:
> y; residuales; yorden
[1] -2 -7 -8 -1 1 2 3 5 10
> [1] -3 -8 -9 -2 0 1 2 4 9
> [1] -8 -7 -2 -1 1 2 3 5 10
> # El valor de z0 y los extremos del I.C. son:
> z0; inferior; superior
[1] 1.960395
> [1] -2.642686
> [1] 4.642686
```

1.6.3 Subrutina LILARS (n par)

```
# Mediana y desviación absoluta mediana.
  # Nombre del archivo en Microsoft Word: T-LILARS-par.
  # Ejemplo con un número par de observaciones
   library ("robeth",T)
  # Ingresar las observaciones.
  y <- c(-40,-2,-10,20)
  s <- lilars(y)
  mediana <- s$theta; desvío <- s$sigma ; MAD <- s$xmad
  varianza <- s$var; residuales <- s$rs
  # La mediana (1.4), desvío (1.6), MAD (1.7), varianza (1.10) son:
  mediana; desvio; MAD; varianza
  # Vector y, los residuales correspondientes y el vector ordenado son:
  yorden <- s$y;    y; residuales; yorden
  # Intervalo de confianza de la forma (1.11) con probabilidad 1-2alfa.
  # Ingresar p tal P(Z<z0)=p con 0<p<1.
  p <- 0.975; s <- quant(p); z0 <- s$x
  inferior <- mediana -z0*(sqrt(varianza));    superior <- mediana +z0*(sqrt(varianza))
  # Valor de z0 y los extremos del I.C.:
  z0; inferior; superior
  rm(y,yorden,mediana,MAD,desvío,varianza,residuales,p,z0,inferior,superior)
```

Salida en S-PLUS

```
  > # La mediana (1.4), desvío (1.6), MAD (1.7), varianza (1.10) son:
  > mediana; desvio; MAD; varianza
  [1] -6
  > [1] 5.930319
  > [1] 4
  > [1] 13.81071
  > # Vector y, los residuales correspondientes y el vector ordenado son:
```

> y; residuales; yorden

[1] -40 -2 -10 20

> [1] -34 4 -4 26

> [1] -40 -10 -2 20

> # Valor de z0 y los extremos del I.C.:

> z0; inferior; superior

[1] 1.960395

> [1] -13.28537

> [1] 1.285371

1.6.4 Subrutina LYHALG

M-estimador de locación con estimación simultánea de escala.

Nombre del archivo en Microsoft Word: T-LYHALG.

Resolución simultánea de los sistemas (1.12) y (1.13) para θ y σ.

En esta subrutinaΨ y χ son funciones definidas por el usuario. Opcionalmente, la

segunda ecuación se puede omitir y la primera se resulve para θ usando un valor

asignado para σ. Alternativamente σ, se puede estimar de acuerdo con (1.14).

LYHALG da la estimación según (1.18) de la varianza del $\widehat{\theta}$ estimado.

Por medio de la subrutina QUANT se obtiene un intervalo de

confianza de la forma (1.19) para θ.

Ejemplo, datos del Lehmann, p. 183 [16].

library ("robeth",T)

Ingresar las observaciones y cantidad de observaciones.

y <- c(6.0,7.0,5.0,10.5,8.5,3.5,6.1,4.0,4.6,4.5,5.9,6.5); n <- 12

dfvals()

dfrpar (as.matrix(y),"huber")

libeth()

s <- lilars(y); t0 <- s$theta; s0 <- s$sigma; s <- lyhalg(y=y, theta=t0, sigmai=s0)

tm <- s$theta; vartm <- s$var; sigmaesti <- s$sigmaf

El M-estimador (tm) de locación y de escala (sigmaesti) son:

tm; sigmaesti

```
# La varianza de tm es:
vartm
# los residuales son:
residuales <- s$rs;  residuales
# Intervalo de confianza de la forma (1.19) con probabilidad 1-2alfa.
# Ingresar p tal P(Z<z0)=p con 0<p<1.
p <- 0.975;  s <- quant(p); linferior <- tm -s$x*sqrt(vartm)
lsuperior <- tm + s$x*sqrt(vartm)
# Los valores de los extremos del I.C.:
linferior; lsuperior
rm(y,n,tm,vartm,sigmaesti,t0,s0,residuales,p,linferior,lsuperior)
```

Salida en S-PLUS

```
# El M-estimador (tm) de locación y de escala (sigmaesti) son:
> tm; sigmaesti
[1] 5.808991
[1] 1.855316
> # La varianza de tm es:
> vartm
[1] 0.2931536
> # los residuales son:
> residuales
 [1] 0.19100857 1.19100857 -0.80899143 4.69100857 2.69100857 -2.30899143
 [7] 0.29100847 -1.80899143 -1.20899153 -1.30899143 0.09100866 0.69100857
> # Los valores de los extremos del I.C.:
> linferior; lsuperior
[1] 4.747561
[1] 6.870421
```

1.6.5 Subrutina LYHDLE

Nombre del archivo en Microsoft Word: T-LYHDLE.

```
# Estimador de Hodges-Lehman e intervalo de confianza para el centro de simetría
# basado en el test estadístico de Wilcoxon para una muestra.
# Si y_1, y_2, ..., y_n representan las n observaciones. Si m=n(n+1)/2, notamos con
# a_[1] <= ... <= a_[m] los m promedios ordenados (y_i + y_j)/2 para i <=j.
# La subrutina LYHDLE da el valor a_[k] correspondiente a un valor específico de k.
# Este valor k es usado para obtener el estimador de Hodges-Lehmann
# dado por (1.26) y obtener un intervalo de confianza de la forma (1.31).
# El índice k con coeficiente de confianza para un intervalo de la forma (1.31) se
# puede obtener de la subrutina LIINDH.
# Ejemplo: (n par), datos del Lehmann, p. 183 [16].
library ("robeth",T)
# Ingresar las observaciones y cantidad de observaciones.
 y <- c(6.0,7.0,5.0,10.5,8.5,3.5,6.1,4.0,4.6,4.5,5.9,6.5);  n <- 12
dfvals( )
dfrpar (as.matrix(y), "huber")
libeth( )
m <- n*(n+1)/2. ;  k1 <- m/2;  k2 <- k1 +1
z1 <- lyhdle(y=y,k=k1);   z2 <- lyhdle(y=y,k=k2)
th <- (z1$hdle + z2$hdle)/2 ;   ku <- liindh(0.95,n);  ku
kl <- liindh(0.05,n);  kl
linferior <- lyhdle(y=y, k=kl$k);  lsuperior <- lyhdle(y=y, k=ku$k)
# Valor de th y valores de los extremos del I.C.:
th ; linferior; lsuperior
rm(y,n,m,k1,k2,z1,z2,th,ku,kl,lsuperior,linferior)
```

Salida en S-PLUS

```
# Valor de th y valores de los extremos del I.C.:
[1] 5.849874
[1] 4.999653
[1] 6.999674
```

1.6.6 Subrutina LIINDS

Nombre del archivo en Microsoft Word: T-LIINDS.

Propósito: la subrutina LIINDS calcula el entero k tal que $\Phi(\frac{2k-n-1}{\sqrt{n}})$ está lo más

próximo posible a un valor $\alpha \in (0,1)$. Ejemplo: n=12, alpha=0.05 y alpha=0.95.

library ("robeth",T)

Ingresar cantidad de observaciones y alpha (0<alpha<1).

n <- 12; alpha <- 0.05; s <- liinds(alpha,n)

valork <- s$k; valoralpha1 <- s$alpha1

El valor de k y alpha1:

valork; valoralpha1

rm(alpha,valork,valoralpha1)

alpha <- 0.95; s <- liinds(alpha,n); valork <- s$k; valoralpha1 <- s$alpha1

El valor de k y alpha1:

valork; valoralpha1

rm(n,alpha,valork,valoralpha1)

Salida en S-PLUS

> # El valor de k y alpha1 para 0.05 y 0.95 son respectivamente:

[1] 4

[1] 0.07445732

[1] 9

[1] 0.9255427

1.6.7 Subrutina LIINDH

Nombre del archivo en Microsoft Word: T-LIINDH.

Propósito: sea m=n(n+1)/2, la subrutina LIINDH calcula el entero k tal

que $\Phi(\frac{2k-m-1}{\sqrt{m(2n+1)/3}})$ está lo más próximo posible a un valor $\alpha \in (0,1)$.

library ("robeth",T)

Ejemplo: n=12, alpha=0.05 y alpha=0.95.

Ingresar cantidad de observaciones y alpha (0<alpha<1).

```
n <- 12;  alpha <- 0.05
s <- liindh(alpha,n);   valork <- s$k;   valoralpha1 <- s$alpha1
# El valor de k y alpha1:
valork; valoralpha1
rm(alpha,valork,valoralpha1)
alpha <- 0.95;   s <- liindh(alpha,n);  valork <- s$k;   valoralpha1 <- s$alpha1
# El valor de k y alpha1:
valork; valoralpha1
rm(n,alpha,valork,valoralpha1)
```

Salida en S-PLUS

```
> # El valor de k y alpha1 para 0.05 y 0.95 son respectivamente:
> valork; valoralpha1
[1] 19
[1] 0.05390089
[1] 60
[1] 0.9460991
```

1.6.8 Subrutina LITTST

```
# Cálculo del test-t para el parámetro principal.
    # Nombre del archivo en Microsoft Word: T-LITTST.
    # Calcula las estimaciones dadas en (1.32), (1.33), (1.34), (1.35) y (1.36).
    # Calcula P(T>t), donde T es la variable aleatoria t con (m+n-2) grados de libertad.
    # Límites de confianza: (1.37a) y (1,37b) para el parámetro principal.
    library ("robeth",T)
    # Ejemplo: ingresar los elementos del primer vector de dimensión m.
    y <- c(1,2,6)
    # Ingresar dimensión m.
    m <- 3
    # Ingresar los elementos del segundo vector de dimensión n.
    x <- c(2,3,11,4)
```

```
# Ingresar dimensión n
n <- 4
# Ingresar alpha (el coeficiente de confianza es 1-2alpha)
alpha <- 0.025;   s <- littst(x,y,alpha);   soldelta <- s$delta
sols1<- s$s1;   sols2 <- s$s2;   solsigma <- s$sigma;   inferior <- s$tl
superior <- s$tu;   raíz <- sqrt (1.0/m + 1.0/n);   raíz
te <- soldelta/(solsigma*raíz);   valor <- s$p
# Las estimaciones dadas en (1.32), (1.33), (1.34), (1.35) y (1.36):
soldelta;  sols1;  sols2;   solsigma;   te
# Los límites inferior y superior del intervalo de confianza y el valor de P(T>t):
inferior;   superior;   valor
rm(x,y,n,m,alpha,soldelta,sols1,sols2,solsigma,te,raíz,valor,inferior,superior)
```

Salida en S-PLUS

```
> # Las estimaciones dadas en (1.32), (1.33), (1.34), (1.35) y (1.36) son:
> soldelta
[1] -2
> sols1
[1] 4.082483
> sols2
[1] 2.645751
> solsigma
[1] 3.577709
> te
[1] -0.7319251
> # Los lmites inferior y superior del intervalo de confianza y el valor de P:
> inferior
[1] -9.026434
> superior
[1] 5.026434
> valor P(T>t)
```

[1] 0.7514682

1.6.9 Subrutina LYTAU2

Nombre del archivo en Microsoft Word: T-LYTAU2.

Tau-test para el parámetro principal y cálculo del P-valor (pe).

Estimación (tm) e intervalo de confianza (tl,tu))

Ejemplo: datos del Lehmann, p. 92 [16].

library ("robeth",T)

Ingresar observaciones del primer vector X de dimensión m y valor de m.

x <- c(7.7,5.0,1.7,0.0,-3.0,-3.1,-10.5); m <- 7

Ingresar observaciones del segundo vector Y de dimensión n y valor de n.

y <- c(17.9,13.3,10.6,7.6,5.7,5.6,5.4,3.3,3.1,0.9); n <- 10

z <- c(x,y)

dfrpar (as.matrix(z), "huber")

libeth( )

s <- lytau2(z=z,m=m,n=n); pe <- s$p

Estimación (tm) e intervalo de confianza (tl,tu).

tm <- s$deltal; tm; c22<- s$cov[3]; c22; s <- quant(0.975)

tl <- tm-s$x*sqrt(c22); tu <- tm+s$x*sqrt(c22); tl; tu; pe

rm(x,y,n,m,z,tm,c22,tl,tu,pe)

Salida en S-PLUS

> # Estimación (tm) e intervalo de confianza (tl,tu)

[1] 6.91759

> tl; tu

[1] 1.562069

[1] 12.27311

> pe

[1] 0.01402915

1.6.10 Subrutina LYMNWT

Nombre del archivo en Microsoft Word: T-LYMNWT.

Estimador no paramétrico e intervalo de confianza para el parámetro

principal basado en el test estadístico de Mann-Whitney.

Si $x_1, x_2, \cdots, x_m$ y $y_1, y_2, \cdots, y_n$ son dos muestras de n y m observaciones.

Si $d_{[1]} < d_{[2]} < .... < d_{[m*n]}$ denota las $m \cdot n$ diferencias ordenadas $(y_j - x_i)$ la

subrutina LYMNWT da el valor $d_{[k]}$ correspondiente a un valor específico de k.

El valor k es usado para la estimación dada en (1.49) y el I.C. dado por (1.53).

Ejemplo: datos del Lehmann, p. 92 [16].

library ("robeth",T)

Ingresar observaciones del primer vector X de dimensión m y valor de m.

x <- c(7.7,5.0,1.7,0.0,-3.0,-3.1,-10.5); m <- 7

Ingresar observaciones del segundo vector Y de dimensión n y valor de n.

y <- c(17.9,13.3,10.6,7.6,5.7,5.6,5.4,3.3,3.1,0.9); n <- 10

dfvals( )

dfrpar (as.matrix(y),"huber")

libeth( )

k1 <- m*n/2; k2 <- k1+1

s1 <- lymnwt(x=x,y=y,k=k1); s2 <- lymnwt(x=x,y=y,k=k2)

dm <- (s1$tmnwt+s2$tmnwt)/2.0

sl <- liindw(0.05,m,n); kl <- sl$k

s <- lymnwt(x=x,y=y,k=kl); linferior <- s$tmnwt

s <- lymnwt(x=x,y=y,k=m*n-kl+1); lsuperior <- s$tmnwt

La estimación dada en (1.49) y el I.C. del 90% son:

dm; linferior; lsuperior

rm(x,y,n,m,k1,k2,dm,lsuperior,linferior)

Salida en S-PLUS

> # La estimación dada en (1.49) y los límites del I.C. del 90 % son:

> dm

[1] 6.349895

> linferior

[1] 2.900083

> lsuperior

[1] 12.89972

1.6.11 Subrutina LIINDW

Nombre del archivo en Microsoft Word: T-LIINDW.

Propósito: la subrutina LIINDW calcula el entero k tal que $\Phi(\frac{2k-mn-1}{\sqrt{mn(m+n+1)/3}})$

está lo más próximo posible a un valor $\alpha \in (0,1)$.

library ("robeth",T)

Ejemplo: m=7, n=10 alpha=0.05.

Ingresar m, cantidad de observaciones del primer vector X.

m <- 7

Ingresar m, cantidad de observaciones del segundo vector Y.

n <- 10

Ingresar alpha (0<alpha<1).

alpha <- 0.05; s <- liindw(alpha,m,n); valork <- s$k

valoralpha1 <- s$alpha1

Los valores de k y alpha1 son:

valork; valoralpha1

rm(m,n,alpha,valork,valoralpha1)

Salida en S-PLUS

> alpha <- 0.05

> # Los valores de k y alpha1 son:

> valork; valoralpha1

[1] 19

[1] 0.05367325

Capítulo 2

M-estimadores de los Coeficientes y del Parámetro de Escala en Regresión Lineal

En esta parte se presentaran procedimientos incluidos en el programa de computación ROBETH para calcular soluciones robustas de los problemas de regresión lineal basadas en M-estimadores. El conjunto de subrutinas a usar estan agrupadas con el nombre de RGMAIN.

2.1 Introducción

Sea $X = (x_{ij})$ la matriz real de n x p, $y = (y_1, y_2, \cdots, y_n)^T$ el vector respuesta de n componentes, y $\theta = (\theta_1, \theta_2, \cdots, \theta_p)^T$ un p-vector de parámetros desconocidos o coeficientes cuyas componentes se desean estimar. La matriz X se llama matriz de diseño. Consideremos el modelo lineal clásico, según Seber [28]

$$\Omega : \quad y = X\theta + e,$$

donde e = $(e_1, e_2, \cdots, e_n)^T$ es un n-vector de errores desconocidos. Se asume que para un X dado las componentes e_i de e son independientes e idénticamente distribuidos de acuerdo a la distribución $L(\cdot/\sigma)$, donde σ es el parámetro de escala (generalmente desconocido). Usualmente $L(\cdot/\sigma) \approx \Phi(\cdot)$, la distribución normal estándar con densidad $\phi(s) = (1/\sqrt{2\pi})\exp(-s^2/2)$.

Notaremos con $r = (r_1, r_2, \cdots, r_n)^T = y - X\theta$ el n-vector de residuales para un valor dado de θ y con x_i^T la i-ésima fila de la matriz X. Con la notación $\mathrm{X}\theta \approxeq y$ queremos indicar el problema de regresión para estimar θ en el modelo Ω. Las referencias principales para los métodos implementados en RGMAIN se encuentran en Huber [10], Hampel et al. [8], Bustos [2] y en Rousseeuw [26].

2.2 M-estimadores de regresión

2.2.1 El problema de mínimos cuadrados

El clásico *estimador de mínimos cuadrados* $\widehat{\theta}_{LS}$ de θ, según Neter et al. [21] se obtiene como solución del problema

$$Q_{LS}(\theta) = mín, \tag{2.1}$$

donde

$$Q_{LS}(\theta) = \frac{1}{2} \sum_{i=1}^{n} {r_i}^2 \ . \tag{2.2}$$

Tomando las derivadas parciales de Q_{LS} con respecto a las componentes de θ e igualando a cero, se obtienen las ecuaciones normales

$$X^T X\theta = X^T y \ . \tag{2.3}$$

Si el $rango(X) = p$, la solución de (2.3) es

$$\widehat{\theta}_{LS} = (X^T X)^{-1} X^T y \ . \tag{2.4}$$

El estimador de mínimos cuadrados es el estimador de máxima verosimilitud cuando $L(\cdot/\sigma) = \Phi(\cdot)$. En este caso, el estimador usual del parámetro de escala σ es

$$\widehat{\sigma} := \sqrt{\frac{1}{(n-p).\beta} Q_{LS}(\widehat{\theta})} \quad \text{con } \beta = 1/2. \tag{2.5}$$

2.2.2 El problema del mínimo del valor absoluto de los residuales

Una estimación alternativa $\widehat{\theta}_{LAR}$ de θ se obtiene como solución de

$$Q_{LAR}(\theta) = mín, \tag{2.6}$$

donde

$$Q_{LAR}(\theta) = \sum_{i=1}^{n} \mid r_i \mid . \tag{2.7}$$

A $\widehat{\theta}_{LAR}$ se lo llama *mínimo de residuales absolutos* o la L_1-*norma estimación* de θ. Es el estimador de máxima verosimilitud cuando $L(\cdot/\sigma)$ es la distribución doble exponencial y tiene algunas propiedades de robustez. Una estimación robusta de σ que usualmente combina con $\widehat{\theta}_{LAR}$ es la *desviación absoluta mediana* (ver Huber, p. 175 [10])

$$\widehat{\sigma} := \frac{\underset{i}{med} \left\{ \mid y_i - x_i^T \widehat{\theta}_{LAR} \mid \right\}}{\beta_0}, \tag{2.8}$$

donde β_0 es una constante adecuada tal que $\widehat{\sigma}$ es asintóticamente insesgada (consistente) cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

2.2.3 El problema del mínimo de los cuadrados pesados

Sea $W = diag(w_i)$ la matriz diagonal n x n de los *pesos* positivos de las observaciones (x_i^T, y_i). El *estimador de mínimos cuadrados pesado* $\widehat{\theta}_{WLS}$ de θ se obtiene como solución del problema

$$Q_{WLS}(\theta) = mín, \tag{2.9}$$

donde

$$Q_{WLS}(\theta) = \frac{1}{2} \sum_{i=1}^{n} w_i \cdot {r_i}^2 . \tag{2.10}$$

Las ecuaciones normales son

$$X^T W X \theta = X^T W y . \tag{2.11}$$

Si el $rango(X^T WX) = p$, la solución de (2.11) es

$$\widehat{\theta}_{WLS} = (X^T WX)^{-1} X^T Wy \ . \tag{2.12}$$

Una estimación asociada de σ es

$$\widehat{\sigma} := \sqrt{\frac{1}{(n-p).\beta} Q_{WLS}(\widehat{\theta})} \ , \tag{2.13}$$

donde β es una constante adecuada tal que $\widehat{\sigma}$ es asintóticamente insesgado (consistente) cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

2.2.4 El problema del mínimo del valor absoluto de los residuales pesados

Sea $W = diag(w_i)$ la matriz diagonal n x n de los *pesos* positivos de las observaciones (x_i^T, y_i). El *estimador del mínimo del valor absoluto de los residuales pesados* $\widehat{Q}_{WLAR}$ de θ se obtiene como solución del problema

$$Q_{WLAR}(\theta) = mín, \tag{2.14}$$

donde

$$Q_{WLAR}(\theta) = \sum_{i=1}^{n} w_i \mid r_i \mid \ . \tag{2.15}$$

Una estimación robusta de σ, que se puede combinar con $\widehat{Q}_{WLAR}$ es

$$\widehat{\sigma} := \frac{\underset{i}{med} \left\{ w_i \mid y_i - x_i^T \widehat{\theta}_{WLAR} \mid \right\}}{\beta_0}, \tag{2.16}$$

donde β_0 es una constante adecuada tal que $\widehat{\sigma}$ es asintóticamente insesgado cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

2.2.5 El problema de regresión de tipo Huber

Si σ es conocido, el *estimador de tipo Huber* $\widehat{\theta}_H$ de θ se obtiene como solución de un

sistema de p equaciones

$$\sum_{i=1}^{n} \Psi(\frac{r_i}{\sigma}) x_{ij} = 0, \text{ para } j = 1, \cdots, p, \tag{2.17}$$

donde Ψ es una función definida por el usuario en forma adecuada y $r = y - X\theta$. En estas ecuaciones el parámetro de escala σ se supone conocido. Sin embargo, σ es usualmente desconocido, en este caso ROBETH da tres opciones para estimar a σ:

1. Usar una estimación preliminar, por ejemplo la estimación definida en (2.8) (ver Huber, pp. 175-176 [10]). Alternativamente, se puede usar la estimación puntual de σ para alto punto de ruptura.

2. Resolver el sistema (2.17) y la ecuación suplementaria (para σ):

$$\frac{1}{n-p} \cdot \sum_{i=1}^{n} \chi(\frac{r_i}{\sigma}) = \beta. \tag{2.18}$$

   Aquí, χ es otra función adecuada (ver Hampel et al., p. 312 [8]).

3. Resolver el sistema (2.17) y la ecuación suplementaria:

$$\sigma = \frac{\underset{i}{med} \mid y_i - x_i^T \theta \mid}{\beta_0} \tag{2.19}$$

   (ver Hampel et al., p. 312 [8]).

En (2.18) y en (2.19) las constantes β y β_0 son elegidas tal que la solución $\widehat{\sigma}$ sea asintóticamente consistente cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

2.2.6 El problema de regresión de tipo Mallows

Si σ es conocido, el *estimador de tipo Mallows* $\widehat{\theta}_M$ de θ se obtiene como solución del sistema

$$\sum_{i=1}^{n} \Psi(\frac{r_i}{\sigma}) w_i x_{ij} = 0, \text{ para } j = 1, \cdots, p, \tag{2.20}$$

donde Ψ es una función definida por el usuario y w_i $(i = 1, \cdots, n)$ son pesos positivos dados. Si σ es desconocido, en este caso ROBETH da tres opciones para estimar a σ:

1. Usar una estimación preliminar, por ejemplo la estimación definida en (2.16) o un estimador con alto punto de ruptura.

2. Resolver el sistema (2.20) y la ecuación suplementaria (para σ):

$$\frac{1}{n-p} \cdot \sum_{i=1}^{n} \chi(\frac{r_i}{\sigma}) \cdot w_i = \beta. \tag{2.21}$$

   Aquí, χ es otra función adecuada.

3. Resolver el sistema (2.20) y la ecuación suplementaria:

$$\sigma = \frac{\underset{i}{med} \mid \sqrt{w_i}(y_i - x_i^T\theta) \mid}{\beta_0}. \tag{2.22}$$

   En (2.21) y en (2.22), las constantes β y β_0 son elegidas tal que la solución $\widehat{\sigma}$ sea asintóticamente consistente cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

2.2.7 El problema de regresión de tipo Schweppe

Si σ es conocido, el *estimador de tipo Schweppe* $\widehat{\theta}_S$ de θ se obtiene como solución del sistema

$$\sum_{i=1}^{n} \Psi(\frac{r_i}{\sigma \cdot w_i}) w_i \cdot x_{ij} = 0, \text{ para } j = 1, \cdots, p, \tag{2.23}$$

donde Ψ es una función definida por el usuario y w_i $(i = 1, \cdots, n)$ son pesos positivos dados. Usualmente el parámetro de escala σ es desconocido, en este caso ROBETH da tres opciones para estimar a σ:

1. Usar una estimación preliminar, por ejemplo la estimación definida en (2.16) o un estimador con alto punto de ruptura.

2. Resolver el sistema (2.23) y la ecuación suplementaria (para σ):

$$\frac{1}{n-p} \cdot \sum_{i=1}^{n} \chi(\frac{r_i}{\sigma \cdot w_i}) \cdot w_i^2 = \beta. \tag{2.24}$$

   Aquí, χ es otra función adecuada.

3. Resolver el sistema (2.23) y la ecuación suplementaria:

$$\sigma = \frac{\underset{i}{med} \mid y_i - x_i^T \theta \mid}{\beta_0}. \tag{2.25}$$

En (2.24) y en (2.25) las constantes β y β_0 son elegidas tal que la estimación de σ sea asintóticamente insesgada cuando $L(\cdot/\sigma) = \Phi(\cdot)$ (ver observación 4).

Observación 1. Varias elecciones de las funciones Ψ y χ son consideradas en Huber, Cap. 4 [10], en Hampel et al., Cap. 6 [8] y en Bunke and Bunke, p. 146 [1]. Usualmente estas funciones deben satisfacer determinadas condiciones para cumplir con la existencia y la unicidad de la solución y convergencia de los algoritmos iterativos usados para su cálculo (ver Huber, p. 138 [10]). Una condición suficiente conocida es que

$$\Psi(s) = \rho'(s) \quad \text{y que } \chi(s) = s.\psi(s) - \rho(s), \tag{2.26}$$

donde ρ es una función positiva y convexa tal que $0 < \underset{|s|\to\infty}{lím}\ \rho(s)/ \mid s \mid = c <= \infty$ y $\rho(0) = 0$. El caso más importante especial que cumple con (2.26) es $\psi(s) = \psi_{_C}(s)$ y $\chi(s) = \psi_{_C}^2(s)/2$,

$$\Psi_{_C}(s) := máx\,[-c, mín(c, s)] \tag{2.27a}$$

es la función de Huber. En este caso $\rho(s) = \rho_c(s)$, donde

$$\rho_c(s) := \begin{cases} c \mid s \mid -c^2/2 & \text{si } \mid s \mid > c \\ s^2/2 & \text{en el resto,} \end{cases} \tag{2.27b}$$

y la constante c es una constante adecuada que el usuario especifica para obtener un compromiso satisfactorio entre resistencia y eficiencia para errores gaussianos. Sin embargo, $\Psi-$funciones decrecientes que no cumplen con (2.26) son también de interés, (ver Hampel et al., p. 149 [8]). Un conocido ejemplo son las funciones de Tukey $\Psi(s) = s \cdot \left[máx(1 - s^2, 0)\right]^2$. El software ROBETH proporciona una lista de funciones FORTRAN tal que el usuario podrá seleccionar pares particulares de funciones Ψ y χ.

Observación 2. Notemos que el estimador de mínimos cuadrados de θ y de σ definido en (2.1)-(2.2) y (2.5) se puede obtener de (2.17) y (2.18) usando $\psi(s) = s$, $\chi(s) = s^2/2$,

y $\beta = 1/2$. Notemos también que este es el caso límite de $\widehat{\theta}_H$ obtenido cuando

$$\Psi(s) = \Psi_c(s), \quad \chi(s) = \chi_c(s), \text{ para } \quad c \to \infty.$$

Similarmente el estimador $\widehat{\theta}_{LAR}$ se puede considerar un caso especial de (2.17) usando $\Psi(s) = sign(s)$. En realidad, todos los estimadores considerados en esta sección pertenecen a la clase de los *M-estimadores generales* de regresión (ver Hampel et al., Cap. 6 [8]).

Observación 3. Varios tipos de pesos "óptimos" se proponen en el capítulo 3 del libro de Marazzi and Randriamiharisoa [18]. Usualmente ellos son de la forma $w_i = \widetilde{w}(x_i)$ donde $\widetilde{w}$ es una función decreciente de las distancias $\mathrm{d}_i = (x_i^T \cdot \widehat{A}^T \cdot \widehat{A} x_i)^{1/2}$ de los x_i desde el orígen y $\widehat{A}$ es una transformación matricial tal que

$$I_p = \frac{1}{n} \sum_{i=1}^{n} u(\mid \widehat{A} x_i \mid) \widehat{A} x_i \cdot x_i^T \cdot \widehat{A}^T. \tag{2.28}$$

donde u y $\widetilde{w}$ son funciones dadas y I_p es la p x p matriz identidad. La elección de las funciones u, $\widetilde{w}$, ψ, y χ depende de las propiedades requeridas para los estimadores. Muchas opciones se dan en el Cap 3. del libro de Marazzi and Randriamiharisoa [18], junto con los algoritmos para la resolución de (2.28). El cálculo de los pesos es un paso preliminar para la resolución de los problemas de regresión de tipo Mallows y de Schweppe.

Observación 4. Por medio de ROBETH se pueden determinar las constantes β y β_0 para distintas estimaciones $\widehat{\sigma}$ de σ que sean asintóticamente insesgadas para errores normales. En otras palabras, se usará:

$\beta = 1/2$	en el caso de mínimos cuadrados,
$\beta = (\sum_{i=1}^{n} w_i) \diagup (2n)$	en el caso de mínimos cuadrados pesados,
$\beta = \int \chi(s) d\Phi(s)$	en el caso de Huber,
$\beta = (\sum_{i=1}^{n} w_i \int \chi(s) d\Phi(s)) \diagup n$	en el caso de Mallows,
$\beta = (\sum_{i=1}^{n} [w_i^2 \int \chi(s/w_i) d\Phi(s)]) \diagup n$	en el caso de Schweppe,
$\beta_0 = \Phi^{-1}(0.75)$	en los casos de Huber, de Schweppe y de L_1-norma
$\beta_0 = F_1^{-1}(0.75)$	en el caso de Mallows,
$\beta_0 = F_2^{-1}(0.75)$	en el caso de la L_1-norma pesada.

Aquí, F_1 y F_2 son las distribuciones acumulativas de $(r_i \cdot \sqrt{w_i})$ y $(r_i \cdot w_i)$ respectiva-

mente. Ellas son estimadas por $F_1(s) = (1/n) \sum \Phi(s/\sqrt{w_i})$ y por $F_2(s) = (1/n) \sum \Phi(s/w_i)$.

2.3 Procedimientos numéricos para problemas de tipo Mallows

2.3.1 Reducción de los problemas

Los cálculos de los estimadores de mínimos cuadrados pesados definidos en (2.9)-(2.10) y (2.13) se pueden reducir al cálculo de los estimadores de mínimos cuadrados de la siguiente forma: Sea

$$\widetilde{w}_i = \sqrt{w_i}, \qquad \widetilde{y}_i = \widetilde{w}_i \cdot y_i \ , \qquad \widetilde{x}_i = \ \widetilde{w}_i \cdot x_i \tag{2.29}$$

y resolver el problema de mínimos cuadrados, donde x_i e y_i son reemplazados por $\widetilde{x}_i$ e $\widetilde{y}_i$. Por esta razón, ROBETH no construye subrutinas especiales para el caso de mínimos cuadrados pesados.

Los cálculos de los estimadores del mínimo del valor absoluto de los residuales pesados definidos en (2.14)-(2.15) y (2.16) se pueden reducir al cálculo de los estimadores del mínimo de los valores absolutos de los residuales en el siguiente sentido: Sea

$$\widehat{w}_i \ = w_i \ , \qquad \widehat{y}_i \ = \widehat{w}_i \cdot y_i \ , \qquad \widehat{x}_i \ = \widehat{w}_i \cdot x_i \tag{2.30}$$

y resolver el problema del mínimo del valor absoluto de los residuales, donde, x_i e y_i son reemplazados por $\widehat{x}_i$ e $\widehat{y}_i$. Por esta razón, ROBETH no construye subrutinas especiales para la solución del mínimo del valor absoluto de los residuales pesados. La trasformación (2.29) reduce el problema tipo Mallows (2.20)-(2.21) al problema tipo Schweppe (es decir, resolviendo (2.23)-(2.24) con x_i, y_i y w_i reemplazados por $\widetilde{x}_i$, $\widetilde{y}_i$ y $\widetilde{w}_i$).

La trasformación inversa que reemplaza w_i, y_i, x_i por $\breve{w}_i = w_i^2$, $\breve{y}_i = y_i/w_i$, $\breve{x}_i = x_i/w_i$ en (2.20)-(2.21) da (2.23)-(2.24). En otras palabras, cualquier algoritmo que resuelve (2.20)-(2.21) se puede usar para resolver (2.23)-(2.24) y viceversa. Notemos también que el problema tipo Huber es un caso particular del de Schweppe y Mallows ($w_i = 1$ para todo i). Por esta razón, sólo se describen los estimadores tipo Mallows. Las subrutinas de ROBETH proporcionan códigos diferentes para los tres casos.

Como se vió en la Observación 2 en 2.2, los estimadores de mínimos cuadrados y el mínimo de los valores absolutos de los residuales se pueden considerar casos especiales de (2.17) y también de (2.20). Por razones computacionales, ROBETH trata los problemas de mínimos cuadrados, del mínimo de los valores absolutos de los residuales y del tipo Mallows separadamente.

2.4 Descripción de las subrutinas en RGMAIN

Estimadores robustos de los coeficientes y del parámetro de escala

RIMTRF	Triangulación superior de la matriz de diseño y cálculo de su seudo-rango.
RIMTRD	Doble precisión de RIMTRF.
RICLLS	Solución del problema de mínimos cuadrados.
RILARS	Solución del mínimo del valor absoluto de los residuales.
RYNALG	Algoritmo de Newton con pasos adaptativos para M-estimadores.
RYHALG	H-algoritmo para M-estimadores.
RYWALG	W-algoritmo para M-estimadores.
RYSALG	S-algoritmo para M-estimadores.
RYSIGM	Algoritmo iterativo para el cálculo del M-estimador del parámetro de escala cuando los residuales estan dados.

Constantes

RIBETH	Cálculo de β cuando $\psi_d^2/2$ y ψ_d es la función de Huber.
RIBETU	Cálculo de β cuando χ es una función externa dada.
RIBETO	Cálculo de la constante β_0.

2.5 Ejemplos con S-PLUS y subrutinas del RGMAIN

2.5.1 Subrutina RIMTRF

```
# Triangulación superior (QR-descomposición) de la matriz de diseño
  # y determinación del seudo-rango. (ver Lawson and Hanson, pp. 11-12 [15])
  # Nombre del archivo en Microsoft Word: T- RIMTRF.
```

```
# Ejemplo: Datos del Draper and Smith, p. 361 [4].
library ("robeth",T)
dfvals( )
# Ingresar la cantidad n de filas de la matriz de diseño X.
n <- 21
# Ingresar la cantidad np de columnas de la matriz de diseño.
np <- 4
# Ingresar los datos de las variables X1, X2, X3.
X1 <- c(80,80,75,62,62,62,62,62,58,58,58,58,58,58,50,50,50,50,50,56,70)
X2 <- c(27,27,25,24,22,23,24,24,23,18,18,17,18,19,18,18,19,19,20,20,20)
X3 <- c(89,88,90,87,87,87,93,93,87,80,89,88,82,93,89,86,72,79,80,82,91)
X0 <- c(rep(1,n))
x <- matrix(c(X1,X2,X3,X0),ncol=np)
# El valor del seudorango=k:
# El vector sf es de dim=np; Si k<np, los elementos sf(1),...,sf(k) muestran los
# primeros k elementos de la diagonal de la matriz X una vez aplicada Q y P.
# El vector sg es de dim=np; Si k<np, los elementos sg(1),...,sg(k) muestran los
# escalares para la transformación V.
# El vector sh es de dim=np; los elementos sh(1),...,sh(np) muestran los
# escalares para la transformación Q
# El vector ip es de dim=np; los elementos ip(1),...,ip(np) muestran
# los índices p1,..., pp que definen las permutaciones de la matriz P.
z <- rimtrf(x)
k <- z$k; sf <- z$sf; sg <- z$sg; ip <- z$ip; sh<- z$sh
# Los valores de z, matriz transformada son:
z
rm(n,np,z,x,sg,ip,k,sf,sh,X1,X2,X3,X0)
```

Salida en S-PLUS

```
> # Los valores de z, matriz transformada son:
$x:
```

```
 [,1] [,2] [,3] [,4]
 [1,] -396.1364 -277.6518860 -96.8277740 -4.574182510
 [2,] 88.0000 -35.6991959 -9.4844227 0.028864484
 [3,] 90.0000 8.6502647 8.9122305 0.041048307
 [4,] 87.0000 -2.1380773 2.3837638 0.272643328
 [5,] 87.0000 -2.1380773 0.3837639 -0.004134518
 [6,] 87.0000 -2.1380773 1.3837639 -0.011505757
 [7,] 93.0000 -6.5613928 2.0727587 -0.089003481
 [8,] 93.0000 -6.5613928 2.0727587 -0.089003481
 [9,] 87.0000 -6.1380773 2.4874194 -0.022787482
[10,] 80.0000 -0.9775424 -2.1497412 0.095882945
[11,] 89.0000 -7.6125159 -2.6162488 -0.009306783
[12,] 88.0000 -6.8752966 -3.5644147 0.009752203
[13,] 82.0000 -2.4519808 -2.2534094 0.072507448
[14,] 93.0000 -10.5613928 -1.8235862 -0.063429005
[15,] 89.0000 -15.6125154 -0.4089382 -0.031870231
[16,] 86.0000 -13.4008579 -0.2534355 0.003193010
[17,] 72.0000 -3.0797882 1.4722435 0.159450233
[18,] 79.0000 -8.2403231 1.1094040 0.077636003
[19,] 80.0000 -8.9775419 2.0575697 0.058577020
[20,] 82.0000 -4.4519811 0.2984182 0.052124105
[21,] 91.0000 2.9130456 -4.0308838 -0.013579580
$k:
[1] 4
$sf:
[1] 0 0 0 0
$sg:
[1] 0 0 0 0
$sh:
[1] 485.1363525 50.8238983 -9.2708492 -0.2915203
$ip:
```

[1] 3 3 3 4

2.5.2 Subrutina RICLLS

Solución del problema de mínimos cuadrados.

Nombre del archivo en Microsoft Word: T-RICLLS.

Ejemplo 1: *Regresión lineal simple*, Rousseeuw and Leroy, p, 22 [27].

library ("robeth",T)

Ingresar cantidad de observaciones y cantidad de parámetros a estimar.

n <- 20; np <- 2

Ingresar los datos de la variable X0, en un vector de long n.

x0 <- c(rep(1,n))

Ingresar los datos de la variable predictora, en un vector de long n.

predictora <- c(123,109,62,104,57,37,44,100,16,28,138,105,159,75,88,164,169,167,149,167)

Ingresar los datos de la Y, en un vector de long n.

y <- c(76,70,55,71,55,48,50,66,41,43,82,68,88,58,64,88,89,88,84,88)

xt <-matrix (c(x0,predictora),ncol=np)

s <- riclls(xt,y,n,np,ix=1,iy=1)

El vector de parámetros estimados es:

theta0 <- s$theta[1:np]

theta0

El valor de sigma según 5., Marazzi and Randriamiharisoa p. 67 [18] es:

sigma0 <- s$sigma; sigma0

El valor de los residuales:

rs10 <- s$rs1, rs10

rm(n,np,x0,predictora,y,xt,theta0,sigma0,rs10)

Salida en S-PLUS

> # El vector de parámetros estimados es:

> theta0

[1] 35.4582672 0.3216082

> # El valor de sigma según 5., Marazzi and Randriamiharisoa p. 67 [18] es:

```
> sigma0
[1] 1.229979
> # El valor de los residuales:
> rs10
 [1] 0.9839131 -0.5135617 -0.3979822 2.0944724 1.2100589 0.6422224 0.3909649
 [8] -1.6190945 0.3959951 -1.4633036 2.1597927 -1.2271357 1.4060191 -1.5788891
[15] 0.2402040 -0.2020220 -0.8100631 -1.1668466 0.6221023 -1.1668466
> rm(n,np,x0,predictora,y,xt,theta0,sigma0,rs10)
```

Ejemplo 2: *Regresión lineal múltiple*, Dobson, p. 69 [3].

```
library ("robeth",T)
# Ingresar cantidad de observaciones.
n <- 20
# Ingresar los datos de la variable X0, en un vector de long n.
x0 <- c(rep(1,n))
# Ingresar los datos de las variables x1, x2, x3 en vectores de long n.
x1 <- c(33,47,49,35,46,52,62,23,32,42,31,61,63,40,50,64,56,61,48,28)
x2 <- c(100,92,135,144,140,101,95,101,98,105,108,85,130,127,109,107,117,100,118,102)
x3 <- c(14,15,18,12,15,15,14,17,15,14,17,19,19,20,15,16,18,13,18,14)
# Ingresar los datos de la Y(predictora), en un vector de long n.
y <- c(33,40,37,27,30,43,34,48,30,38,50,51,30,36,41,42,46,24,35,37)
# Ingresar cantidad de parámetros a estimar.
np <- 4
xt <-matrix (c(x0,x1,x2,x3),ncol=np)
s <- riclls(xt,y,ix=1,iy=1)
sigma0 <- s$sigma; theta0 <- s$theta[1:np];  rs10 <- s$rs1; rs20 <- s$rs2
# Los valores de sigma0 y teta0 son:
sigma0;  theta0
# Los residuales son:
rs10
rm(n,np,x0,x1,x2,x3,y,xt,theta0,sigma0,rs10)
```

Salida en S-PLUS

```
> # Los valores de sigma0 y teta0 son:
> sigma0
[1] 5.956419
> theta0
[1] 36.9600487 -0.1136763 -0.2280174 1.9577127
```

2.5.3 Subrutina RILARS

```
# Solución del problema del mínimo de los valores absolutos de los residuales.
    # Nombre del archivo en Microsoft Word: T-RILARS.
    # Ejemplo: Draper and Smith, p. 361 [4].
    library ("robeth",T)
    # Ingresar cantidad de observaciones y cantidad de parámetros a estimar.
    n <- 21;   np <- 4
    # Ingresar los datos de la variable x0, en un vector de long n.
    x0 <- c(rep(1,n))
    # Ingresar los datos de las variables x1,x2,x3 y de la variable y.
    x1 <- c(80,80,75,62,62,62,62,62,58,58,58,58,58,58,50,50,50,50,50,56,70)
    x2 <- c(27,27,25,24,22,23,24,24,23,18,18,17,18,19,18,18,19,19,20,20,20)
    x3 <- c(89,88,90,87,87,87,93,93,87,80,89,88,82,93,89,86,72,79,80,82,91)
    y <- c(42,37,37,28,18,18,19,20,15,14,14,13,11,12,8,7,8,8,9,15,15)
    z <- matrix(c(x0,x1,x2,x3),byrow=F,ncol=4)
    # La matriz con los datos es:
    z;  s <- rilars(z,y)
    # El rango de la matriz de diseño es:
    k <- s$k;  k
    # Cantidad de iteraciones para el algoritmo:
    nit <- s$nit;  nit
    theta1 <- s$theta
    estsigma <- s$sigma
```

```
# teha1(1), ..., teha1(np) dan los estimadores de los valores absolutos
# de los residuos de Theta, definidos en (2.6)-(2.7).
theta1
# El valor de kode debe ser 0, 1 o 2.
kode <- s$kode;  kode
# La estimación de sigma de acuerdo con (2.8) es:
estsigma;  rs <- s$rs
# El vector con los residuales es:
rs
rm(n,np,x0,x1,x2,x3,y,k,z,nit,theta1,estsigma,rs,kode)
```

Salida en S-PLUS

```
> # El rango de la matriz de diseño es:
> k
[1] 4
> # cantidad e iteraciones para el algoritmo:
> nit
[1] 7
> # teha1(1), ..., teha1(np) dan los estimadores de los valores absolutos
> # de los residuos de Theta, definidos en (2.6)-(2.7).
> theta1
 [1] -39.68985748 0.83188385 0.57391322 -0.06086949 1.21739066 1.79130375
 [7] 1.00000000 1.46376956 5.06087255 0.02028821 0.52753741 5.42898655
[13] 2.89854980 1.80289757 1.18260884 0.42608649 7.63478327 0.04058089
[19] 0.48695672 1.61739194 9.48115635
> # El valor de kode debe ser 0,1 o 2
> kode
[1] 1
> # La estimación de sigma de acuerdo con (2.8) es:
> sigma1
[1] Inf
```

> # El vector con los residuales es:

> rs

[1] 5.06087255 0.00000000 5.42898655 7.63478327 -1.21739066 -1.79130375

[7] -1.00000000 0.00000000 -1.46376956 -0.02028821 0.52753741 0.04058089

[13] -2.89854980 -1.80289757 1.18260884 0.00000000 -0.42608649 0.00000000

[19] 0.48695672 1.61739194 -9.48115635

> rm(k,x0,x1,x2,x3,y,n,np,z,theta1,sigma1,rs,kode)

2.5.4 Subrutina RYHALG, RYWALG, RYSALG y RYNALG

Archivo en Microsoft Word: T-RYNALG-RYWALG-RYSALG-RYNALG

Ejemplo: Datos del Finney, p. 70 [5]. Estimación tipo Huber usando varios algoritmos.

Los datos relacionados con un ensayo de vitamina D_3 de aceite de hígado de bacalao para

medir su efecto antirraquítico en pollos. Se usaron 55 pollos, cada uno de ellos recibió

cierta dosis de una preparación estándar S o una preparación T. Una escala logarítmica

apropiada hizo que S tome los valores -2, 0 y 2; que T tome los valores -3, -1, 1 y 3. La

respuesta **y** considerada fue porcentaje de ceniza de hueso. Al ejemplo se agregó un

punto "artificial" (el pollo número 56) con una respuesta remota que aumentó la clásica

estimación del desvío, Marazzi and Randriamiharisoa, p. 101 [18] y

Marazzi et al. [19]. Las subrutinas RYHALG, RYWALG y RYSALG

aplican respectivamenste: algoritmo de Newton, H-algoritmo y W-algoritmo

para M-estimadores para resolver el problema de tipo Huber, Mallows o de Schweppe

con estimación simultánea del parámetro de escala. La subrutina RYSALG aplica el

S-algoritmo para M-estimadores para la función de Huber produciendo la solución exacta.

library ("robeth",T)

dfvals( )

Ingresar los datos y declarar el vector de pesos wgt por defecto.

z <- c(-1, -2, 0, 35, 1, 0,-3, 20, -1, -2, 0, 30, 1, 0, -3, 39, -1, -2, 0, 24, 1, 0, -3, 16,

-1, -2, 0, 37, 1, 0, -3, 27, -1, -2, 0, 28, 1, 0, -3, -12, -1, -2, 0, 73, 1, 0, -3, 2,

-1, -2, 0, 31, 1, 0, -3, 31, -1, -2, 0, 21, 1, 0, -1, 26, -1, -2, 0, -5, 1, 0, -1, 60,

-1, 0, 0, 62, 1, 0, -1, 48, -1, 0, 0, 67, 1, 0, -1, -8, -1, 0, 0, 95, 1, 0, -1, 46,

-1, 0, 0, 62, 1, 0, -1, 77, -1, 0, 0, 54, 1, 0, 1, 57, -1, 0, 0, 56, 1, 0, 1, 89,

```
-1, 0, 0, 48, 1, 0, 1, 103,    -1, 0, 0, 70, 1, 0, 1, 129,    -1, 0, 0, 94, 1, 0, 1, 139,
-1, 0, 0, 42, 1, 0, 1, 128,    -1, 2, 0, 116, 1, 0, 1, 89,    -1, 2, 0, 105, 1, 0, 1, 86,
-1, 2, 0, 91, 1, 0, 3, 140,    -1, 2, 0, 94, 1, 0, 3, 133,    -1, 2, 0, 130, 1, 0, 3, 142,
-1, 2, 0, 79, 1, 0, 3, 118,    -1, 2, 0, 120, 1, 0, 3, 137,    -1, 2, 0, 124, 1, 0, 3, 84,
-1, 2, 0, -8, 1, 0, 3, 101)
xx <- matrix(z,ncol=4, byrow=T)
dimnames(xx) <- list(NULL,c("z2","xS","xT","y"))
z2 <- xx[,"z2"]; xS <- xx[,"xS"]; xT <- xx[,"xT"]
x <- cbind(1, z2, xS+xT, xS-xT, xS^2+xT^2, xS^2-xT^2, xT^3)
y <- xx[,"y"]
wgt <- vector("numeric",length(y))
n <- 56; np <- 7
# Conjunto de parámetros para la estimación de Huber.
dfrpar(x, "huber")
# Cálculo de las constantes beta, bet0, epsi2 y epsip.
ribeth(wgt)
ribet0(wgt)
s <- liepsh();  epsip <- s$epsip; epsi2 <- s$epsi2
epsip;  epsi2
#-------------------------------------------------------------------------
# Solución (theta0, sigma0) por medio de mínimos cuadrados.
z <- riclls(x, y)
theta0<- z$theta; sigma0 <- z$sigma
# Estimación de la matriz de covarianza de los coeficientes.
cv <- kiascv(z$xt, fu=epsi2/epsip^2, fb=0.);  cov <- cv$cov
#-------------------------------------------------------------------------
# Solución (theta1, sigma1) por medio de RYHALG.
zr <- ryhalg(x,y,theta0,wgt,cov,sigmai=sigma0,ic=0)
theta1<- zr$theta[1:np]; sigma1 <- zr$sigmaf; nit1 <- zr$nit
# -------------------------------------------------------------------------
# Solución (theta2, sigma2) por medio de RYWALG y cálculo de covarinza.
```

```
cv <- ktaskv(x, f=epsi2/epsip^2)
zr <- rywalg(x, y, theta0, wgt, cv$cov, sigmai=sigma0)
theta2 <- zr$theta[1:np]; sigma2 <- zr$sigmaf; nit2 <- zr$nit
#-------------------------------------------------------------------------
# Solución (theta3, sigma3) por medio de RYSALG con ISIGMA=2.
zr <- rysalg(x,y, theta0, wgt, cv$cov, sigma0, isigma=2)
theta3 <- zr$theta[1:np]; sigma3 <- zr$sigmaf; nit3 <- zr$nit
# -------------------------------------------------------------------------
# Solución (theta4, sigma4) por medio de RYNALG con ICNV=2 y ISIGMA=0.
# Inversa de la cov
covm1 <- cv$cov;  zc <- mchl(covm1,np)
zc <- minv(zc$a, np);  zc <- mtt1(zc$r,np);  covm1 <- zc$b
zr <- rynalg(x,y, theta0, wgt, covm1, sigmai=sigma3,
iopt=1, isigma=0, icnv=2)
theta4 <- zr$theta[1:np]; sigma4 <- zr$sigmaf; nit4 <- zr$nit
# -------------------------------------------------------------------------
# Solución (theta0, sigma0) por medio de mínimos cuadrados.
theta0[1:np]
sigma0
# Solución (theta1, sigma1) por medio de RYHALG.
theta1[1:np]
sigma1
nit1
# Solución (theta2, sigma2) por medio de RYWALG
theta2[1:np]
sigma2
nit2
# Solución (theta3, sigma3) por medio de RYSALG con ISIGMA=2.
theta3[1:np]
sigma3
nit3
```

```
# Solución (theta4, sigma4) por medio de RYNALG
theta4[1:np]
sigma4
nit4
```

Salida en S-PLUS

```
> # Solución (theta0, sigma0) por medio de mínimos cuadrados.
> theta0[1:np]
[1] 68.6339188 3.6339283 24.0808525 -8.0530758 -0.4464282 -0.1785715
[7] -1.6339287
> sigma0
[1] 26.63453
> # Solución (theta1, sigma1) por medio de RYHALG.
> theta1[1:np]
[1] 70.00614166 5.00614071 24.74172401 -6.24567747 -0.07897823 0.43415147
[7] -1.48664904
> sigma1
[1] 23.56445
> nit1
[1] 7
> # Solución (theta2, sigma2) por medio de RYWALG
> theta2[1:np]
[1] 70.00643921 5.00643778 24.74151230 -6.24533176 -0.07896668 0.43420276
[7] -1.48657823
> sigma2
[1] 23.56336
> nit2
[1] 7
> # Solución (theta3, sigma3) por medio de RYSALG con ISIGMA=2.
> theta3[1:np]
[1] 69.9932022 5.0015516 24.7659245 -6.2144694 -0.0548972 0.4398556
```

[7] -1.4803939

> sigma3

[1] 22.24895

> nit3

[1] 3

> # Solución (theta4, sigma4) por medio de RYNALG

> theta4[1:np]

[1] 69.99320221 5.00155163 24.76592445 -6.21446896 -0.05489732 0.43985555

[7] -1.48039389

> sigma4

[1] 22.24895

> nit4

[1] 3

Capítulo 3

Matriz de Covarianza de los Coeficientes Estimados

En este capítulo se presentan procedimientos computacionales incluidos en el programa ROBETH para estimar la matriz de covarianza de los coeficientes estimados según el capítulo 2 del Marazzi and Randriamiharisoa [18]. El conjunto de subrutinas a usar estan agrupadas con el nombre de KVMAIN.

3.1 Introducción

La matriz de covarianza asintótica de un M-estimador general de regresión está definida, según Huber, Cap. 7 [10] y Hampel et al., Cap. 6 [8] por:

$$\sum_{i=1}^{n} \eta(x_i, (y_i - x_i^T \widehat{\theta})/\sigma) x_i = 0 \tag{3.1}$$

es

$$K_v(\widehat{\theta}) = S_1^{-1} S_2 S_1^{-1}, \tag{3.2}$$

donde

$$S_1 = \int \eta'(x,s) x x^T \cdot dM(x, s/\sigma) \quad \text{y} \quad S_2 = \int \eta^2(x,s) x x^T \cdot dM(x, s/\sigma). \tag{3.3}$$

El programa de computación ROBETH proporciona la estimación de $K_v(\widehat{\theta})$ para :

- $\widehat{\theta} = \widehat{\theta}_{LS}$ estimador de mínimos cuadrados
- $\widehat{\theta} = \widehat{\theta}_{WLS}$ estimador de mínimos cuadrados pesado
- $\widehat{\theta} = \widehat{\theta}_{LAR}$ estimador del mínimo del valor absoluto de los residuales

$\widehat{\theta} = \widehat{\theta}_H$ estimador del tipo Huber

$\widehat{\theta} = \widehat{\theta}_M$ estimador del tipo Mallows

$\widehat{\theta} = \widehat{\theta}_S$ estimador del tipo Schweppe

Las estimaciones $\widehat{K}_u$ o K_u se muestran en la Tabla 3.1 donde $\widehat{\sigma}$ denota un estimador robusto de σ. Para el caso de Huber, Mallows, y Schweppe se dan tres opciones; que llamaremos opción *a esperada*, opción *b promedio* y opción *c empírica*.

Las subrutinas estan divididas en tres gurpos:

1. subrutinas para el cálculo de estimadores de la forma: $f\cdot\ (X^TX)^{-1}$, donde f es un factor escalar dado;

2. subrutinas para el cálculo de estimadores de la forma $f \cdot S_1^{-1}S_2S_1^{-1}$, donde $S_1 = (1/n)X^TDX$, $S_2 = (1/n)X^TEX$, y f es un factor escalar, D y E son matrices diagonales dadas;

3. subrutinas para el cálculo del factor f y las matrices diagonales D y E en casos particulares.

Las constantes $\int \psi^2(s) \cdot d\Phi(s)$ y $\int \psi'(s) \cdot d\Phi(s)$ usadas en ciertas expresiones de la Tabla 3.1 se pueden calcular mediante las subrutinas LIEPSH y PIEPSU dadas en el capítulo 2.

Observación 1: La matriz de covarianza $C(\widehat{\theta})$ de los coeficientes estimados se puede estimar multiplicando $\widehat{K}_u$ por $\widehat{\sigma^2}/n$.

Observación 2: El cálculo de la matriz de covarianza para el caso de Mallows se puede reducir al caso de Schweeppe usando la trasformación (2.29) del capítulo 2.

Observación 3: En algunos casos especiales, S_1 es de la forma $S_1 = f_1\cdot\ A^{-1}(A^{-1})^T$, con A una matriz triangular superior o de la forma $S_1 = f_1 \cdot (A^{-1})$, donde A es simétrica.

Ejemplos son (ver Cap. 3 de Marazzi and Randriamiharisoa [18])

Estimador de Hampel-Krasker:	$S_1 = A^{-1}$
Estimador óptimo no estandarizado de tipo Mallows:	$S_1 = \int \psi_c'(s) \cdot d\Phi(s) \cdot A^{-1}$
Estimador de tipo Mallows para el τ-test:	$S_1 = \int \psi_c'(s) \cdot d\Phi(s) \cdot A^{-1} \cdot (A^{-1})^T$
Estimador de tipo Schweppe para el τ-test.	$S_1 = A^{-1} \cdot (A^{-1})^T$

Tabla 3.1 Estimadores de la matriz de covarianza asintótica de los coeficientes estimados

$\widehat{\theta}$	Opción	Estimación de $\widehat{K}$ o de $K_\upsilon(\widehat{\theta})$
$\widehat{\theta}_{LS}$		$n(X^TX)^{-1}$
$\widehat{\theta}_{WLS}$		$n(X^TWX)^{-1}X^TWWX(X^TWX)^{-1}$
$\widehat{\theta}_{LAR}$		$nf_{LAR}\cdot\ (X^TX)^{-1}$, donde f_{LAR}=$0.25/l(0)^2$, l=densidad de e_i/σ
$\widehat{\theta}_H$	a	$nf_H\cdot\ (X^TX)^{-1}$, donde
		$f_H=[\int\psi^2(s)d\Phi(s)]\diagup(\int\psi'(s)d\Phi(s))^2$
	b	$n\widehat{f}_H\cdot\ (X^TX)^{-1}$, donde
		$\widehat{f}_H=n\widehat{f}_0^2\ ave\ \psi^2/[(n-p)(ave\ \psi')^2]$
		$\widehat{f}_0=1+(p/n)\ var\ \psi'/(ave\ \psi')^2$
		$ave\ \psi^2=(\frac{1}{n}\sum\psi^2(r_i/\widehat{\sigma})$
		$ave\ \psi'=(\frac{1}{n}\sum\psi'(r_i/\widehat{\sigma})$
		$var\ \psi'=(\frac{1}{n}\sum(\psi'(r_i/\widehat{\sigma})-ave\ \psi')^2$
$\widehat{\theta}_M$	a	$S_1^{-1}S_2S_1^{-1}$, donde
		$S_1=n^{-1}X^TD_MX$ y $D_M=\int\psi^{'}(s)\cdot d\Phi(s)\cdot diag(w_i)$
		$S_2=n^{-1}X^TE_MX$ y $E_M=\int\psi^2(s)\cdot d\Phi(s)\cdot diag(w_i^2)$
	b	$\widehat{S}_1^{-1}\widehat{S}_2\widehat{S}_1^{-1}$, donde
		$\widehat{S}_1=n^{-1}X^T\widehat{D}_MX$ y $\widehat{D}_M=\left[\sum_j\psi^{'}(r_j/\widehat{\sigma})/n\right]\cdot diag(w_i)$
		$\widehat{S}_2=n^{-1}X^T\widehat{E}_MX$ y $\widehat{E}_M=\left[\sum_j\psi^2(r_j/\widehat{\sigma})/n\right]\cdot diag(w_i^2)$
	c	$(\breve{S_1})^{-1}(\breve{S_2})(\breve{S_1})^{-1}$ donde
		$(\breve{S_1})=n^{-1}X^T\ \breve{D_M}\ X$ y $\breve{D_M}=diag(\psi^{'}(r_i/\widehat{\sigma})\cdot w_i)$
		$(\breve{S_2})=n^{-1}X^T\ \breve{E_M}\ X$ y $\breve{E_M}=diag(\psi^2(r_i/\widehat{\sigma})\cdot w_i^2)$

Tabla 3.1 (continuación)

$\widehat{\theta}_S$	a	$S_1^{-1}S_2S_1^{-1}$, donde
		$S_1 = n^{-1}X^T D_S X$ y $D_S = diag(\int \psi^{'}(s/w_i)\cdot d\Phi(s))$
		$S_2 = n^{-1}X^T E_S X$ y $E_S = diag(\int \psi^2(s/w_i))w_i^2 d\Phi(s))$
	b	$\widehat{S}_1^{-1}\widehat{S}_2\widehat{S}_1^{-1}$, donde
		$\widehat{S}_1 = n^{-1}X^T \widehat{D}_S X$ y $\widehat{D}_S = diag[\sum_j \psi^{'}(r_j/(\widehat{\sigma}w_i))/n]$
		$\widehat{S}_2 = n^{-1}X^T \widehat{E}_S X$ y $\widehat{E}_S = diag[w_i^2 \cdot \sum_j \psi^2(r_j/(\widehat{\sigma}w_i))/n]$
	c	$(\breve{S_1})^{-1}(\breve{S_2})(\breve{S_1})^{-1}$, donde
		$(\breve{S_1})= n^{-1}X^T \breve{D_S} X$ y $\breve{D_S}= diag(\psi^{'}(r_i/\widehat{\sigma})\cdot w_i)$
		$(\breve{S_2})= n^{-1}X^T \breve{E_S} X$ y $\breve{E_S}= diag(\psi^2(r_i/\widehat{\sigma})\cdot w_i^2)$

3.2 Descripción de las subrutinas en KVMAIN

Matriz de covarianza de los coeficientes estimados

KTASKV	Matriz de covarianza de los coeficientes estimados de la forma $f\cdot(X^TX)^{-1}$.
KTASKW	Matriz de covarianza de los coeficientes estimados de la forma . $f\cdot S_1^{-1}S_2S_1^{-1}$.
KIASCV	Matriz de covarianza de los coeficientes estimados de la forma $f\cdot(X^TX)^{-1}$.
	en el sistema de coordenadas transformado.
KFASCV	"Backtransformation" de la matriz de covarianza de los coeficientes estimados.
KFFACV	Factor de corrección $\widehat{f}_H$ para la matriz de covarianza de los
	estimadores tipo Huber.
KIEDCH	Matrices diagonales D_M, E_M, D_S y E_S, donde ψ es la función de Huber.
KIEDCU	Matrices diagonales D_M, E_M, D_S y E_S, donde es una función dada.
KFEDCB	Matrices diagonales Matrices diagonales $\widehat{D}_M$, $\widehat{E}_M$, $\widehat{D}_S$ y $\widehat{E}_S$.
KFEDCC	Matrices diagonales $\breve{D_M}$, $\breve{E_M}$, $\breve{D_S}$y $\breve{E_S}$.

3.3 Ejemplos con S-PLUS y subrutinas del KVMAIN

3.3.1 Subrutina KTASKV

El nombre del archivo en Microsoft Word: Sub T-KTASKV

KTASKV, matriz de covarianza de los coeficientes estimados por f.$(X^TX)^{-1}$

La subrutina KTASKV realiza los siguientes pasos:

1. Cálculo de X^TX.

2. Descomposición de X^TX por el método Cholesky, esto es, encontrar una matriz

triangular superior A^{-1} tal $X^TX=A^{-1}(A^{-1})^T$.

3. Cálculo de la inversa de A^{-1} , llamaremos A a la inversa de A^{-1} .

4. Cálculo de K=f .A^TA, donde f es un factor dado.

La subrutina KTASKV da las matrices A y K.

Ejemplo: datos del Dobson, p. 75 [3], con f=0.0, valor tomado por defecto).

```
library ("robeth",T)
dfvals()
# Ingresar n, cantidad de filas de la matriz X y p, cantidad de columnas.
n <- 20; np <- 4
# Ingresar los valores de X, por filas.
z <- c(1,33,100,14,1,47,92,15,1,49,135,18,1,35,144,12,1,46,140,15,1,52,101,15,
1,62,95,14,1,23,101,17,1,32,98,15,1,42,105,14,1,31,108,17,1,61,85,19,1,63,130,19,
1,40,127,20,1,50,109,15,1,64,107,16,1,56,117,18,1,61,100,13,1,48,118,18,1,28,102,14)
x <- matrix(z,ncol=np, byrow=TRUE)
s <- ktaskv(x,n,np,f);  matrizA <- s$a;   cov <- s$cov;   matrizA;  cov
# rm(n,np,z,x,matrizA,cov,f)
```

Salida en S-PLUS

```
> matrizA
[1] 0.2236067951 -0.8288046718 0.0179589316 -1.5638757944 0.0007707351
[6] 0.0138058392 -1.2779455185 -0.0037094671 -0.0022189845 0.1065898761
> cov
[1] 4.8157691956 -0.0113492832 0.0003368774 -0.0187548772 0.0000188719
```

[6] 0.0001955251 -0.1362160593 -0.0003953916 -0.0002365213 0.0113614015

3.3.2 Subrutina KIASCV y KFASCV

```
# El nombre del archivo en Microsoft Word: T-KIASCV KFASCV.
  # KIASCV, matriz de covarianza de los coeficientes estimados por f.(X^T X)^-1
  # en el sistema de coordenadas transformadas.
  # KFASCV, "backtransformation" de la matriz de covarianza
  # de los coeficientes estimados. Ejemplo: datos del Draper and Smith, p. 361 [4].
  rm(z,x,n,np,epsi2,epsip,z,xt,sg,ip,zc,covi,covf)
  library ("robeth",T)
  dfvals()
  # Ingrasar los datos:
  z <- c(80, 27, 89, 1, 80, 27, 88, 1, 75, 25, 90, 1,
  62, 24, 87, 1, 62, 22, 87, 1, 62, 23, 87, 1,    62, 24, 93, 1, 62, 24, 93, 1, 58, 23, 87, 1,
  58, 18, 80, 1, 58, 18, 89, 1, 58, 17, 88, 1,    58, 18, 82, 1, 58, 19, 93, 1, 50, 18, 89, 1,
  50, 18, 86, 1, 50, 19, 72, 1, 50, 19, 79, 1,    50, 20, 80, 1, 56, 20, 82, 1, 70, 20, 91, 1)
  # Ingresar n, cantidad de filas de la matriz X y p, cantidad de columnas.
  n <- 21;   np <- 4;   x <- matrix(z, ncol=4, byrow=TRUE)
  # Matriz de covarianza de los coeficientes estimados tipo Huber.
  dfrpar(x, "huber"); s <- liepsh(); epsi2 <- s$epsi2; epsip <- s$epsip
  z <- rimtrf(x); xt <- z$x; sg <- z$sg; ip <- z$ip;   zc <- kiascv(xt, fu=epsi2/epsip^2, fb=0.)
  covi <- zc$cov;   covi;   zc <- kfascv(xt, covi, f=1, sg=sg, ip=ip);   covf <- zc$cov;   covf
  rm(z,x,n,np,epsi2,epsip,xt,sg,ip,zc,covi,covf)
```

Salida en S-PLUS

```
> covi
[1] 2.444386e-003 -7.148215e-004 1.819867e-003 1.047777e-006 -3.653463e-003
[6] 1.355307e-002 -1.677421e-001 2.877762e-002 -6.522219e-002 1.416076e+001
> covf
[1] 1.819867e-003 -3.653463e-003 1.355307e-002 -7.148215e-004 1.047777e-006
[6] 2.444386e-003 2.877762e-002 -6.522219e-002 -1.677421e-001 1.416076e+001
```

Capítulo 4

Alto Punto de Ruptura en Regresión

En este capítulo se presentan procedimientos computacionales incluidos en el programa ROBETH para el cálculo de estimadores con alto punto de ruptura en regresión lineal. El conjunto de subrutinas a usar se denomina HBMAIN.

4.1 Introducción

Siguiendo con la notación de Rousseeuw and Leroy [27] y de García Peña [22], se considera el modelo lineal

$y_i = x_{i1}\theta_1 + \cdots + x_{ip}\theta_p + e_i$ para $i = 1, \cdots, n$, donde n es el tamaño de la muestra,

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_p \end{bmatrix} \quad \text{y} \quad \text{casos} \begin{bmatrix} & & \text{variables} & \\ x_{11} & \cdots & x_{1p} & y_1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{i1} & & x_{ip} & y_i \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{np} & y_n \end{bmatrix}$$

Es decir, $\Omega : y_i = x_i^T\theta + e_i$, $i = 1, \cdots, n$.

4.1.1 Estimadores con alto punto de ruptura

En la introducción de esta tesis se vió que, en análisis de regresión lineal simple un simple "outlier" afecta al estimador de mínimos cuadrados (LS) de manera tal que éste

no ajusta bien al conjunto de datos. Pero existen otros estimadores que dan buen ajuste a conjuntos de datos que incluyen no sólo uno, sino gran cantidad de puntos "outliers".

Una medida de la robustez global de un estimador está dada por su *punto de ruptura* "*breakdown point*". La definición más antigua, dada por Hodges [9] se restringe a los estimadores de locación unidimensionales mientras que Hampel [7] da una formulación más general que es asintótica en su naturaleza. En palabras, según Maronna y Yohai [20] [38], punto de ruptura de un estimador representa la menor proporción de "outliers" que provocan que el estimador deje de dar información sobre el parámetro que se está estimando, en este caso, sobre los coeficientes de regresión.

Además de la versión asintótica de punto de ruptura, existe otra para muestras finitas, y es la que se considera en este capítulo. Pero para definir el punto de ruptura de un estimador $\widehat{\theta}_n$ para una muestra finita $Z = (z_1, ..., z_n)$, es necesario previamente definir el máximo sesgo producido por m "outliers":

Sea una muestra cualquiera de n datos,

Z=$\{(x_{11}, ..., x_{1p}, y_1), ..., (x_{n1}, ..., x_{np}, y_n)\}$, y sea T un estimador de regresión, esto significa que aplicando T a tal muestra Z se obtiene un vector de coeficientes de regresión de la forma $T(Z) = \begin{bmatrix} \widehat{\theta}_1 \\ \vdots \\ \widehat{\theta}_p \end{bmatrix} = \widehat{\theta}.$

Consideremos ahora cualquier muestra Z' que se obtiene de la original reemplazando m de los datos originales por valores arbitrarios (esto permite una cantidad considerable de malos "outliers").

Notemos con sesgo(m;T,Z) el mayor sesgo que se puede lograr con tal contaminación:

$$\text{sesgo(m;T,Z)}=\sup_{Z'} \| T(Z') - T(Z) \|,$$

donde el supremo es sobre todas las muestras Z' posibles. Es decir sesgo(m;T,Z) es la máxima variación que puede experimentar el estimador $\widehat{\theta}_n$ cuando la muestra Z se contamina con m "outliers". Si sesgo(m;T,Z) es infinito, significa que m "outliers" pueden tener un efecto arbitrariamente grande sobre T, lo que se expresa diciendo que el estimador "*breaks down*". El punto de ruptura para una muestra finita Z se define por

$$\varepsilon_n^*(T,Z) = mínimo\left\{\frac{m}{n}; sesgo(m;T.Z) \text{ es infinito}\right\}.$$

Es decir, ε_n^* es la mínima proporción de "outliers" que puede hacer que el sesgo del estimador supere cualquier límite. Si el $\underset{n\to\infty}{lím}\ \varepsilon_n^*(T,Z) = \varepsilon^0$ el valor ε^0 se denomina punto de ruptura asintótico. Típicamente ε_n^* está comprendido entre 1/n y $[n/2]+1$ y ε^0 entre 0 y 0.5. Para el estimador LS, sólo un "outlier" es suficiente para ubicar a T fuera de todos los límites, es decir su punto de ruptura es $\varepsilon_n^*(T,Z) = \frac{1}{n}$ y cuando el tamaño n de la muestra tiende a infinito, podemos decir que LS tiene un punto de ruptura del 0%, reflejandose así la extrema sensibilidad de LS, ante la presencie de "outliers". Se puede ver también, por ejemplo, que el punto de ruptura de la media aritmética es del 0%, mientras que el de la mediana es del 50%. El punto de ruptura de los estimadores de tipo Huber y estimadores de los valores absolutos de los residuos es del 0% (no dando solución a los puntos palancas) y no pude ser mas grande que $1/p$ para estimadores del tipo Mallows y del tipo Schweppe. El punto de ruptura igual a 1/2 es el más alto que se puede alcanzar, ya que si la contaminación fuera superior, no es posible decidir cuál parte de los datos es correcta, según Rousseeuw and Leroy [27] para el caso de estimadores de regresión equivariantes vale el siguiente teorema:

Teorema: *Para cualquier estimador T de regresión equivariante se cumple*

$\varepsilon_n^*(T,Z) <= ([(n-p)/2]+1)/n$ *para cualquier muestra Z, donde* $[x]$ *denota la parte entera de x.*

Entre los *estimadores de regresión con alto punto de ruptura* que trata el programa ROBETH se encuentran el *least median of squares* (LMS) (Souvaine and Steele, [30]), el *least trimmed squares* (LTS) y *S-estimadores*. Una combinación de las subrutinas dadas para los anteriores con las del Cap. 2, permite el cálculo del *MM-estimador* (Yohai [37]).

4.1.2 Estimador de mediana de cuadrados mínimos

El estimador de *mediana de cuadrados mínimos* (*LMS*) está definido como el p-vector

$$\widehat{\theta}_{LMS} = \underset{\theta}{mínimo}\, Q_{LMS}(\theta), \tag{4.1}$$

donde

$$Q_{LMS}(\theta) = \underset{i}{med}\ (y_i - x_i^T\theta)^2. \tag{4.2}$$

4.1.3 Estimador de mínimos cuadrados truncados

El estimador de *mínimos cuadrados truncados* (LTS) está definido como el p-vector

$$\widehat{\theta}_{LTS} = \underset{\theta}{mínimo}\, Q_{LTS}(\theta), \tag{4.3}$$

donde

$$Q_{LTS}(\theta) = \sum_{i=1}^{k} r^2_{[i]}, \tag{4.4}$$

donde $r^2_{[1]} <= r^2_{[2]} <= \cdots <= r^2_{[n]}$ son los cuadrados ordenados de los residuales $r_i^2 = (y_i - \sum_j x_{ij}\theta_j)^2$, donde $(i = 1, \cdots, n)$, y k es el mayor entero $<= n/2 + 1$.

4.1.4 S-estimadores

Sea χ una función tal que:

a_1: $\chi(0) = 0, \quad \chi(-s) = \chi(s)$;

a_2: χ es continua;

a_3: $0 <= s <= t \Longrightarrow \chi(s) <= \chi(t)$;

a_4: $0 < a = \sup \chi(s) < \infty$;

a_5: $\chi(s) < a$ y $0 <= s < t \Longrightarrow \chi(s) < \chi(t)$.

Entonces el *S-estimador* (ver Yohai and Zamar [39]) está definido por el p-vector

$$\theta_* = \underset{\theta}{mínimo}\, S(\theta), \tag{4.5}$$

donde S(θ) es la solución de

$$\frac{1}{n-p}\sum_{i=1}^{n} \chi(\frac{y_i - x_i^T\theta}{S}) = \beta \tag{4.6}$$

y β es una constante adecuada.

4.1.5 MM-estimadores

Sean χ y ρ dos funciones tales que:

a_1: $\chi(0) = 0, \quad \chi(-s) = \chi(s);$ b_1: $\rho(0) = 0, \quad \rho(-s) = \rho(s);$

a_2: χ es continua; b_2: ρ es continua;

a_3: $0 <= s <= t \Longrightarrow \chi(s) <= \chi(t);$ b_3: $0 <= s <= t \Longrightarrow \rho(s) <= \rho(t);$

a_4: $0 < a = \sup \chi(s) < \infty;$ b_4: $0 < b = \sup \rho(s) < \infty;$

a_5: $\chi(s) < a$ y $0 <= s < t \Longrightarrow \chi(s) < \chi(t).$ b_5: $\rho(s) < b$ y $0 <= s < t \Longrightarrow \rho(s) < \rho(t).$

c_1: $\chi(s) = \rho(s);$

c_2: $a = b.$

Entonces el *MM-estimador* $\widehat{\theta}_{MM}$ está definido en tres pasos como sigue:

1. Cálculo de un estimador(consistente) con alto punto de ruptura $\widehat{\theta}'$.

2. Cálculo $r'_i = (y_i - x_i^T \widehat{\theta}')$ y cálculo de $\widehat{\sigma}'$ tal que

$$\frac{1}{n-p} \sum_{i=1}^{n} \chi(\frac{r'_i}{\widehat{\sigma}'}) = \beta, \tag{4.7}$$

   donde $\beta = a/2$.

3. Cálculo de un mínimo local $\widehat{\theta}_{MM}$ de

$$Q_{MM}(\theta) = \sum_{i=1}^{n} \rho(\frac{y_i - x_i^T \theta}{\widehat{\sigma}'}) \tag{4.8}$$

   tal que

$$Q_{MM}(\widehat{\theta}_{MM}) <= Q_{MM}(\widehat{\theta}'). \tag{4.9}$$

   Observación 1: Todos los estimadores definidos en este capítulo son equivariantes afines (Rousseeuw and Leroy [27]).

Observación 2: A pesar que el punto de ruptura del estimador LMS es del 50%, es altamente ineficiente cuando todas las observaciones cumplen con el modelo de regresión gaussiano

$$y_i = x_i^T \theta + e_i, \tag{4.10}$$

donde (x_i, y_i) son variables aleatorias independientes e idénticamente distribuidas tal que las x_i siguen alguna distribución G y los e_i son independientes de los x_i distribuidos de

acuerdo con $\Phi(s/\sigma)$. El punto de ruptura del LTS es del 50% y tasa de convergencia es $n^{(-1/2)}$. Tiene la misma eficiencia asintótica (relativa a los estimadores de mínimos cuadrados) para el modelo gaussiano como para los estimadores de tipo Huber definidos por Marazzi and Randriamiharisoa [18].

$$\psi(s) = \begin{cases} s & \text{con} \mid s \mid <= \Phi^{-1}(3/4) \\ 0 & \text{en el resto.} \end{cases} \tag{4.11}$$

La principal desventaja del estimador LTS es que su cálculo requiere que los residuales se los ordene generando más operaciones para el cálculo de la mediana.

Observación 3: Para S-estimadores la elección usual de χ es

$$\chi_k(s) := \begin{cases} 3(s/k)^2 - 3(s/k)^4 + (s/k)^6 & si \mid s \mid <= k \\ 1 & si \mid s \mid > k. \end{cases} \tag{4.12}$$

Aquí, $a = \sup_s \chi_k(s) = \chi_k(k) = 1$. Usualmente β es igual a $\int \chi(s) \cdot d\Phi(s)$; es este caso $\widehat{\theta}_*$ y $S(\widehat{\theta}_*)$ son estimadores asintóticamente consistentes de θ y σ para el modelo de regresión gaussiano. Es posible elegir la constante $k = k_0$ tal que el punto de ruptura de $\widehat{\theta}_*$ [y $S(\widehat{\theta}_*)$] sea igual a un valor dado de α. La condición es tal que $\alpha \cdot \sup_s \chi_k(s) = \beta$; esto es,

$$\alpha = \int \chi_{k_0}(s) \cdot d\Phi(s). \tag{4.13}$$

Para $\alpha = 50\%$ se obtiene $k_0 = 1.548$. La correspondiente eficiencia asintótica (relativa) para el modelo gaussiano es del 28.7%.

Observación 4: Para MM-estimadores un camino para la elección de χ y ρ tal que cumplan con las condiciones es el siguiente: Sea χ^0 una función que cumple con a_1-a_5, y sea $0 < k_0 < k$. Por ejemplo, sea $\chi^0(s) = \chi_1(k)$, donde χ_1 es la función definida en la observación 3 con $k = 1$. Sea $\chi(s) = \chi^0(s/k_0)$ y $\rho(s) = \chi^0(s/k_1)$ y tomando $k_0 = 1.548$ y $\beta = \int \chi(s) \cdot d\Phi(s)$. En este caso, si $\widehat{\theta}'$ es un estimador asintóticamente consistente de θ, entonces $\widehat{\sigma}'$ es un estimador asintóticamente consistente de σ en el modelo gaussiano y tiene el mismo punto de ruptura que $\widehat{\theta}'$. El MM-estimador $\widehat{\theta}_{MM}$ tiene el mismo punto de ruptura que $\widehat{\theta}'$ y la misma eficiencia (determinada por k_1) que el estimador de Huber dado

en (4.8) y (4.9) con estimador de escala separado. El valor de k_1 que da una eficiencia relativa del 95% con respecto a $\widehat{\theta}_{LS}$ es 4.687.

4.2 Descripción de las subrutinas en HBMAIN

Estimadores con alto punto de ruptura en regresión

HYLMSE	Algoritmo para el cálculo del estimador LMS.
HYLTSE	Algoritmo para el cálculo del estimador LTS.
HYSEST	Algoritmo para el cálculo de S-estimadores.

4.3 Ejemplos con S-PLUS y subrutinas del HBMAIN

4.3.1 Subrutina HYLMSE

Algoritmo para calcular el estimador LMS.

Nombre del archivo en Microsoft Word: T-HYLMSE.

Ejemplo: Draper and Smith, p. 361 [4].

Se considera el modelo $y_i = x_{i1}\theta_1 + x_{i2}\theta_1 + x_{i3}\theta_p + \theta_4 + e_i$ donde $i = 1, \cdots 21$.

library ("robeth",T)

Ingresar cantidad de observaciones y cantidad de parámetros a estimar.

n <- 21; np <- 4; nq <- np+1

Ingresar los datos de las variables x1, x2, x3, y, por filas.

z <- c(80, 27, 89 ,42, 80, 27 ,88, 37 ,75 ,25 ,90 ,37 ,62 ,24 ,87 ,28 ,62 ,22 ,87 ,18 ,62 ,23 ,87 ,18 ,62 ,24 ,93 ,19 ,62 ,24 ,93 ,20 ,58 ,23 ,87, 15, 58, 18, 80, 14, 58, 18 ,89 ,14 ,58,17 ,88, 13, 58, 18, 82, 11, 58, 19 ,93, 12, 50, 18, 89, 8, 50, 18, 86, 7,50 ,19,72,8,50,19,79,8, 50,20,80, 9,56,20,82, 15,70,20,91,15)

x <- matrix(z, ncol=4,byrow=T)

La matriz con los datos es:

x; y <- x[,4]; x[,4] <- 1

dfvals( )

```
zr<- hylmse(x,y, nq, ik=1, iopt=3, intch=1)
theta1 <- zr$theta; xmin1 <- zr$xmin
zr <- hylmse(x, y, nq, ik=2, iopt=3,intch=1)
theta2 <- zr$theta; xmin2 <- zr$xmin
zr <- hylmse(x,y, nq, ik=1, iopt=1, intch=1)
theta3 <- zr$theta; xmin3 <- zr$xmin
# Los coeficientes estimados con ik=1, iopt=3, intch=1 son:
theta1
# Los coeficientes estimados con ik=2, iopt=3, intch=1 son:
theta2
# Los coeficientes estimados con ik=1, iopt=3, intch=1 son:
theta3
rs <- zr$rs
# El vector con los residuales es:
rs
rm(n,np,nq,z,x,y,theta1,xmin1,theta2,xmin2,theta3,xmin3,rs)
```

Salida en S-PLUS

```
> > # Los coeficientes estimados con ik=1, iopt=3, intch=1 son:
> theta1
[1] 0.73123991 0.36053634 -0.00959783 -34.43706894
> # Los coeficientes estimados con ik=2, iopt=3, intch=1 son:
> theta2
[1] 0.7217115164 0.3386301100 -0.0009179359 -34.1763229370
> # Los coeficientes estimados con ik=1, iopt=3, intch=1 son:
> theta3
[1] 0.727516413 0.364732981 0.002924738 -35.371856689
> # El vector con los residuales es:
> rs
 [1] 9.06245232 4.06537628 8.42657661 9.25779533 -0.01273727 -0.37747192
 [7] 0.24024582 1.24024582 -0.46740341 0.37673187 0.35040665 -0.28193283
```

[13] -2.62911987 -2.02602005 0.17053986 -0.82068253 -0.14447403 -0.16494751

[19] 0.46739960 2.09644699 -8.11510086

4.3.2 Subrutina HYLTSE

Algoritmo para calcular el estimador LTS.

Nombre del archivo en Microsoft Word: T-HYLTSE.

Ejemplo: Draper and Smith, p. 361 [4].

Se considera el modelo $y_i = x_{i1}\theta_1 + x_{i2}\theta_1 + x_{i3}\theta_p + \theta_4 + e_i$ donde $i = 1, \cdots 21$.

library ("robeth",T)

Ingresar cantidad de observaciones y cantidad de parámetros a estimar.

n <- 21: np <- 4; nq <- np+1

Ingresar los datos de las variables x1, x2, x3, y, por filas.

z <- c(80, 27, 89 ,42, 80, 27 ,88, 37 ,75 ,25 ,90 ,37 ,62 ,24 ,87 ,28 ,62 ,22 ,87 ,18 ,62 ,23 ,87 ,18 ,62 ,24 ,93 ,19 ,62 ,24 ,93 ,20 ,58 ,23 ,87, 15, 58, 18, 80, 14, 58, 18 ,89 ,14 ,58,17 ,88, 13, 58, 18, 82, 11, 58, 19 ,93, 12, 50, 18, 89, 8, 50, 18, 86, 7,50 ,19,72,8,50,19,79,8, 50,20,80, 9,56,20,82, 15,70,20,91,15)

x <- matrix(z, ncol=4,byrow=T)

La matriz con los datos es:

x

y <- x[,4]; x[,4] <- 1

dfvals( )

zr<- hyltse(x,y, nq, ik=1, iopt=3, intch=1)

theta4 <- zr$theta; xmin4 <- zr$smin

zr <- hyltse(x, y, nq, ik=2, iopt=3,intch=1)

theta5 <- zr$theta; xmin5 <- zr$smin

zr <- hyltse(x,y, nq, ik=1, iopt=1, intch=1)

theta6 <- zr$theta; xmin6 <- zr$smin

Los coeficientes estimados con ik=1, iopt=3, intch=1 son:

theta4

Los coeficientes estimados con ik=2, iopt=3, intch=1 son:

theta5

Los coeficientes estimados con ik=1, iopt=3, intch=1 son:

theta6

rs <- zr$rs

El vector con los residuales es:

rs

rm(n,np,nq,z,x,y,theta4,xmin4,theta5,xmin5,theta6,xmin6,rs)

Salida en SPLUS

> > # Los coeficientes estimados con ik=1, iopt=3, intch=1 son:

> theta4

[1] 0.727516413 0.364732951 0.002924707 -35.357639313

> # Los coeficientes estimados con ik=2, iopt=3, intch=1 son:

> theta5

[1] 0.71541142 0.34502181 0.01235466 -35.07965851

> # Los coeficientes estimados con ik=1, iopt=3, intch=1 son:

> theta6

[1] 0.727516413 0.364732981 0.002924738 -35.357643127

> rs <- zr$rs

> # El vector con los residuales es:

> rs

[1] 9.04823875 4.05116272 8.41236305 9.24358177 -0.02695084 -0.39168549

[7] 0.22603226 1.22603226 -0.48161697 0.36251831 0.33619308 -0.29614639

[13] -2.64333344 -2.04023361 0.15632629 -0.83489609 -0.15868759 -0.17916107

[19] 0.45318604 2.08223343 -8.12931442

4.3.3 Subrutina HYSEST

Subrutina HYSEST

Algoritmo para calcular el S-estimador.

Nombre del archivo en Microsoft Word: T-HYSEST.

Ejemplo: Draper and Smith, p. 361 [4].

Se considera el modelo $y_i = x_{i1}\theta_1 + x_{i2}\theta_1 + x_{i3}\theta_p + \theta_4 + e_i$ donde $i = 1, \cdots 21$.

```
library ("robeth",T)
# Ingresar cantidad de observaciones y cantidad de parámetros a estimar.
n <- 21;   np <- 4;   nq <- np+1
# Ingresar los datos de las variables x1, x2, x3, y, por filas.
z <- c(80, 27, 89 ,42, 80, 27 ,88, 37 ,75 ,25 ,90 ,37 ,62 ,24 ,87 ,28 ,62 ,22 ,87 ,18 ,62 ,23 ,87
,18 ,62 ,24 ,93 ,19 ,62 ,24 ,93 ,20 ,58 ,23 ,87, 15, 58, 18, 80, 14, 58, 18 ,89 ,14 ,58,17 ,88, 13, 58,
18, 82, 11, 58, 19 ,93, 12, 50, 18, 89, 8, 50, 18, 86, 7,50 ,19,72,8,50,19,79,8, 50,20,80, 9,56,20,82,
15,70,20,91,15)
x <- matrix(z, ncol=4,byrow=T)
# La matriz con los datos es:
x
y <- x[ ,4]; x[ ,4] <- 1
dfvals( )
z <- dfrpar(x,´S')
 z <- ribetu(y)
 zr <- hysest(x,y, nq, iopt=3, intch=1)
 theta7 <- zr$theta[1:np]; xmin7 <- zr$smin
 zr <- hysest(x,y, nq, iopt=1, intch=1)
 theta8 <- zr$theta[1:np]; xmin8 <- zr$smin
# Los coeficientes estimados con iopt=3, intch=1 son:
theta7
# Los coeficientes estimados con iopt=1, intch=1 son:
theta8
rs <- zr$rs
# El vector con los residuales es:
rs
rm(n,np,nq,z,x,y,theta7,xmin7,theta8,xmin8,rs)
```

Salida en SPLUS

```
> # Los coeficientes estimados con iopt=3, intch=1 son:
> theta7
```

```
[1] 0.84954464 0.43052724 -0.07361189 -36.91865158
> # Los coeficientes estimados con iopt=1, intch=1 son:
> theta8
[1] 0.84954435 0.43054006 -0.07362577 -36.91772461
> rs <- zr$rs
> # El vector con los residuales es:
> rs
 [1] 5.88228846 0.80866277 6.06471634 8.31845570 -0.82046443 -1.25100446
 [7] -0.23978996 0.76021004 -0.85282713 -0.21550721 0.44712469 -0.19596103
[13] -3.06825566 -1.68891227 1.24347949 0.02260216 -0.43869862 0.07668174
[19] 0.71976745 1.76975286 -9.46123600
```

Apéndice A

Programación y convenciones en ROBETH

Software ROBETH

El software ROBETH es un completo paquete de porgramas que nos permite resolver una amplia clase de procedimientos basados en M-estimadores y estimadores con alto punto de ruptura, incluyendo regresión robusta, test robustos para hipótesis lineales y covarianza robusta. ROBETH está escrito en ANSI-FORTRAN 77 y el manual que se presenta en este apéndice, basado en el libro de Marazzi and Randriamiharisoa [18], da algunas pautas para la programación y convenciones respecto de las subrutinas.

Estructura del ROBETH

El ROBETH agrupa las subrutinas en 15 archivos de acuerdo a su propósito.

Llamada general de un programa

1. Las *subrutinas principales* del ROBETH están agrupadas en 10 *archivos principales* que se dan a continuación:

   LCMAIN.F, RGMAIN.F, WGMAIN.F, KVMAIN.F, AEMAIN.F,
   TSMAIN.F, HBMAIN.F, CVMAIN.F, MXMAIN.F, GLMAIN.F.
   Usualmente al programar se llaman las subrutinas de los archivos principales.

2. Las sub-rutinas principales hacen uso de *subrutinas auxiliares*, agrupadas en 10 archivos:

LCAUXI.F, RGAUXI.F, WGAUXI.F, KVAUXI.F, AEAUXI.F,
TSAUXI.F, HBAUXI.F, CVAUXI.F, MXAUXI.F, GLAUXI.F.

3. Un grupo especial de subrutinas se usa para la integración numérica, tomadas de la librería QUADPACK adaptadas para ROBETH, y agrupadas en QDAUXI.F.

4. Las subrutinas principales y auxiliares son las que llaman a las *funciones peso* y las de *utilidad* o *de servicio* que están agrupadas en WGTFUN.F y COMUTL.F respectivamente. También las subrutinas principales y las auxiliares llaman a las *subrutinas de monitoreo*, agrupadas en SRDPRT.F.

5. En DFCOMN.F están agrupadas dos subrutinas adicionales: "handling defaults" y "common blocks parameters".

Rasgos especiales de programación y convenciones
Convenciones para los nombres de las subrutinas

a. El nombre de cualquier subrutina que pertenece a un archivo principal comienza con la misma letra que el archivo por ejemplo: RYWALG pertenece a RGMAIN.F, MYMVLM pertenece a MXMAIN.F.

b. La segunda letra será I, F, Y ó T de la siguiente manera:

   i. I, si son subrutinas que se deben llamar al comienzo de la programación (por ejemplo, el cálculo de un valor inicial).

   ii. F, si son subrutinas que se llaman al final de la programación (por ejemplo, el cálculo de los coeficientes de la matriz de covarianza).

   iii. Y, si usan algoritmos iterativos.

   iv. T, en todos los otrs casos.

Convenciones para la descripción de los parámetros

a. La letra **I** se usa para entrada de datos o parámetros (**Imput**).

b. La letra **O** se usa para salida de datos o parámetros (**Output**).

c. La letra **W** para vectores o matrices de trabajo.

d. La letral **E** para funciones externas.

e. Caracteres como [**MXT**] o [ ] se usan para generar la interface con el S-PLUS y toman valores por defecto.

Principales bloques y parámetros comunes

- Las funciones peso suministradas en WGTFUN.F se pueden usar como argumentos (externos) en muchas de las subrutinas de ROBETH. Los parámetros de estas funciones de peso pertenecen a "COMMONS" que se debe definir en el programa principal. Una extensa lista de estos "COMMONS" se encuentra en la Tabla A.1.
- Las subrutinas LYHALG, RILARS, RYNALG, RYHALG, RYWALG, RYSALG, RYSIGM requieren de COMMON/BETA/BETA,BETO. El valor de BETA y/o BETO se debe definir en el programa principal.

Subrutinas auxiliares para asignar valores por defecto

El DFCOMN.F contiene las subrutinas en FORTRAN: DFCOMN, COMVAL, DFRPAR, DFCOMN y CONVAL que son usadas por la interfase en S-PLUS para asignar valores a los elementos comunes listados en la Tabla A.1 y obtener los valores de dichos parámetros.

La interfase con el S-PLU. Rasgos generales

En esta parte se dará información respecto de los rasgos generales de la interfase, su implementación y su uso. Se supone que el lector es conocedor de las nociones básicas de los lenguajes S y S-PLUS, en caso contrario puede consultar los manuales del S-PLUS [31] [32] [33]. Para más información se recomienda leer READ.ME. del diskette del libro de Marazzi and Randriamiharisoa [18].

La interfase principal suministrada por INTRFC.S es un conjunto de funciones S-PLUS que invocan a las subrutinas ROBETH. Hay una función S para cada subrutina

descripta en ROBETH. Las características de las S funciones se pueden obtener de la descripción de las rutinas FORTRAN, de acuerdo a las reglas que se dan a continución.

Tabla A.1	Bloques comunes	parámetros comunes
Nombre	Variables comunes	usadas en
/PSIPR/	IPSI, CPSI,	PSI, RHO, PSP, CHI,
	H1, H2, H3, XK, DCHI	PSIA, PHOA, PSPA, CHIA, DFRPAR, RYSALG, AIREFO, AIREFQ, GINTAC, MYHBHE, MIRTSR, MFRAGR
/UCVPR/	IUCV, A2, B2, CHK, CKW, BB, BT, CW	UCV, UPCV, VCV, VPCV, WCV, WPCV, UCVA, UPCVA, VCVA, VPCVA, WCVA, WPCVA, DFPAR, GINTAC, WYFCOL
/BETA/	BETA, BETO	LIBETH, LIBETU, LIBETO, LYHALG, DFRPAR LYTAU2, RIBETH, RIBETU, RIBETO, RILARS, LYHALG, RYWALG, RYSALG, RYNALG, RYSIGM, HYSEST, MYHBHE, DFRPAR
/WWWPR/	IWW	DFRPAR, WWW,WWWA

Reglas para la obtención de funciones S de las subrutinas ROBETH.

Nombre de las funciones

Las funciones S tiene el mismo nombre que su correspondiente subrutina ROBETH. Los nombres de las funciones S serán escritas en letras minúsculas, excepto *Random* y *Dbinom*.

Nombre de los argumentos

Los argumentos de las funciones S tiene el mismo nombre y significado que los correspondientes argumentos (parámetros) de las subrutinas ROBETH. Los argumentos de las funciones S serán escritos en letras minúsculas.

Listado de los argumentos

El orden de los argumentos en las S funciones es igual al orden del ROBETH. El orden de los argumentos en las subrutinas FORTRAN es mayor.

Argumentos que no deben aparecer en las funciones S

a- La dimensión de los vectores y arreglos conocidos en el entorno de S-PLUS.

b- Argumentos superfluos, como NCOV=NPx(NP+1)/2 cuando NP es conocido en el entorno del S-PLUS (ver por ejemplo subrutina RYWALG, Cap. 2).

c- Argumentos redundantes, como MDXN cuando N es conocido en el entorno del S-PLUS. La cota N es usada en este caso (ver MDX en la RYWALG, Cap. 2).

d- Parámetros de salida y arreglos de trabajo.

Los argumentos del tipo a, b, y c serán reconocidos en las subrutinas por la presencia del corchete ([ ]) en la descripción del parámetro y **no** se deben escribir en el S-PLUS. Los parámetros I/O tales como el arreglo XT en la subrutina RICLLS del Cap.2, son opcionales; si se dan, deben considerarse tanto en la entrada (Imput) como en la salida (Output); si no se dan, aparecen sólo en la salida.

Valores respuesta

- Los valores que se obtienen de las funciones S a través de las subrutinas ROBETH constituyen una lista donde los parámetros de salida se obtienen de las subrutinas por el operador **$**. Los nombres de los elementos de la lista con los parámetros de salida de las subrutinas deben escribirse en letras minúsculas. Los valores de salida o respuesta de una función son los valores de la función FORTRAN.

Argumentos y parámetros asignados por defecto

- Muchos argumentos y parámetros se pueden asignar por defecto cuando se usa el S-PLUS. Por ello debe llamarse la función **dfvals** al comienzo.

- Los valores por defecto son indicados entre corchetes ([ ]) en las subrutinas, como por ejemplo [TLO] es un parámetro por defecto que se da en la subrutina RYWALG descripta en el Cap. 2; su valor es ($1.E^{-3}$) y es asignado por dfvals.
- Los parámetros que tienen una asignación por defecto se pueden omitir en el llamado de las funciones S.
- Los valores de las funciones externas como PSI, CHI, RHO (ver, por ejemplo, la subrutina RYWALG en el Cap. 2) son asignadas por defecto. Esos valores que se asignan por defecto se pueden ver en el monitor de la computadora tipeando el nombre de la función S sin los parámetros.

Valores por defecto y bloques comunes

DE archivo DEFLTS.S contiene las S-funciones: **dfvals**, **dfcomn**, **comval** y **dfrpar**.

El objetivo de la función **dfvals** es crear una lista de objetos, asignándole valores por defecto para muchos de los escalares usados en ROBETH. Dicha lista, llamada **.dFv** contiene valores que se sugieren por defecto y que se dan en la Tabla A.2.

Tabla A.2 Valores que se sugieren por defecto

aa2 = 0	fff = 0.0	icv = 1	isr=1	mxx=50	upr=10
apr = 1.25	fu1 = 1.0	ik1 = 2	itc=1	ntm=0	xfd=1.0
apr = 1.25	gma = 1.0	ilc = 1	ite=1	tli=0.0001	
bb2 = 9	ia1 = 1	ilg = 1	ith=1	tlm=0.001	
ccc = 1.345	ia2 = 1	ilm = 1	itw=0	tlo=0.001	
esp = 0.1	ial = 1	ini=2	iug=1	tlr=0.001	
fb1 = 1.0	ic1 = 1	iop=2	iwg=2	tls=0.001	
fct = 0.0	ich = 1	ipo=1	ix1=1	tlu=$1.e^{-011}$	
ff1 = 0.0	icn = 1	ipt=1	iy1=1	tlv=-6.9078	
ffc = 0.0	ics = 1	isq=1	msx=30	tua=$1.e^{-006}$	

Los valores por defecto se pueden extraer de la lista: por ejemplo, el valor **aa2** se obtiene escribiendo la expresión **.dFv$aa2**.

Con una convención especial se indican en las subrutinas los valores por defecto, por ejemplo, [*DDD*] para el argumento **D** en LIBETH (Cap. 1) informa que por defecto

el valor de **d**(S-PLUS) será proporcionado por **ddd**. De igual manera, **tlo** [TLO] en la descripción de subrutinas que usan algoritmos iterativos será proporcionado por tlo. El usuario puede modificar los valores que se han dado por defecto.

El objetivo de la función **dfcomn** es dar valores corrientes de los parámetros incluidos en la lista de bloques comunes (ver Tabla A.2). La función **dfrpar** es la versión S-PLUS de la subrutina DFRPAR en FORTRAN. El conjunto DFRPAR otorga valores por defecto a muchos parámetros de ROBETH usados en problemas de regresión de acuerdo al tipo de estimadores y los escribe en bloques apropiados. La función **comval** da los valores corrientes de los parámetros incluidos en la Tabla A.1.

Subrutinas de Sevicio: COMUTL

Visión general del COMUTL

Las rutinas de servicio utilizadas en el ROBETH están agrupadas en un subconjunto con el nombre de COMUTL. Por medio de estas subrutinas el usuario podrá particularmente:

1. Ejecutar las operaciones más comunes de vectores y matrices.
2. Evaluar diversas funciones de distribución usadas en el análisis de datos robustos con ROBETH.

Composición de COMUTL

A continuación se presenta una lista de las subrutinas incluidas en COMUTL. Muchas de ellas, tomadas de otras librerías, fueron adaptadas y agregadas al ROBETH. Por ejemplo, rutinas del álgebra lineal básica fueron copiadas de LINPACK y transformaciones elementales de Householder del Lawsonand Hanson [15]. Subrutinas para el cálculo de las funciones de distribución y otras funciones fueron tomadas de softwares de dominio público. Éstas aportan un amplio conjunto de herramientas para muchos análisis usados en ROBETH.

Aritmética de vectores y matrices

MSS / MSSD	Multiplica una matriz simétrica por otra simétrica.
MFF / MFFD	Multiplica una matriz completa por otra completa.
MSF / MSFD	Multiplica una matriz simétrica por otra completa.
MSF1 / MSF1D	Multiplica una matriz simétrica por otra completa cuando el resultado es una simétrica.
MFY / MFYD	Multiplica una matriz completa por un vector.
MTT1 / MTT1D	Multiplica una matriz triangular superior por su traspuesta.
MTT2 / MTT2D	Multiplica una matriz triangular inferior por su traspuesta.
MTT3 / MTT3D	Multiplica una matriz triangular por otra triangular.
MLY / MLYD	Multiplica una matriz triangular inferior por un vector
MTY / MTYD	Multiplica una matriz triangular superior por un vector.
MINV / MINVD	Invierte una matriz triangular.
SCAL / SCALD	Multiplica un vector por un escalar.
DOTP / DOTPD	Efectúa el producto escalar de dos vectores
NRM2 / NRM2D	Cálculo de la norma euclidiana de un vector.
XSY / XSYD	Evalúa la forma cuadrática.
MCHL / MCHLD	Descomposición de Cholesky de una matriz simétrica.
MHAT	Calcula los elementos-diagonal de la matriz sombrero.
ADDC	Adiciona un vector columna a la matriz de diseño transformada. y computa su QR-descomposición.
RMVC	Quita un vector columna de la matriz de diseño transformada. y computa su QR-descomposición.
H12 / H12D	Construye y/o aplica una transformación elemental de Householder.

Ordenamiento y permutaciones

SRT1	Ordena las componentes de un vector en orden ascendente.
SRT2	Ordena las componentes de un vector en orden ascendente y permuta las componentes de otro vector adecuadamente.
SWAP / SWAPD	Intercambia dos vectores.
EXCH / EXCHD	Intercambia dos columnas de una matriz simétrica.
PERMC	Permuta las columnas de una matriz por medio de su trasposición.
PERMV	Permuta los elementos de un vector.

Distribuciones y funciones de densidad

BINPRD	Distribución Binomial.
POISSN	Distribución de Poisson.
XERF	Función de densidad de la N(0,1).
XERP	Densidad de la norma de un vector N(0,1) con p componentes.
GAUSS / GAUSSD	Función de distribución acumulativa de la N(0,1).
QUANT	Inversa de la función acumulativa de la N(0,1).
CHISQ	Acumulativa de la función de distribución χ^2.
CQUANT	Inversa de la distribución χ^2.
RUBEN	Función de densidad y de distribución de una combinación lineal de variables aleatorias χ^2.
PROBST	Función de distribución acumulativa de la t de Student.
TQUANT	Inversa de la distribución acumulativa de la t de Student.
FCUM	Función de distribución acumulativa de la F.

Funciones especiales

CERF/CERFD	Complemento de la función error.
INGAMA	Función incompleta Gamma.
NLGM	Logaritmo de la función Gamma en el punto n/2, n entero.
LGAMA	Logaritmo de la función Gamma en el punto x, con x >0.

Otras

RANDOM	Genera números aleatorios de la uniforme $(0, 1)$.
RGFL	Resuelve la ecuación $y = f(x)$.
LMDD	Cálculo de la mediana y del desvío absoluto mediano.
FSTORD	A un vector y de n componentes, establece el j-ésimo estadístico de orden.

Reglas de almacenamiento de matrices y vectores

1. Una *matriz completa* F(n x m) es almacenada por columnas en un arreglo bidimensional F(N,M) con M=m y N=n de la siguiente forma:

$$\text{F}\rightarrow \begin{pmatrix} \text{F(1,1)} & \text{F(1,2)} & \cdots & \text{F(1,M)} \\ \text{F(2,1)} & \text{F(2,2)} & \cdots & \text{F(2,M)} \\ \cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots \\ \text{F(N,1)} & \text{F(N,2)} & \cdots & \text{F(N,M)} \end{pmatrix}.$$

2. Una *matriz triangular superior* T(n x n) es almacenada por columnas en un arreglo unidimensional T(NCOV) con NCOV= n *(n+1)/2 de la siguiente forma (*modo compacto de almacenamiento*):

$$\text{T}\rightarrow \begin{pmatrix} \text{T(1)} & \text{T(2)} & \text{T(4)} & \cdots & \text{T(NCOV-N+1)} \\ \cdots & \text{T(3)} & \text{T(5)} & \cdots & \cdots\cdots\cdots \\ \cdots & \cdots & \text{T(6)} & \cdots & \cdots\cdots\cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots\cdots\cdots \\ \cdots & \cdots & \cdots & \cdots & \text{T(NCOV)} \end{pmatrix}$$

3. Una *matriz triangular inferior* T(n x n) es almacenada por filas en un arreglo unidimensional T(NCOV) con NCOV= n * (n+1)/2 de la siguiente forma (*modo compacto de almacenamiento*):

$$\text{T}\rightarrow \begin{pmatrix} \text{T(1)} & \cdots & \cdots & \cdots & \cdots\cdots \\ \text{T(2)} & \text{T(3)} & \cdots & \cdots & \cdots\cdots \\ \text{T(4)} & \text{T(5)} & \text{T(6)} & \cdots & \cdots\cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots\cdots \\ \text{T(NCOV-N+1)} & \cdots & \cdots & \cdots & \text{T(NCOV)} \end{pmatrix}$$

4. Una *matriz simétrica* S(n x n) es almacenada en un arreglo unidimensional S(NCOV) con NCOV= n * (n+1)/2 de la siguiente forma (*modo compacto o simétrico de almacenamiento*):

$$\text{S}\rightarrow \begin{pmatrix} S(1) & S(2) & S(4) & \cdots & S(NCOV-N+1) \\ S(2) & S(3) & S(5) & \cdots & \cdots\cdots\cdots.. \\ S(4) & S(5) & S(6) & \cdots & \cdots\cdots\cdots.. \\ S(NCOV-N+1) & \cdots & \cdots & \cdots & S(NCOV) \end{pmatrix}$$

5. Las componentes de un vector y de longitud n son generalmente almacenadas en los n primeros lugares de un arreglo unidimensional Y(NY) con NY$\geq n$. Ocasionalmente son almacenados con un incremento IYE de la siguiente forma:

$y_1, y_2, ..., y_n \rightarrow$ Y(1), Y(IYE+1),...,Y((N-1)*(IYE+1)) si IYE > 0,

$y_1, y_2, ..., y_n \rightarrow$ Y((N-1)* | IYE | +1), $\cdots$, Y(| IYE | +1),Y(1) si IYE < 0.

En ambos casos N=n, y NY $\geq$ ((N-1)*|IYE|+1)).

6. Algunas veces la i-ésima fila x_i^T de la matriz completa X(n x m), almacenada en un arreglo X(N,M) (con M=m y N=n), debe ser asignada a un arreglo unidimensional Y(NY) tal que aparezca en la lista de los parámetros de la subrutina. En este caso, la fila esta nombrada por sus primeros elementos X(I,1) con I=i, por lo tanto, de acuerdo con las reglas de almacenamiento FORTRAN, el arreglo

$$\begin{pmatrix} \cdots & \cdots & \cdots\cdots & \cdots & \cdots\cdots\cdots \\ \cdots & \text{Y(N)} & \cdots\cdots & \cdots & \cdots\cdots\cdots \\ \text{Y(1)} & \text{Y(N+1)} & \text{Y(2*N+1)} & \cdots & \text{Y((M-1)*N+1)} \\ \text{Y(2)} & \text{Y(N+2)} & \text{Y(2*N+2)} & \cdots & \cdots\cdots\cdots \\ \cdots & \cdots.. & \cdots\cdots & \cdots & \cdots\cdots\cdots \end{pmatrix}$$

contiene los elementos

$$\begin{pmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & x_{i-1,2} & \cdots & \cdots & \cdots \\ x_{i,1} & x_{i,2} & x_{i,3} & \cdots & x_{i,n} \\ x_{i+1,1} & x_{i+1,2} & x_{i+1,3} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix}$$

de X. Por lo tanto los m elementos de x_i^T seran ordenados en

Y(1), Y(IYE+1), Y(2*IYE+1),Y(3*IYE+1),...,Y(NY),

donde IYE=N y NY=(M-1)*N+1. Notemos que los primeros *i-1* elementos de la primera columna de X y los *n-i* últimos elementos de la última columna no son almacenados en Y.

7. Algunas veces la j-ésima columna de la matriz completa X(n x m), almacenada en un arreglo X(N,M) (con M=m y N=n), debe ser asignada a un arreglo unidimensional Y(NY) tal que aparezca en la lista de los parámetros de la subrutina. En este caso, la columna esta nombrada por sus primeros elementos X(1,J) con J=j, por lo tanto, de acuerdo con las reglas de almacenamiento FORTRAN, sus elementos son almacenados en $Y(1), Y(2), ..., Y(N)$. La condición NY$\geq n$ se debe cumplir.

El software ROBETH posee dos versiones disponibes: la de simple y la de doble precisión para las subrutinas de utilidad y para la aritmética de vectores y matrices. Las subrutinas de simple precisión toman los parámetros y almacenan sus resultados en simple precisión. Cualquier producto por escalares se ejecuta en doble precisión, aún cuando la subrutina esté en simple precisión.

Apéndice B

Cómo utilizar el ROBETH

A continuación se da un detalle de cuáles fueron los pasos seguidos para poner a punto las subrutinas, mediante el ejemplo de la subrutina MSS.

a) Escribir el programa en Microsoft Word donde se debe especificar:

- El enlace con ROBETH mediante: **library("robeth",T)** Esta conexión se debe ingresar antes de la llamada de los parámetros.
- Los parámetros de entrada, indicados con Imput(**I**). Observar que **nn** y **mcd** no se ingresan, según se vió en "Argumentos que no deben aparecer en las funciones S". Para el caso de nuestro ejemplo escribir:

$$\mathbf{n} <- \mathbf{4}$$
$$\mathbf{a} <- \mathbf{c(9, 15, -2, -4, 3, 5, 7, 1, -8, 6)}$$
$$\mathbf{b} <- \mathbf{c(2, 0, -3, 5, -2, 1, 3, 0, 1, 4)}$$

- Los comentarios, que deben comenzar con el símbolo #, por ejemplo: # Producto de dos matrices simétricas A(nxn) por B(nxn).
- El llamado a la subrutina con: **s<-mss(a,b,n)**
- Los parámetros de salida, indicados con Output(**O**).

$$\mathbf{respuesta} <- \mathbf{s\$respuesta}$$
$$\mathbf{respuesta}$$

b) Copiar con el mouse, el programa escrito en Microsoft Word.

c) Una vez abierto el programa S-PLUS, pegar el programa realizado en Word.

Ejemplo:

Subrutina MSS (simple precisión)

Producto de dos matrices simétricas A(nxn) por B(nxn). (A es simétrica si $A = A^T$).

Ejemplo: A(4x4)·B(4x4)=C(4x4)

$$\# \underbrace{\begin{bmatrix} \mathbf{9} & \mathbf{15} & \mathbf{-4} & \mathbf{7} \\ 15 & \mathbf{-2} & \mathbf{3} & \mathbf{1} \\ -4 & 3 & \mathbf{5} & \mathbf{-8} \\ 7 & 1 & -8 & \mathbf{6} \end{bmatrix}}_{\text{compact storage mode}} \cdot \underbrace{\begin{bmatrix} \mathbf{2} & \mathbf{0} & \mathbf{5} & \mathbf{3} \\ 0 & \mathbf{-3} & \mathbf{-2} & \mathbf{0} \\ 5 & -2 & \mathbf{1} & \mathbf{1} \\ 3 & 0 & 1 & \mathbf{4} \end{bmatrix}}_{\text{compact storage mode}} = \begin{bmatrix} \mathbf{19} & \mathbf{-37} & \mathbf{18} & \mathbf{51} \\ \mathbf{48} & \mathbf{0} & \mathbf{83} & \mathbf{52} \\ -7 & -19 & \mathbf{-29} & \mathbf{-39} \\ -8 & 13 & 31 & \mathbf{37} \end{bmatrix}$$

```
library(robeth,T)
# Ingresar el orden (n) de la matriz A y B.
n <- 4
# Ingresar los elementos de A y B, en el sentido de las columnas
# en vectores a y b de dimensión (nx(n+1)/2), "compact storage mode".
a <- c(9,15,-2,-4,3,5,7,1,-8,6)
b <- c(2,0,-3,5,-2,1,3,0,1,4)
s <- mss(a,b,n)
respuesta <- s$c
respuesta
rm(a,b,n,respuesta)
```

Salida en S-PLUS

```
> respuesta
[,1] [,2] [,3] [,4]
[1,] 19 -37 18 51
[2,] 48 0 83 52
[3,] -7 -19 -29 -39
[4,] -8 13 31 37
```

Observación: Los ejemplos numéricos para probar las subrutinas de servicio referentes a la aritmética de vectores y matrices fueron tomados del Jealy [11] y del Johnson and Wichern [12]; y para las distribuciones y funciones de densidad del Kendall and Stuart [13] [14], Rohatgi [24] [25], Rice [23] y León García [17].

Bibliografía

[1] H. Bunke and O. Bunke. *Nonlinear Regression: Functional Relations and Robust Methods.* John Wiley & Sons, New York, 1989.

[2] O. Bustos. Estimação robusta no modelo de posicão. *XIII Coloquio Brasileiro de Matemática*, Notas del curso, Río de Janeiro, 1981.

[3] A. J. Dobson. *An Introduction to Generalized Linear Models.* Chapman and Hall, New York, 1999.

[4] N. R. Draper and H. Smith. *Applied Regression Analysis.* John Wiley & Sons, New York, 1981.

[5] D. J. Finney. *Statistical Method in Biological Assay.* Charles Griffin, 3rd Ed., London, 1978.

[6] M. P. Guillemin, D. Baur, and A. Marazzi. Human exposure to styrene, part iv, industrial hygiene investigations and biological monitoring in the polyester industry. *Int. Arch. Occup. Environ. Health*, 51:139–150, 1982.

[7] F. R. Hampel. A general qualitative definition of robustness. *The Annals of Mathematics Statistics*, 42:1887–1896, 1971.

[8] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics: The apprroach based on influence functions.* John Wiley & Sons, New York, 1986.

[9] J. L. Hodges. Efficiency in normal samples and tolerance of extreme values for some estimates of location. *Proc. Fifth Berkeley Symp. Math. Stat. Probab.*, 1(1), 1967.

[10] P. J. Huber. *Robust Statistics.* John Wiley & Sons, New York, 1981.

[11] M. J. R. Jealy. *Matrices for Statistics*. Oxford University Press, New York, 1986.

[12] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis*. Prentice-Hall, New Jersey, 1992.

[13] M. G. Kendall and A. Stuart. *The Advanced Theory of Statistics. Volume 1: Distribution Theory*. Hafner Publishing Company, New York, 1969.

[14] M. G. Kendall and A. Stuart. *The Advanced Theory of Statistics. Volume 2: Inference and Relationship*. Hafner Publishing Company, New York, 1973.

[15] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*. Prentice-Hall, New York, 1974.

[16] E. I. Lehmann. *Nonparametrics: Statistical Methods Based on Ranks*. Holden-Day, Inc., San Francisco, 1975.

[17] A. Leon-García. *Probability and Random Processes for Electrical Engineering*. Addison-Wesley, New York, 1990.

[18] A. Marazzi and J. J. A. Randriamiharisoa. *Algorithms, Routines, and S Functions for Robust Statistics*. Chapman and Hall, New York, 1993.

[19] A. Marazzi, C. Ruffieux, and A. Randriamiharisoa. Robust regression in biological assay: application to the evaluation of alternative experimental techniques. *Experientia*, 44(2), 1988.

[20] R. A. Maronna and V. J. Yohai. The breakdown point of simultaneous general m-estimate of regression and scale. *Journal of the American Statistical Association*, 86(415):699–703, 1991.

[21] J.Ñeter, M. H. Kutner, C. J. Nachtsheim, and W. Wasserman. *Applied Linear Regression Models*. Irwin, New York, 1990.

[22] J. García Peña. Análisis de regresión con aplicaciones. *VI Coloquio del Departamento de Matemáticas. Centro de Investigaciones y de Estudios Avanzados del IPN*, (Notas del curso). México, 1989.

[23] J. A. Rice. *Mathematical Statistics and Data Analysis*. Duxbury Press, U.S.A., 1995.

[24] V. K. Rohatgi. *An Introduction to Probability Theory and Mathematical Statistics*. John Wiley & Sons, New York, 1976.

[25] V. K. Rohatgi. *Statistical Inference*. John Wiley & Sons, New York, 1984.

[26] P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

[27] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[28] G. A. F. Seber. *Linear Regression Analysis*. John Wiley & Sons, New York, 1977.

[29] G. A. F. Seber. *Multivariate Observations*. John Wiley & Sons, New York, 1984.

[30] D. L. Souvaine and J. M. Steele. Time and space efficient algorithms for least median of squares regresssion. *Journal of the American Statistical Association*, 82(399):794–801, 1987.

[31] *S-PLUS 2000 Guide to Statistics*. volume 1, 2. Data Analysis Products Division, MathSoft, Seattle, Washington, 1999.

[32] *S-PLUS 2000. Professional Edition for Windows, Release 3*. MathSoft Inc., 2000.

[33] *S-PLUS 2000 Programmer's Guide*. Data Analysis Products Division, MathSoft, Seattle, Washington, 1999.

[34] P. Sprent. *Applied Nonparametric Statistical Methods*. Chapman and Hall, London, 1989.

[35] W. Venables and B. D. Ripley. *Modern Applied Statistics with S-PLUS*. Springer-Verlag, New York, 1997.

[36] V. J. Yohai. Regresión robusta. *Sociedad Argentina de Estaística*, Cuaderno número 1, 1983.

[37] V. J. Yohai. High breakdown point and high efficiency robust estimates for regression. *The Annals of Statistics*, 15(2):642–656, 1987.

[38] V. J. Yohai. Notas sobre regresión robusta. *Sociedad Argentina de Estaística*, Cuaderno número 7, 1991.

[39] V. J. Yohai and R. H. Zamar. High breakdown-point estimates of regression by means of the minimization of an efficient scale. *Journal of the American Statistical Association*, 83(402):406–413, 1988.