UNIVERSIDAD DE BUENOS AIRES

Facultad de Ciencias Exactas y Naturales

Departamento de Matemática

Tesis de Licenciatura

Análisis Estadístico con el Método Bootstrap: Aplicaciones en Problemas de Regresión

Gaspard Kerner

Director: Dr. Pablo E. Verde

Septiembre de 2015

Indice

Lista de Figuras

Lista de Tablas

Capítulo 1

Introducción

La estadística es la ciencia del aprendizaje a través de la experiencia introduce Efron en su libro *An introduction to the bootstrap methods,1993*. Sus orígenes son inciertos aunque el siglo XX ha visto el mayor esplendor de esta ciencia a través de sus aplicaciones en diversos ámbitos como la biomedicina, la psicología, la economía, la epidemiología y una larga lista de otras disciplinas. De forma general, la estadística propone encontrar patrones en un *río de información confusa*. Para ello, surgen diversas cuestiones que deben ser analizadas: Cómo y qué información o datos elegir? Cómo analizar y resumir el análisis efectuado?

Este trabajo abarcará el estudio de problemas de inferencia, variabilidad de parámetros provenientes de distintas poblaciones tales como la media, la mediana u otras medidas de resumen más complejas como en el análisis de los coeficientes de un modelo de regresión.

El bootstrap como metodología para tratar problemas de inferencia estadística

Los métodos Bootstrap desarrollados por Efron (1979) permiten abarcar y tratar problemas de inferencia estadística de manera simple y efectiva a través del uso de técnicas de simulación. Si bien desde el invento de las computadoras las técnicas de simulación han sido utilizadas para resolver los problemas planteados por la estadística, uno de lo grande logros de Efron (1979) es haberles dado un marco general. Tal es así que desde la introducción del método Bootstrap han sido publicados miles de trabajos al respecto.

El método Bootstrap ha vivido un auge en las distintas disciplinas ligadas a su uso por las caracterísiticas que se desprenden de su forma y su puesta en práctica:

1. son métodos sencillos de usar y suelen ser comprendidos a través de un algoritmo de fácil aplicación.
2. son flexibles y se adaptan muy bien a cuestiones de la estadística tan disímiles como intervalos de confianza o problemas de datos faltantes.
3. tienen la ventaja en muchos casos de reemplazar cálculos complicados propuestos por la estadística clásica por simples simulaciones logrando un nivel de eficacia similar.

Objetivos del trabajo

El trabajo que aquí se presenta intentará por un lado, en un nivel elemental, hacer una revisión práctica de las técnicas bootstrap. Se tratará en este marco entender la construcción de intervalos de confianza así como la aplicación a problemas de regresión lineal múltiple y regresión logística. Se destacan dos contribuciones principales del trabajo:

1. El análisis de datos provenientes de la Facultad de Medicina de la Universidad de Duesseldorf.
2. La implementación en R de programas asociados a problemas metodológicos.

Se describirá asimismo como aplicar el método bootstrap para el cálculo de intervalos de confianza, la selección de variables en regresiones, etc.

Resumen de los capítulos

La tesis está compuesta por 11 capítulos ordenados según un nivel de dificultad creciente. Aún así, si el lector lo desea, es posible leer capítulos separadamente. A pesar de algunas referencias entre los mismos, son prácticamente independientes. El Capítulo 2 realiza una descripción general del bootstrap, las generalidades de su aplicación y algunos ejemplos sencillos de aplicación como lo son el caso de la media, la mediana y el coeficiente de correlación usando la función *sample*() de R.

El Capítulo 4 buscará comprender el uso del bootstrap, los diversos algoritmos de remuestreo y los procedimientos de simulación en problemas de regresión lineal mientras que

el Capítulo 3 describirá diversos métodos para la construcción de intervalos de confianza en donde se destacan los intervalos percentil y BC_a. A su vez, en dichos capítulos, se presenta la librería *boot* de R para el cálculo de las réplicas bootstrap y otras características de la metodología.

El Capítulo 5 desarrollará métodos bootstrap para el cálculo del error de predicción de un modelo de regresión lineal, mientras que el Capíutlo 6 abarcará el problema de selección de variables en el mismo contexto. Se describirán los métodos AIC, $Validacion\ Cruzada$ y $Bootstrap\ mejorado$.

Se tratará el tema de bootstrap en regresión logśitica en el Capítulo 7 en el cual se analizarán medidas como la razón de chances y el error de predicción en clasificación. Los Capítulos 8 y 9 estudiarán ejemplos originales de datos reales en donde se pondrán en práctica los conocimientos y métodos descriptos en los capítulos previos. En el Capítulo 8 se analizará un conjunto de datos de anemia para el que el modelo de regresión lineal parece adecuado. Por otra parte, en el Capítulo 9 se analizará un caso de aplicación en investigación clínica relacionado con predictores de mortalidad para infecciones de tejidos blandos necrotizantes. En este caso, el modelo adecuado es un modelo de regresión logística. El Capítulo **??** presenta un resumen y una discusión del trabajo realizado y en el Capítulo 10 se dan los códigos de R utilizados a lo largo del trabajo y las tablas con los datos que no han sido mencionados en el texto. Se intercalará además de forma constante el texto y el código de R en todos los capítulos para un mejor seguimiento de los métodos.

Capítulo 2

Introducción al bootstrap

Los métodos bootstrap permiten abarcar problemas de inferencia estadística usando técnicas de simulación en computadoras. Estos procesos tienen una ventaja esencial: el analista puede analizar y tratar situaciones de alta complejidad en donde los cálculos teóricos son imposibles o en donde el tamaño muestral es demasiado pequeño. Estos métodos permiten a su vez obtener resultados rápidos cuando el contexto así lo requiere.

El algoritmo bootstrap en el caso más simple, es decir, en el caso univariado, se basa en un modelo de probabilidad simple. Se supone en un principio que el conjunto de observaciones $\{x_1, ..., x_n\}$ es la realización de una muestra aleatoria tomada de forma independiente e idénticamente distribuida a partir de una función de distribución desconocida F, es decir,

$$X_1, ..., X_n \sim_{iid} F. \tag{2.1}$$

A partir de este conjunto, se intenta inferir cierta o ciertas características de la población de origen. Formalmente, esto puede entenderse como la inferencia sobre un parámetro

$$\theta = t(F),$$

donde t es un funcional sobre un espacio de funciones de distribución. Como ejemplo sencillo puede pensarse en θ como la media poblacional. En ese caso, se tiene:

$$t(F) = \int x dF.$$

En el proceso de inferencia, se considera un estimador del parámetro, que se denotará $\widehat{\theta}$ y que se define en función de la muestra, i.e,

$$\widehat{\theta} = \delta(x_1, ..., x_n).$$

El estimador $\widehat{\theta}$ se supone, además, simétrico en el sentido que $\delta(x_1, ..., x_n) = \delta(x_{\pi_1}, ..., x_{\pi_n})$ donde $(\pi_1, ..., \pi_n)$ es una permutación de los índices $1, ..., n$. En particular, si $\widehat{\theta} = t(F_n)$ donde F_n es la distribución empírica asociada a $x_1, ..., x_n$, $\widehat{\theta}$ cumple este supuesto. La elección del estimador es evidentemente un aspecto clave en la estimación. Una propuesta posible es considerar la distribución empírica de los datos, denotada F_n, y tomar

$$\widehat{\theta} = t(F_n).$$

Esto se conoce como la estimación *plug-in* del parámetro y como se verá, es un aspecto importante de la metodología bootstrap. Retomando el ejemplo de la media poblacional se tiene para el caso que

$$\widehat{\theta} = t(F_n) = \frac{\sum_{i=1}^{n} x_i}{n},$$

con desvío $\sigma(F) = \sigma/\sqrt{n}$. En general, el valor de $\sigma(F)$ es desconocido y difícil de calcular analíticamente. Es justamente en este problema en donde interviene el procedimiento bootstrap ya que permite estimar la precisión del estimador $\widehat{\theta}$. La noción de precisión incluye aquí conceptos tales la estimación del desvío estándar, el cálculo de intervalos de confianza, etc. Aún cuando en el caso de la media poblacional estas estimaciones pueden obtenerse de forma sencilla, en general, puede resultar útil tener una forma de estimar la distribución de $\widehat{\theta}$. Si bien se propuso la estimación plug-in de θ con la distribución empírica de los datos para el cálculo de $\widehat{\theta}$, existen diversas metodologías posibles. De forma general, $\widehat{\theta}$ se distribuye a partir de una función de distribución G que está determinada por F y δ, i.e., $G = G(\delta, F)$. El método bootstrap permite estimar la distribución de $\widehat{\theta}$ con un procedimiento que puede resumirse en dos pasos:

1. Estimar F por $\widehat{F}$.
2. Estimar G por $\widehat{G} = G\left(\delta, \widehat{F}\right)$.

La elección de $\widehat{F}$ es una de las claves del éxito del método y, de forma general, se conocen dos maneras de estimar F. Por un lado, cuando el desconocimiento sobre la distribución que ha generado el conjunto de datos es absoluto, se propone la elección de $\widehat{F} = F_n$, la distribución empírica de los datos. Los métodos bootstrap que involucren esta estimación de F se llamarán métodos no paramétricos. Por otro lado, si se supone que la distribución F pertenece a una familia paramétrica F_η donde η es un parámetro que se estima con $\widehat{\eta}$, entonces la estimación de F puede hacerse con $F = F_{\widehat{\eta}}$. Esto se conoce como el bootstrap paramétrico. Existen otras formas de estimar G, que no se desarrollarán en este trabajo, y que pueden encontrarse en Efron (1993) y en Davison y Hinkley (1997). En este capítulo, se definirá una de ellas conocida como Bootstrap Suavizado mientras que en el Capítulo 4 se describirá brevemente otra de ellas denominada Wild Bootstrap. Una vez estimada la distribución F se puede proceder con el algoritmo bootstrap. La distribución $\widehat{G}$ se conoce

como distribución bootstrap y si bien, en algunos casos es posible calcularla anlíticamente, se suele aproximar por simulación. El principio bootstrap puede describirse entonces de la siguiente manera:

1. Obtenga una gran cantidad B de muestras aleatorias de tamaño n a partir de la distribución $\widehat{F}$. Estas muestras se notarán $(x_1^*, ..., x_n^*)$ y se llamarán muestras bootstrap. Por ejemplo, en el caso en que $\widehat{F}$ es F_n, estas muestras deberán tomarse de forma aleatoria y con repetición a partir de la muestra de datos original $(x_1, ..., x_n)$.

2. Calcule, para cada una de las B muestras bootstrap, una réplica de $\widehat{\theta}$. Es decir, obtenga para cada muestra bootstrap, $\widehat{\theta}^* = s\left(x_1^*, ..., x_n^*\right)$. Por ejemplo, si $\widehat{\theta} = t\left(F_n\right)$ entonces $\widehat{\theta}^* = t\left(F_n^*\right)$, donde F_n^* es la distribución empírica asociada a $x_1^*, ..., x_n^*$.

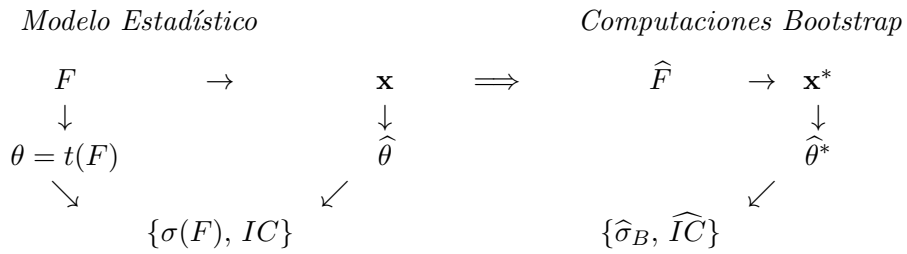
Una vez realizados los pasos 1 y 2 se obtendrán las denominadas réplicas o replicaciones bootstrap $\widehat{\theta}_1^*, ..., \widehat{\theta}_n^*$ a partir de las cuales se tiene una estimación de la distribución bootstrap $\widehat{G}$ que es a su vez una estimación de la distribución real del estimador $\widehat{\theta}$. Es decir, que en principio, se tienen dos niveles de error: el error estadístico y el error de simulación. Un análisis de estos tipos de error puede encontrarse en Davison y Hinkley (1997). Por ejemplo, la estimación bootstrap del desvío estándar para $\widehat{\theta}$ viene dada por la siguiente expresión.

$$\widehat{\sigma}_B = \sqrt{\frac{1}{(B)} \sum_{b=1}^{B} \left(\hat{\theta}_b^* - \hat{\theta}_{(\cdot)}^*\right)^2}, \tag{2.2}$$

donde $\hat{\theta}_{(\cdot)}^* = (1/B) \sum_{b=1}^{B} \hat{\theta}_b^*$. De la misma forma el sesgo de $\widehat{\theta}$ puede ser aproximado por:

$$\text{sesgo} = \hat{\theta}_{(\cdot)}^* - \hat{\theta}. \tag{2.3}$$

El siguiente esquema gráfico presenta una síntesis de la metodología bootstrap.



En la práctica, el éxito de la puesta en acción del análisis bootstrap puede estar ligado a los siguientes factores:

1. La elección del modelo $\widehat{F}$ que imita el hipotético modelo F. En situaciones simples como las presentadas en este capítulo esto puede no ser un inconveniente pero en casos de estructura más compleja como en los casos de regresión de los Capítulos 4 y 7, la elección pasa a tener un rol predominante.
2. Por construcción, la distribución de $\widehat{\theta}^*$ en el caso no paramétrico es discreta si bien la distribución de $\widehat{\theta}$ puede no serlo. El Ejemplo 2.3 de este capítulo presenta un procedimiento capaz de corregir este inconveniente.
3. La presencia de valores atípicos influye de forma directa en los resultados bootstrap. Un ejemplo de esto último puede encontrarse en Stine (1989).
4. La suavidad del estimador $\widehat{\theta}$ es de particular importancia en la construcción de intervalos de confianza. La cobertura y la precisión del método BC_a que se presentará en el Capítulo 3 dependen notablemente de este aspecto.

Como última observación se destaca que a lo largo de este trabajo se hablará repetidas veces del bootstrap para hablar del bootstrap no paramétrico ya que se corresponde con la forma de aplicación más general de la metodología.

2.1 Inferencia para la media y la mediana

En los próximos ejemplos se usará el conjunto de datos VO_2max que se corresponden con la máxima cantidad de oxígeno consumida en ml/kg/min por 24 atletas profesionales medidos en el CNARD (Centro Nacional de Alto Rendimiento Deportivo, Buenos Aires, Argentina) y que figuran en la Tabla 2.1. Este parámetro es considerado como el *golden standard* en cardiología. En un primer análisis simplista se estudia la media de los datos.

62.90	56.50	43.30	61.50	45.90	58.60	56.60	57.00
63.80	63.20	63.70	40.00	57.00	51.00	61.00	52.90
60.00	63.00	50.50	50.50	53.80	62.80	58.80	58.10

Tabla 2.1: Datos del CNARD

El ejemplo se analizará con R como se hará a lo largo de este trabajo con los demás ejemplos y para el caso en cuestión se usará la función `sample()` para generar las muestras bootstrap, es decir, las muestras obtenidas de forma aleatoria, con reemplazo, a partir de la muestra original.

```
#Boostrap con la funcion sample()
mvo2 <- c(62.9, 57, 56.5, 51, 43.3, 61, 61.5, 52.9,
 45.9, 60, 58.6, 63, 56.6, 50.5, 57, 50.5, 63.8, 53.8,
 63.2, 62.8, 63.7, 58.8, 40, 58.1)

 set.seed(123); m <- 5000; b.res.1 <- numeric(m)
for(i in 1:m)
    {
    b.res.1[i] <- mean(sample(mvo2, replace=T))
    }

#Sesgo bootstrap
mean(b.res.1 - mean(mvo2))
[1] -0.015

# Desvio estandar bootstrap de la media
sd(b.res.1)
[1] 1.33
```

El sesgo estimado es en este caso -0.015 que es muy chico para este problema médico y el desvío es de 1.33. Para entender la precisión del desvío bootstrap encontrado es posible comparar este resultado con un resultado aportado por la teoría clásica para la cual se sabe que

1. si las observaciones $x_i \sim N\left(\mu, \sigma^2\right)$, $\bar{x} \sim N\left(\mu, \sigma^2/n\right)$
2. si $E\left(x_1^2\right) < \infty$ y x_i son iid, $\sqrt{n}\left(\bar{x} - \mu\right) \rightarrow^D N\left(0, \sigma^2\right)$, siendo $\mu = E\left(x_1\right)$ y $\sigma^2 = Var\left(x_1\right)$, de donde $Var\left(\bar{x}\right) \simeq \sigma^2/n$ si n es grande.

Se tiene entonces que el desvío es aproximadamente:

```
sd(mvo2)/sqrt(24)
[1] 1.36
```

lo que destaca la buena precisión del desvío bootstrap.

El mismo análisis puede realizarse para la mediana. El proceso es completamente análogo gracias a la flexibilidad del bootstrap.

```
set.seed(123); m <- 5000; b.res.1 <- numeric(m)
for(i in 1:m)
    {
    b.res.1[i] <- median(sample(mvo2, replace=T))
    }

 #Sesgo bootstrap
mean(b.res.1 - median(mvo2))
[1] 0.13432

# Desvio estandar bootstrap de la mediana
sd(b.res.1)
[1] 1.491345
```

Para este caso se sabe que si los datos tienen distribución simétrica con única mediana en θ, es decir, si $x_i \sim F = G\left(. - \theta\right)$ con densidad $f\left(x\right) = g\left(x - \theta\right)$, entonces

$$\sqrt{n}\left(\widehat{\theta} - \theta\right) \rightarrow^{D} N\left(0, \left(\frac{1}{2f\left(\theta\right)}\right)^2 = \left(\frac{1}{2g\left(0\right)}\right)^2\right)$$

donde $\widehat{\theta}$ es la mediana muestral. Es decir, el desvío de $\widehat{\theta}$ es aproximadamente $1/(2g(0)\sqrt{n})$. En particular, si $G \sim N\left(0, \sigma^2\right)$, estimando a σ por el desvío estándar de las observaciones que se nota $\widehat{\sigma}$, se tiene que la mediana muestral tiene varianza asintótica que puede estimarse por $\left(\widehat{\sigma}\sqrt{2\pi}/2\sqrt{n}\right)^2$ con lo cual el desvío de $\widehat{\theta}$ es aproximadamente $\widehat{\sigma}\sqrt{2\pi}/2\sqrt{n}$.

```
n<-24
(sd(mvo2)*sqrt(2*pi))/(2*sqrt(n))
[1] 1.707027
```

Se ve nuevamente que la precisión del resultado es razonablemente buena respecto de la teoría asintótica.

2.2 El coeficiente de variación

En este ejemplo, se considera un parámetro menos trivial que en el ejemplo anterior: el coeficiente de variación $\theta = \sigma/\mu$, es decir el cociente entre el desvío y la media. El ejemplo subraya otra bondad del bootstrap: su adaptabilidad a contextos de difícil tratamiento teórico. Repitiendo exactamente el mismo proceso se obtiene la estimación bootstrap del desvío, lo que permite al menos tener una noción del comportamiento del estimador usado.

```
m <- 5000; b.res.1 <- numeric(m);
cv<-sd(mvo2)/mean(mvo2)
cv
[1] 0.118411
for(i in 1:m)
{
  muestra<-sample(mvo2, replace=T)
  b.res.1[i] <- sd(muestra)/
    mean(muestra)
}
sd(b.res.1)
[1] 0.01778369
```

La Figura 2.1 presenta la densidad estimada de las replicaciones bootstrap para el coeficiente de variación usando el comando densityplot de la librería *lattice* de R. Se recuerda que el estimador de densidad de una variable aleatoria X basado en una muestra $x_1, ..., x_n$ viene dado por

$$\widehat{f}\,(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x_i - x}{h}\right),$$

donde h es el parámetro de suavizado o ventana y k: $\mathbb{R} \to \mathbb{R}$ es una función no negativa tal que $\int k\,(u)\,du = 1$. El parámetro h regula el compromiso entre sesgo y varianza y su elección es más importante que la del núcleo k. En la Figura 2.1 se tomó un núcleo gaussiano y un parámetro de suavizado igual a 0.9 veces el valor mínimo entre el desvío estándar y la distancia intercuartil dividido por 1.34 veces el tamaño muestral a la potencia $-1/5$. Esta elección de h se conoce como la regla de oro de Silverman.

2.3 Bootstrap suavizado

Por construcción, la distribución de $\widehat{\theta}^*$ es discreta si la elección de $\widehat{F}$ es F_n, mientras que la distribución de $\widehat{\theta}$ podría no serlo. Este es un problema que el mismo bootstrap puede solucionar. En el ejemplo del consumo de oxígeno por 24 atletas profesionales que se viene estudiando, el histograma de la mediana de los bootstrap presenta claramente esta particularidad y puede apreciarse en la Figura 2.2. Para remediar este inconveniente puede usarse el *bootstrap suavizado* (ver Davison y Hinkley (1997)). Las réplicas bootstrap para el método bootstrap suavizado se definen como sigue. Si $\widetilde{x}_1, ..., \widetilde{x}_n$ es una muestra con reposición de $\{x_1, ..., x_n\}$ se define

$$x_i^* = \widetilde{x}_i + \epsilon_i,$$

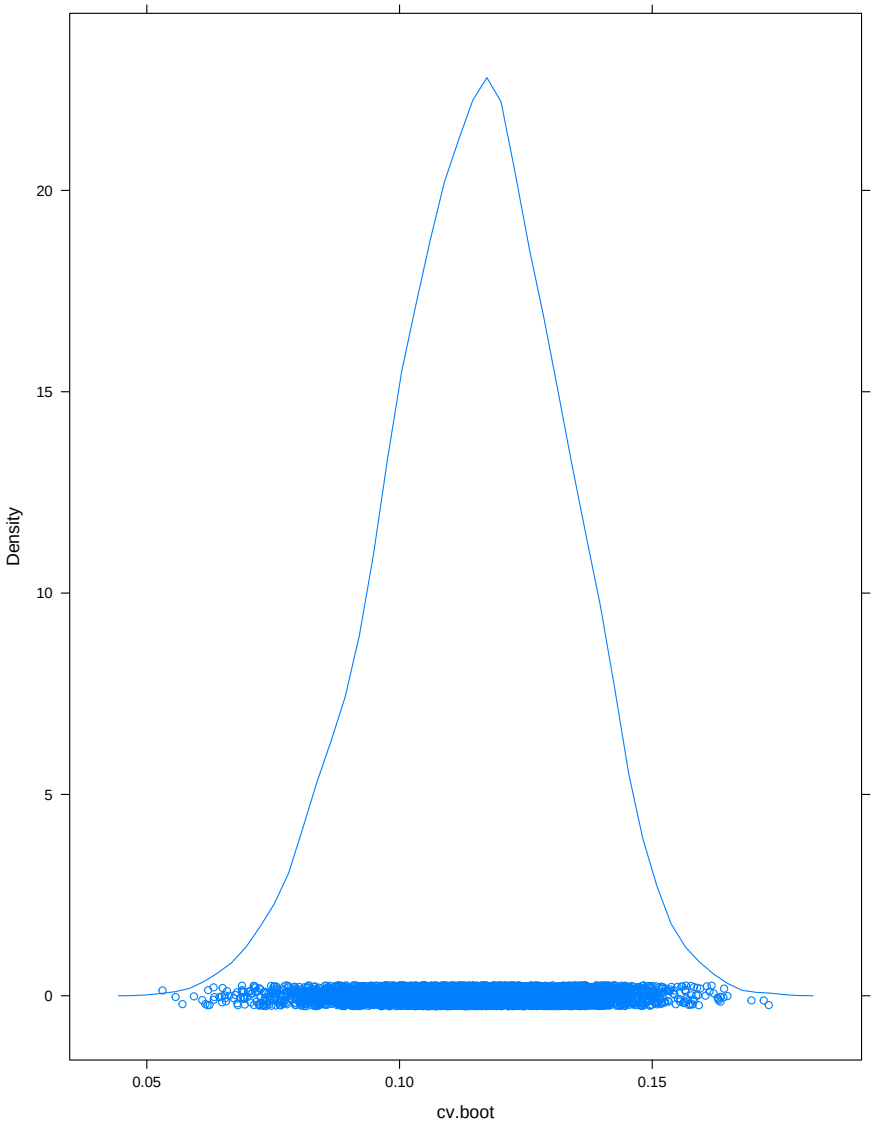


Figura 2.1: *Densidad Bootstrap del coeficiente de variación para los datos de atletas y B=500 replicaciones. Los círculos en la base de la función de densidad representan los datos utilizados. Se tomó un núcleo gaussiano y un parámetro de suavizado igual a 0.9 veces el valor mínimo entre el desvío estándar y la distancia intercuartil dividido por 1.34 veces el tamaño muestral a la potencia* $-1/5$.

donde $\epsilon_i \sim N\left(0, \tau^2\right)$. Es decir, en este bootstrap se generan las muestras bootstrap a partir de

$$\widehat{F} = F_n * \Phi\left(./\tau\right),$$

donde F_n es la distribución empírica, $*$ indica la convolución y Φ es la distribución normal estándar. El siguiente código utiliza este método aplicando a las replicaciones un suavizado normal con desvío 1/2:

```
b.res.2 <- numeric(m);
for(i in 1:m)
{
  b.res.2[i] <- median(sample(mvo2, replace=T)+rnorm(n=24)*.5)
}
hist(b.res.2,breaks=80,main="smoothed bootstrap",xlab="median.boot")
```

En la Figura 2.3 puede apreciarse el resultado de este método en la que se observa el histograma de las réplicas bootstrap para la mediana cuando se usa el bootstrap suavizado.

2.4 Secuencia aleatoria falsa ?

El siguiente ejemplo extraído de Verde (2011) ilustra el poder del bootstrap en un caso de aplicación de datos reales. Se toman dos secuencias binarias y se sabe que una de ellas fue producida por estudiantes del octavo año.

Secuencia 1:

00111000110010000100001000100010000000010010

01010110000111111001100010101100100100010000

00011111001

Secuencia 2:

01000101001100010100111010011000111101000111

01000110001101111000100101101101110001100100

010010000100

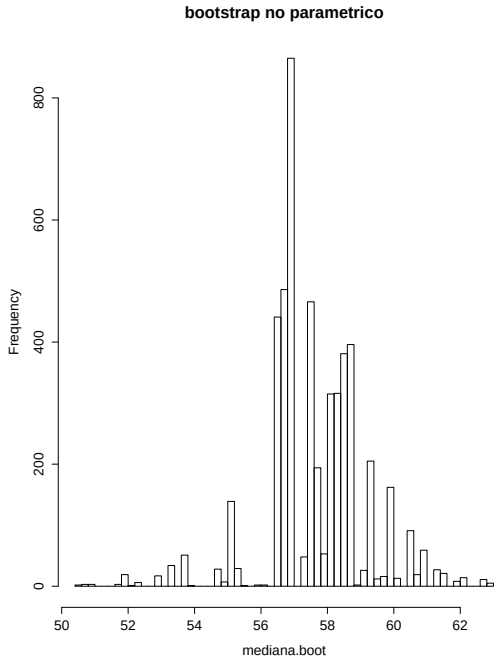


Figura 2.2: *Histograma para el bootstrap de la mediana usando B=5000 replicaciones*

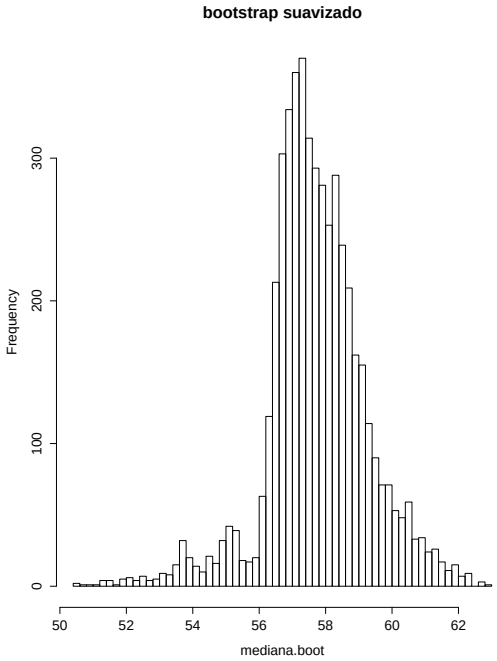


Figura 2.3: *Histograma para el bootstrap de la mediana usando el método bootstrap suavizado, con B=5000 replicaciones, un núcleo normal y un desvío igual a 1/2.*

La pregunta relevante es: Cuál de estas dos secuencias no fue producida de forma aleatoria? La pregunta no parece trivial a simple vista. De hecho, se tiene que la proporción de 1's en la primera secuencia es de 0.38 y de 0.45 en la segunda, es decir valores muy similares. Para intentar responder de forma concisa se consideran en este caso dos parámetros de interés: 1) el largo de la corrida más larga de números idénticos y 2) el número de cambios entre dígitos. Resumiendo la información para las dos secuencias se obtiene que para la secuencia 1 el número de cambios es 43, el largo de la corrida más larga es 8 y la proporción de 1's es 0.38 mientras que para la secuencia 2, el número de cambios es 52, el largo de la corrida más larga es 4 y la proporción de 1's es 0.45. Se puede pensar que el ejemplo propone testear la hipótesis siguiente: La secuencia ha sido generada a partir de lanzamientos independientes e idénticos de una moneda equilibrada. Es por eso que el algoritmo bootstrap adecuado para este problema puede describirse de la siguiente manera:

1. Genere una secuencia binaria de largo 100 a partir de una moneda equilibrada
2. Cuente el número de cambios

3. Cuente el largo de la corrida más larga
4. Repita 1) a 3) una gran cantidad de veces B=2000 y compare los resultados con las secuencias originales.

La Figura 2.4 realiza el diagrama de dispersión de los parámetros analizados y ubica las dos secuencias iniciales en rojo. El diagrama establece acertadamente el resultado final. Para las dos secuencias analizadas se tiene que:

```
# Secuencia 1

mean(sw<43)
 # proporcion de cambios menores al valor
 # de la secuencia 1
[1] 0.0785
mean(long<8)
 # proporcion de corridas menores al valor
 # de la secuencia 1
[1] 0.6785

# Secuencia 2

mean(sw<52)
[1] 0.6605

mean(long<4)
[1] 0.001
# Valor muy pequeno.
```

La proporción de corridas más largas menores al valor de la corrida más larga para la secuencia 2 es 0.001 que es exageradamente chico mientras que la misma medida para la secuencia 1 es 0.6785. Se concluye entonces que la secuencia 2 debe ser falsa.

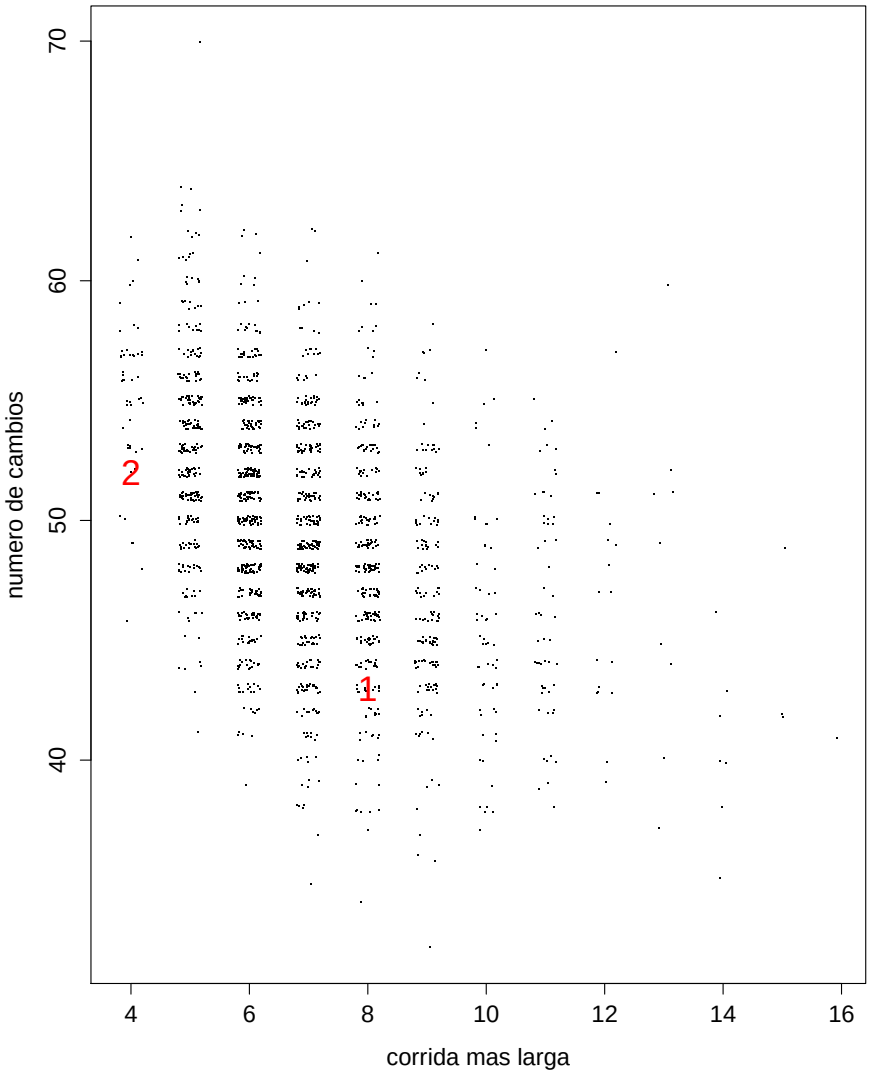


Figura 2.4: *Diagrama de dispersión de los datos bootstrap. Se han ubicado además las dos secuencias iniciales en rojo.*

Capítulo 3

Intervalos de confianza

Se han descripto hasta ahora métodos bootstrap para estimaciones puntuales, medidas de dispersión, sesgo y otras características poblacionales. No debe ser éste el único análisis pues las posibles conclusiones serían pobres o mismo erróneas. Los intervalos de confianza son una herramienta estadística fundamental en el análisis de precisión de un estimador y en este capítulo se desarrollan algunos de los métodos para la construcción de intervalos de confianza bootstrap.

3.1 Intervalos normales

3.1.1 Caso no paramétrico

Como primer enfoque al análisis de métodos bootstrap en la generación de intervalos de confianza se ha optado por analizar el método normal, seguramente el enfoque más simple. Éste supone que la distribución de las replicaciones bootstrap $\widehat{\theta}_1^*, ..., \widehat{\theta}_B^*$ es asintóticamente normal, es decir,

$$\widehat{\theta}_b^* \longrightarrow^D N(\widehat{\theta}, \tau^2),$$

donde $\tau > 0$ es una constante.

Definición 3.1. A partir de la teoría asintótica se define el intervalo de confianza bootstrap normal para θ de nivel $1 - 2\alpha$ de la siguiente manera:

$$IC_{norm} = \left[\widehat{\theta} - z^{(1-\alpha)}\widehat{\sigma}_B \, , \, \widehat{\theta} + z^{(1-\alpha)}\widehat{\sigma}_B\right], \tag{3.1}$$

donde $z^{(\alpha)}$ es el percentil α de una normal estándar, es decir, $\Phi\left(z^{(\alpha)}\right) = \alpha$, $z^{(\alpha)} = -z^{(1-\alpha)}$ y $\widehat{\sigma}_B$ es la estimación bootstrap del desvío estándar para $\widehat{\theta}$ definido en el Capítulo 2.

El uso de este intervalo carece de sentido cuando se tiene certeza de la no-normalidad de las réplicas bootstrap. Muchas veces es fácil verificar esto último a través de un qqplot, a través de un histograma o tan simplemente mediante un boxplot. Se verán ejemplos de uno y otro caso a lo largo de este trabajo. A modo de ilustración, y como ejemplo de uso del intervalo bootstrap normal, se consideran, en lo que sigue, los datos de los atletas de la Sección 2.1 del Capítulo 2:

```
boot.fun<-function(data,ind) mean(data[ind])
boot.media<-boot(mvo2,boot.fun,R=1000)
```

Se utiliza la función *boot* de la librería con el mismo nombre de R para generar mil réplicas bootstrap de la media de los 24 datos. La librería propone una forma alternativa para calcular las réplicas bootstrap sin hacer uso de la función sample. Las ventajas son considerables y se irán exlicando a lo largo del trabajo. La función boot exige como parámetro una función que explique el proceso de replicación para las muestras bootstrap generadas. La Figura 3.1 reproduce el histograma de los datos en el que se aprecia una apariencia claramente normal. En rojo se han representado además los límites inferior y superior del intervalo de confianza normal de nivel 95% y en azul los percentiles 2.5% y 97.5% de las réplicas bootstrap $\widehat{\theta}^*$. La similitud entre los límites y los percentiles es llamativa y no es casualidad. De hecho, si $\widehat{\theta}^* \sim N(\widehat{\theta}, \tau^2)$ entonces los límites superior e inferior del intervalo normal de nivel $1-2\alpha$ se corresponden con los percentiles α y $1-\alpha$ de de la función de distribución bootstrap, es decir, de la distribución de $\widehat{\theta}^*$ (ver Efron (1993)). Por otro lado, R permite el cálculo veloz de los intervalos de confianza bootstrap gracias a la misma librería boot. El comando *boot.ci* aplicado al comando boot usado para generar las réplicas devuelve los intervalos de confianza bootstrap clásicos al nivel pedido y se pueden observar a continuación. (Por defecto $\alpha = 0.05$). El intervalo básico, percentil y BC_a se explican con detalle más adelante.

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = boot.media, conf = 0.95)

Intervals :
Level      Normal              Basic
95%   (53.74, 59.01 )   (53.84, 59.09 )

Level     Percentile            BCa
95%   (53.61, 58.86 )   (53.43, 58.71 )
Calculations and Intervals on Original Scale
```

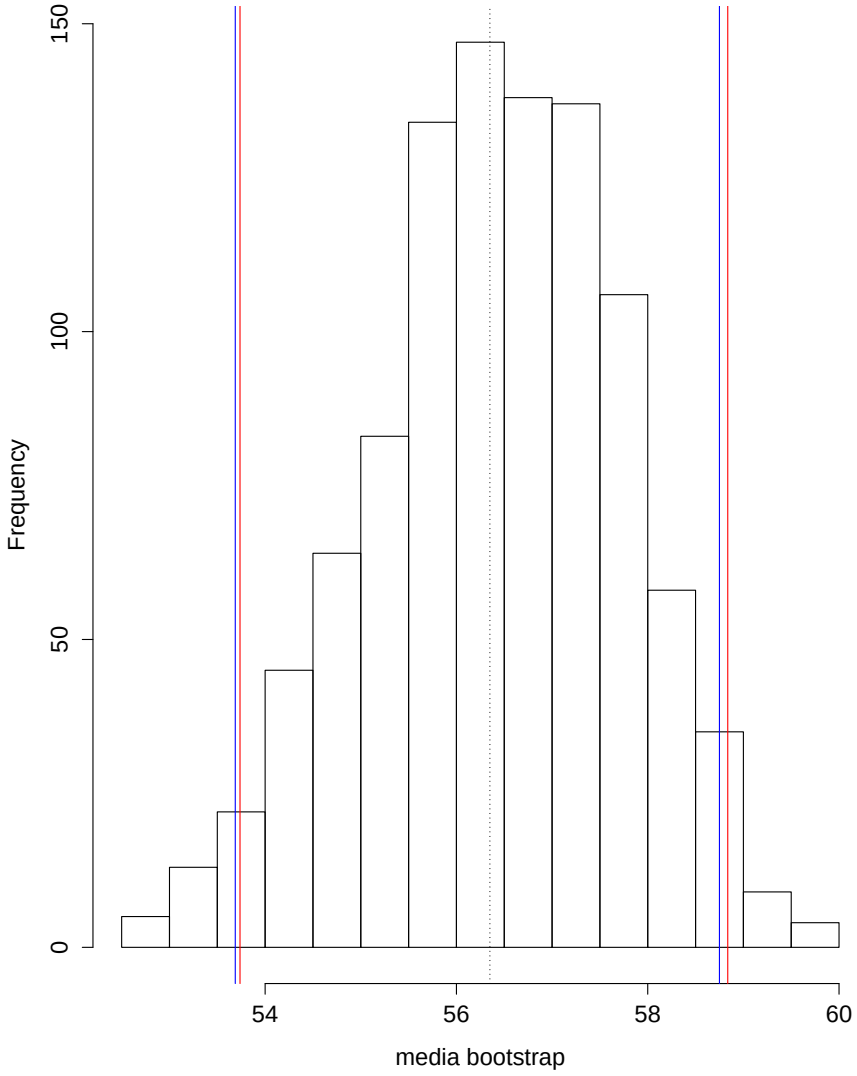


Figura 3.1: *Histograma de* $B = 1000$ *réplicas bootstrap de la media de los datos atletas.*

3.2 Intervalos por percentiles

3.2.1 Intervalo básico bootstrap

Los intervalos de confianza percentiles son una manera más natural de encarar el problema ya que respetan la distribución empírica de las replicaciones bootstrap, replicaciones que en conjunto son consideradas como una aproximación de la distribución real del estimador del parámetro estudiado.

Definición 3.2. Sea a_α el valor tal que $P(\widehat{\theta} - \theta \leq a_\alpha) = \alpha$. Análogamente se entiende que $a_{1-\alpha}$ es el valor tal que $P(\widehat{\theta} - \theta \geq a_{1-\alpha}) = \alpha$.

El conocimiento respecto de estos últimos valores supone el conocimiento de la distribución de $\widehat{\theta} - \theta$ que generalmente es desconocida. Aún así un intervalo de confianza teórico de nivel $1 - 2\alpha$ para θ viene dado por el intervalo

$$IC_t = \left[\widehat{\theta} - a_{1-\alpha}, \widehat{\theta} - a_\alpha\right]. \tag{3.2}$$

Una forma simple para aproximarlo consiste en tomar réplicas bootstrap de $\widehat{\theta}^* - \widehat{\theta}$ y estimar $a_{1-\alpha}$ y a_α por los cuantiles $1 - \alpha$ y α de las réplicas. Las estimaciones de los cuantiles se notan de la siguiente manera:

$$\widehat{a}_\alpha = \widehat{\theta}^*_{B\alpha} - \widehat{\theta},$$
$$\widehat{a}_{1-\alpha} = \widehat{\theta}^*_{B(1-\alpha)} - \widehat{\theta},$$

donde $\widehat{\theta}^*_{B\alpha}$ es la réplica $B\alpha$ en la lista ordenada de las réplicas bootstrap. Se define de forma análoga $\widehat{\theta}^*_{B(1-\alpha)}$. Definiendo,

$$IC_{basico} = \widehat{IC_t} = \left[\widehat{\theta} - \widehat{a}_{1-\alpha}, \widehat{\theta} - \widehat{a}_\alpha\right]$$

$$= \left[\widehat{\theta} - (\widehat{\theta}^*_{B(1-\alpha)} - \widehat{\theta}), \widehat{\theta} - (\widehat{\theta}^*_{B\alpha} - \widehat{\theta})\right]$$

$$= \left[2\widehat{\theta} - \widehat{\theta}^*_{B(1-\alpha)}, 2\widehat{\theta} - \widehat{\theta}^*_{B\alpha}\right],$$

se tiene la siguiente definición:

Definición 3.3. El *intervalo de confianza básico bootstrap* de nivel $1 - 2\alpha$ se define por:

$$IC_{basico} = \left[2\widehat{\theta} - \widehat{\theta}^*_{B(1-\alpha)}, 2\widehat{\theta} - \widehat{\theta}^*_{B\alpha}\right]. \tag{3.3}$$

Puede ser que en el cálculo de (3.3), $B\alpha$ no sea un valor entero. Para los valores de α más comunes, $B = 1000$ suele ser una buena tentativa. Aún así, en los casos en los que no se pueda conseguir un valor entero es conveniente hacer uso de la técnica de interpolación lineal en la escala cuantil normal. Dicha técnica propone una aproximación para $\widehat{\theta}^*_{B\alpha}$ dada por:

$$\widehat{\widehat{\theta}^*_{B\alpha}} = \widehat{\theta}^*_{Bk} + \frac{z_\alpha - z_{k/B}}{z_{(k+1)/B} - z_{k/B}}(\widehat{\theta}^*_{B(k+1)} - \widehat{\theta}^*_{Bk})$$

donde k es la parte entera de $B\alpha$. El método falla evidentemente con valores de $k = 0, B-1, B$.

3.2.2 El intervalo percentil

A diferencia del método normal, el intervalo denominado percentil, no presupone normalidad en la distribución de los $\widehat{\theta}^*$ y trata de inferir directamente usando los valores de los percentiles. Aún así, en un aspecto que se explicará más adelante, la suposición de normalidad forma parte del método. Si $\widehat{G}$ es la función de distribución bootstrap definida en el Capítulo 2, es decir, la función de distribución acumulada de $\widehat{\theta}^*$ entonces,

Definición 3.4. Se define el intervalo de confianza bootstrap percentil de nivel $1-2\alpha$ por:

$$IC_{perc} = \left[\widehat{G}^{-1}(\alpha), \widehat{G}^{-1}(1-\alpha)\right],$$

lo que, por definición de $\widehat{G}$, puede escribirse como:

$$IC_{perc} = \left[\widehat{\theta}^{*(\alpha)}, \widehat{\theta}^{*(1-\alpha)}\right], \tag{3.4}$$

donde $\widehat{\theta}^{*(\alpha)}$ es el percentil 100α de la distribución bootstrap.

Este intervalo, se denomina *intervalo percentil ideal bootstrap*. Como en la práctica sólo se puede afrontar un número finito de replicaciones, no se podrá obtener exactamente este último intervalo pero se podrá aproximarlo correctamente. Es importante repetir el algoritmo bootstrap una gran cantidad B de veces para obtener mejor precisión, donde B dependerá del contexto. Para este caso se requieren cerca de 1000 replicaciones generalmente a diferencia de lo que ocurre con la estimación del desvío que no exige más de 200 (ver Davison y Hinkley (1997)) pues se necesitan aproximar percentiles bajos y altos de la distribución. Es natural entonces considerar la siguiente aproximación del intervalo percentil ideal bootstrap dada por,

$$IC_{perc} \simeq [\widehat{\theta}^*_{B\alpha}, \widehat{\theta}^*_{B(1-\alpha)}]$$

donde una vez más $\widehat{\theta}^*_{B\alpha}$ es el $B\alpha-$ésimo valor en la lista ordenada de las B réplicas bootstrap.

Es importante notar que este método, como casi todos los métodos bootstrap se aplican tanto en casos paramétricos como en no paramétricos. Si la distribución bootstrap es aproximadamente normal ambos intervalos de confianza no deberán ser tan distintos. Antes de pasar a una mejora de este método en la construcción de intervalos de confianza se verá brevemente un ejemplo simulado en donde se evidencia la ventaja del método percentil frente al normal cuando la distribución de las replicaciones no es tal. A continuación del ejemplo, se detallan las propiedades y los supuestos del intervalo de confianza percentil bootstrap.

Ejemplo: un caso teórico simple

Sea $X_1 = x_1, ..., X_{10} = x_{10}$ observaciones obtenidas a partir de variables aleatorias normales estándar. Se supone que el parámetro de interés es $\theta = e^{\mu}$ donde μ es la media poblacional. Se define entonces $\widehat{\theta} = e^{\bar{x}}$. El valor real de θ, que aquí puede conocerse de forma exacta es $e^0 = 1$. Se obtuvo por otro lado $\widehat{\theta} \simeq 0.84$. El código utilizado en R y los resultados se detallan a continuación:

```
x<-rnorm(10)
x
  -0.03884065 -0.23215923  0.70586967 -0.46621674
  -0.76745185  0.57826208  0.61400508 -2.36820628
   0.57040009 -0.31536755
exp(mean(x))
 0.842004
```

La Figura 3.2 muestra el histograma de los $\widehat{\theta}_b^*$ a partir de $B = 1000$ replicaciones. Por otro lado, un intervalo de confianza bootstrap normal de nivel 95% para θ viene dado por

$$IC_{norm}[0.05] = [0.37, 1.27],$$

mientras que el intervalo de confianza percentil bootstrap del mismo nivel resulta ser

$$IC_{perc}[0.05] = [0.46, 1.36].$$

La diferencia resulta evidente por la forma en la que se han simulado los datos y se puede apreciar fácilmente con los histogramas. De hecho, sólo se cuentan 23 replicaciones bootstrap menores a 0.46. La Figura 3.3 presenta el histograma de $\widehat{\phi}^* = \log(\widehat{\theta}^*)$ la correción necesaria para tener normalidad en las réplicas. Una vez realizada la transformación, el

intervalo de confianza normal para ϕ resulta ser, $IC_{NORM}[0.05] = [-0.74, 0.32]$. Si se toma la función exponencial y se la aplica al intervalo de confianza bootstrap normal para ϕ se obtiene $[0.47, 1.37]$ muy cercano al intervalo percentil para θ.

Propiedades y supuestos del intervalo percentil bootstrap

Esto último se explica por una de las buenas propiedades del intervalo percentil.

Proposición 3.1. *El intervalo percentil bootstrap es invariante por transformaciones, es decir que, si ψ es una función monóntona y $\psi(\theta) = g$ entonces un intervalo de confianza para g viene dado por,*

$$[g_{PERC}[\alpha], g_{PERC}[1-\alpha]] = [\psi(\theta_{PERC}[\alpha]), \psi(\theta_{PERC}[1-\alpha])].$$

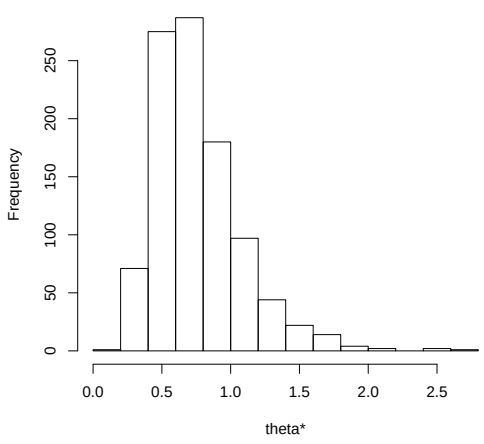


Figura 3.2: *Histograma de 1000 replicaciones bootstrap de $\widehat{\theta}$*

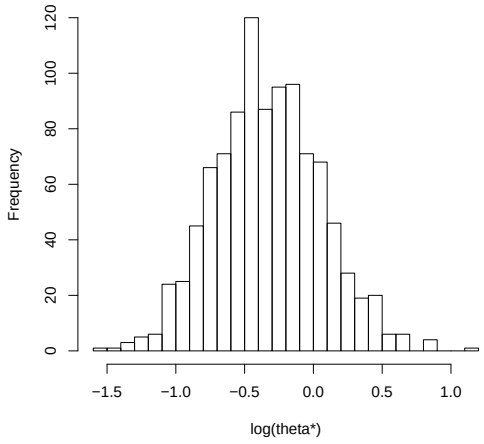


Figura 3.3: *Histograma de 1000 replicaciones bootstrap del estimador transformado* $\log(\widehat{\theta})$

Si se conociese en cada caso una transformación normalizadora para cada parámetro de interés, el uso de intervalos normales sería razonable aplicado a la escala correcta. Dado que esto parece impractible es interesante notar que el método usado por el intervalo percentil parece conocer de antemano dicha transformación como se sigue del siguiente resultado que puede hallarse en Efron (1993):

Lema 3.1. *Si existe h: $\mathbb{R} \to \mathbb{R}$ tal que $\widehat{\psi} = h(\widehat{\theta})$ es aproximadamente $N(\psi, c^2)$, donde*

$c > 0$ es constante, entonces el intervalo percentil bootstrap para θ es igual a $\left[h^{-1}(\widehat{\psi} - z^{(1-\alpha)}c), h^{-1}(\widehat{\psi} + z^{(1-\alpha)}c)\right]$, donde $z^{(\alpha)}$ es el valor tal que $\Phi\left(z^{(\alpha)}\right) = \alpha$.

Proposición 3.2. *El intervalo de confianza dado por (3.4) da el nivel adecuado si existe h tal que $h(\widehat{\theta})$ es simétrica.*

Demostración. La idea es suponer que existe una transformación monótona creciente $h(.)$ de forma tal que $h(\widehat{\theta}) = \widehat{\psi}$ tenga distribución simétrica. De existir, entonces $a_\alpha = -a_{1-\alpha}$, donde en este caso a_α se define por $P(\widehat{\psi} - \psi \leq a_\alpha) = \alpha$. Gracias a esto, pueden escribirse de forma distinta los límites en (3.3) para el parámetro ψ como,

$$\left[\widehat{\psi}^*_{B\alpha}, \widehat{\psi}^*_{B(1-\alpha)}\right]. \tag{3.5}$$

Antitransformando, se recupera el intervalo para θ, es decir:

$$\left[\widehat{\theta}^*_{B\alpha}, \widehat{\theta}^*_{B(1-\alpha)}\right]. \tag{3.6}$$

Esto implica que si el intervalo percentil es correcto para alguna escala transformada $\psi = h(\theta)$, entonces también lo será en la escala original θ. □

El analista no necesita conocer la transformación, sólo debe saber que existe. En este caso, *correcto* quiere decir que para todo θ, existe una transformación monótona $h(\theta) = \psi$, $h(\widehat{\theta}) = \widehat{\psi}$ de forma tal que $\widehat{\psi} \sim N(\psi, \tau^2)$ con τ una constante fija. De hecho, se puede ver que, si existe tal transformación, el intervalo percentil otorga límites exactos (ver Efron y Tibshirani (1986)). El primer ejemplo de esta sección exhibe un caso en donde no existe una transformación monótona que cumpla con este supuesto ya que, si bien existe una transformación que simetrize la distribución del estimador, la varianza de esta última no es constante respecto del parámetro.

Proposición 3.3. *El intervalo percentil bootstrap, además, preserva el rango, es decir, los límites del intervalo son valores en el rango de valores posibles del parámetro de interés.*

Es posible que el parámetro de interés tome sólo un cierto rango de valores posibles, como es por ejemplo el caso del coeficiente de correlación que toma valores en el intervalo [-1,1]. A diferencia del método normal, ya sea porque la estimación plug-in de θ respeta las mismas restricciones en los valores que θ o porque los límites del intervalo son valores del estadístico bootstrap $\widehat{\theta}^*$, el intervalo percentil tiene la propiedad mencionada. Esto permite, generalmente, obtener mayor precisión en los resultados.

El método es sensible a outliers y no corrige sesgos en caso de haberlos. Por otro lado, el intervalo lleva consigo el defecto del método bootstrap básico: esto es que la forma de la distribución de $\widehat{\theta}$ cambia con la estimación de F por $\widehat{F}$. Se introduce a continuación una nueva forma de construir intervalos que permite corregir algunos de los defectos mencionados.

3.2.3 Intervalos BC_a

BC_a es una abreviación de *Bias Corrected and Accelerated* (corrección del sesgo y aceleración). Este intervalo de confianza bootstrap fue propuesto por Efron (1987) con el fin de contrarrestar las dificultades del intervalo percentil respecto de sus supuestos. Está pensado para estadísticos $\widehat{\theta}$ sesgados y/o con desvíos τ^2 dependientes del valor de $\widehat{\theta}$. El método, al igual que para el caso anterior, utiliza percentiles de los bootstrap como puntos límite del intervalo de confianza aunque no tienen por qué ser los mismos. De hecho, el algoritmo de construcción considera dos parámetros nuevos a y z_0. Cada parámetro cumple un rol específico.

Definición 3.5. EL parámetro z_0 cumple la función de corrector del sesgo y se define como,

$$z_0 = \Phi^{-1}\left\{\widehat{G}(t)\right\},$$

lo que en términos de simulaciones puede estimarse por:

$$z_0 = \Phi^{-1}\left(\frac{\#\{\widehat{\theta}_b^* < \widehat{\theta}\}}{B}\right),$$

donde $\Phi(.)$ es la función de distribución acumulada de una normal estándar.

Es decir que $\widehat{z_0}$ se obtiene a partir de la proporción de observaciones bootstrap que están por debajo de $\widehat{\theta}$. De alguna manera, podría pensarse que $\widehat{z_0}$ es una medida del sesgo de $\widehat{\theta}^*$ respecto de $\widehat{\theta}$. Esto permite simetrizar la distribución de $\widehat{\theta}^*$ que es uno de los supuestos del intervalo percentil. En caso de no existir sesgo, el parámetro toma el valor 0.

El otro parámetro, a, denominado *constante de aceleración* refiere a la tasa de cambio del desvío estándar de $\widehat{\theta}$ respecto al verdadero valor θ (ver Di Ciccio y Efron (1996)). El intervalo percentil supone que existe una transformación que normaliza la distribución del estimador de forma tal que la varianza del estimador transformado es constante respecto del parámetro. El parámetro a corrige la variabilidad de la varianza del estimador respecto de θ en caso de que ésta exista. En otro caso, toma simplemente el valor 0.

Definición 3.6. Se define el parámetro a del intervalo de confianza bootstrap BC_a en el caso paramétrico por:

$$a = \frac{1}{6} \frac{E\left\{l'(\theta)^3\right\}}{Var\left\{l'(\theta)\right\}^{3/2}},$$

donde $l(\theta)$ es el logaritmo de la función de verosimilitud de $\widehat{\theta}$ y $l'(\theta)$ es la derivada de dicha función.

Esta expresión puede derivarse de la explicación heurística que se da en el enfoque paramétrico para explicar el método BC_a explicado a continuación. En la práctica, y en el caso no paramétrico, el parámetro a se estima por:

$$\widehat{a} = \frac{\frac{1}{n}\sum_{i=1}^{n}(\bar{\widehat{\theta}} - \widehat{\theta}_{(i)})^3}{6[\frac{1}{n}\sum_{i=1}^{n}(\bar{\widehat{\theta}} - \widehat{\theta}_{(i)})^2]^{3/2}},$$

donde $\widehat{\theta}_{(i)}$ se calcula como $\widehat{\theta}$ a partir de la muestra sin la observación i-ésima. Estos parámetros corrigen el intervalo percentil definido previamente y dan lugar al intervalo de confianza BC_a.

Definición 3.7. El intervalo de confianza bootstrap BC_a de nivel $1 - 2\alpha$ se define como

$$IC_{bc_a} = \left[\widehat{G}^{-1}(\alpha_1), \widehat{G}^{-1}(\alpha_2)\right],$$

donde

$$\alpha_1 = \Phi\left(z_0 + \frac{z_0 + z^{(\alpha)}}{1 - a(z_0 + z^{(\alpha)})}\right),$$

$$\alpha_2 = \Phi\left(z_0 + \frac{z_0 + z^{(1-\alpha)}}{1 - a(z_0 + z^{(1-\alpha)})}\right)$$

Una vez más, en la práctica, este intervalo se estima por:

$$IC_{bc_a} \simeq \left[\widehat{\theta}^*_{B\widehat{\alpha}_1}, \widehat{\theta}^*_{B\widehat{\alpha}_2}\right],$$

donde

$$\widehat{\alpha}_1 = \Phi\left(\widehat{z}_0 + \frac{\widehat{z}_0 + z^{(\alpha)}}{1 - \widehat{a}(\widehat{z}_0 + z^{(\alpha)})}\right),$$

y

$$\widehat{\alpha}_2 = \Phi\left(\widehat{z}_0 + \frac{\widehat{z}_0 + z^{(1-\alpha)}}{1 - \widehat{a}(\widehat{z}_0 + z^{(1-\alpha)})}\right).$$

Estos intervalos tienen las mismas propiedades mencionadas para los intervalos percentiles además de las correcciones por sesgo y variabilidad del desvío.

Un enfoque paramétrico para explicar el método BC_a

Se puede explicar este método en términos más formales y en términos paramétricos, de donde surgen los límites del intervalo (ver Davison y Hinkley (1997)). Se supone para ello que el conjunto de datos original proviene de un modelo F con un único parámetro desconocido θ estimado por $\widehat{\theta}$. Se supone además que existe una transformación $h(.)$, monótona creciente, en principio desconocida, un factor de corrección del sesgo z_0 y otro factor de corrección a de la variabilidad del desvío de manera que si $\widehat{\phi} = h(\widehat{\theta})$,

$$\widehat{\phi} \sim N(\phi - z_0(1 + a\phi), (1 + a\phi)^2).$$

Para calcular los límites del intervalo BC_a se calculan primero los límites para el intervalo de ϕ y se transforman los resultados de modo a retornar a la escala de θ usando la distribución bootstrap de $\widehat{\theta}$. Al igual que para el intervalo percentil, si la suposición sobre la transformación normalizante es correcta su puede ver que los límites ofrecidos por BC_a son exactos (ver Di Ciccio y Efron (1986)). Respecto del parámetro a, se puede entender el mismo como una medidad sobre la velocidad con la que el desvío estándar está cambiando en la escala normalizada. Se considera:

$$\widehat{\phi} = \phi + (1 + a\phi)(Z - z_0), \tag{3.7}$$

con Z una variable con distribución $N(0, 1)$ y z_α el cuantil α. Se sigue que

$$\log(1 + a\widehat{\phi}) = \log(1 + a\phi) + \log(1 + a(Z - z_0)).$$

Resta calcular $\widehat{\phi}_\alpha$ de modo que $P(\phi \leq \widehat{\phi}_\alpha) = \alpha$.

$$P\left(\log(1 + a\phi) \leq \log(1 + a\widehat{\phi}_\alpha)\right) = \alpha,$$

entonces, por (3.7)

$$P\left(\log\left(\frac{1 + a\widehat{\phi}}{1 + a(Z - z_0)}\right) \leq \log\left(1 + a\widehat{\phi}_\alpha\right)\right) = \alpha.$$

Tomando la función exponencial, se obtiene que

$$P\left(\frac{1 + a\widehat{\phi}}{1 + a(Z - z_0)} \leq 1 + a\widehat{\phi}_\alpha\right) = \alpha,$$

luego,

$$P\left(\frac{1+a(Z-z_0)}{1+a\widehat{\phi}} \geq \frac{1}{1+a\widehat{\phi}_\alpha}\right) = \alpha,$$

de donde,

$$P\left(Z \geq \left[\frac{1+a\widehat{\phi}}{1+a\widehat{\phi}_\alpha} - 1\right]\frac{1}{a} + z_0\right) = \alpha,$$

es decir,

$$P\left(Z \leq \left[\frac{1+a\widehat{\phi}}{1+a\widehat{\phi}_\alpha} - 1\right]\frac{1}{a} + z_0\right) = 1-\alpha,$$

de donde,

$$\frac{1+a\widehat{\phi}}{1+a\widehat{\phi}_\alpha} - 1 = (z^{(1-\alpha)} - z_0)a,$$

osea,

$$\frac{1+a\widehat{\phi}}{1+a\widehat{\phi}_\alpha} = 1 + (z^{(1-\alpha)} - z_0)a,$$

$$\frac{1+a\widehat{\phi}_\alpha}{1+a\widehat{\phi}} = \frac{1}{1+(z^{(1-\alpha)} - z_0)a},$$

entonces,

$$1+a\widehat{\phi}_\alpha = \frac{1+a\widehat{\phi}}{1+(z^{(1-\alpha)} - z_0)a},$$

por lo que,

$$\widehat{\phi}_\alpha = \left[\frac{1+a\widehat{\phi}}{1+a(z^{(1-\alpha)} - z_0)} - 1\right]\frac{1}{a} = \frac{\left[a\widehat{\phi} - a(z^{(1-\alpha)} - z_0)\right]}{1+a(z^{(1-\alpha)} - z_0)}\frac{1}{a}$$

$$= \frac{\left(\widehat{\phi} - z^{(1-\alpha)} + z_0\right)}{\left(1+a(z^{(1-\alpha)} - z_0)\right)}.$$

Como $z^{(1-\alpha)} = -z^{(\alpha)}$, se obtiene

$$\widehat{\phi}_\alpha = \frac{\widehat{\phi} + z^{(1-\alpha)} + z_0}{1-a(z^{(\alpha)} + z_0)} = \widehat{\phi} + \frac{(1+a\widehat{\phi}(z^{(\alpha)} + z_0))}{1-a(z^{(\alpha)} + z_0)}.$$

Ahora bien, $\widehat{\theta}_\alpha = h^{-1}(\widehat{\phi}_\alpha)$. El problema es que h es desconocida. Se considera entonces $\widehat{G}$, es decir, la distribución de $\widehat{\theta}^*$. Se tiene que,

$$\widehat{G}(\widehat{\theta}_\alpha) = P(\widehat{\theta}^* \leq \widehat{\theta}_\alpha) = P(\widehat{\phi}^* \leq \widehat{\phi}_\alpha) = \Phi\left(\frac{\widehat{\phi}_\alpha - \widehat{\phi}}{1+a\widehat{\phi}} + z_0\right)$$

$$= \Phi\left(z_0 + \frac{z_0 + z^{(\alpha)}}{1 - a(z_0 + z^{(\alpha)})}\right),$$

de donde

$$\widehat{\theta}_\alpha = \widehat{G}^{-1}\left[\Phi\left(z_0 + \frac{z_0 + z^{(\alpha)}}{1 - a(z_0 + z^{(\alpha)})}\right)\right].$$

Se obtiene entonces que, $\widehat{\theta}_\alpha = \widehat{\theta}^*_{B\beta}$, con

$$\beta = \Phi\left(z_0 + \frac{z_0 + z^{(\alpha)}}{1 - a(z_0 + z^{(\alpha)})}\right),$$

y $\widehat{\theta}^*_{B\beta}$ el $B\beta-$ésimo valor en la lista ordenada de las B réplicas bootstrap. De la misma forma, usando la distribución bootstrap, se pueden deducir las expresiones para z_0 y a.

$$P^*(\widehat{\theta}^* \leq \widehat{\theta}) = P^*(\widehat{\phi}^* \leq \widehat{\phi}) = P(N(\widehat{\phi} - z_0(1 + a\widehat{\phi}), (1 + a\widehat{\phi})^2) \leq \widehat{\phi}) = \Phi(z_0).$$

Esto implica que $z_0 = \Phi\{\widehat{G}(\widehat{\theta})\}$. Como simple observación, aunque no se hará aquí, se puede deducir para el caso paramétrico la expresión exacta para a usando el logaritmo de la verosimilitud.

3.3 Ejemplo: el estimador de la varianza

Este ejemplo realiza un análisis práctico del método BC_a, un análisis teórico del nivel de cobertura del método en comparación con los otros métodos bootstrap estudiados en un caso paramétrico, y el análisis del mismo nivel de cobertura en un caso de remuestreo no paramétrico.

Un análisis práctico

Los datos para el análisis práctico han sido extraídos de la librería *bootstrap* de R con el comando *spatial*. Veintiséis niños con daños neurológicos acudieron a dos pruebas A y B de percepción espacial. Los datos se representan por los pares $z_i = (A_i, B_i)$ y en este caso, el parámetro llevado a análisis es $\theta = Var(A)$. Se sabe que el estimador plug-in de θ es $\widehat{\theta} = \sum_{i=1}^{n}(A_i - \bar{A})^2/n$ que además es sesgado. Generando replicaciones bootstrap de este

estimador, se ha construido el histograma de la Figura 3.4 y los intervalos de confianza de nivel 90%.

```
boot_var<-function(data, ind) (1/n)*
                sum((data[ind]-mean(data[ind]))^2)
boot_1<-boot(A,boot_var,R=1000)
```

Los datos se han generado una vez más con el comando boot, lo que permite calcular fácilmente los intervalos de confianza pertinentes. El proceso de simulación involucra aquí el cálculo de la estimación de la varianza para cada muestra bootstrap.

```
hist(boot_1$t,main="",xlab="bootstrap")
boot.ci(boot_1,conf=0.9)
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = boot_1, conf = 0.9)

Intervals :
Level      Normal              Basic
90%   (110.9, 244.7 )   (109.1, 243.0 )

Level     Percentile            BCa
90%   (100.0, 234.0 )   (116.4, 262.0 )
Calculations and Intervals on Original Scale
```

Es clara la diferencia entre el intervalo BC_a y los otros. Aún así, no se está en condiciones de asegurar que uno es *mejor* que otro. Una forma de medir esta bondad es estudiando niveles de cobertura a través de simulaciones y es lo que se hará en tercera instancia. Antes de eso, se ha optado por estudiar el nivel de cobertura exacto de los métodos cuando los datos se obtienen a partir de distribuciones normales.

Un análisis teórico y paramétrico del estimador plug-in de la varianza

Concretamente, se supone ahora que se tienen k muestras independientes $(x_{i1}, ..., x_{im})$, $i = 1, ..., k$, generadas a partir de distribuciones normales con distinta media λ_i pero con

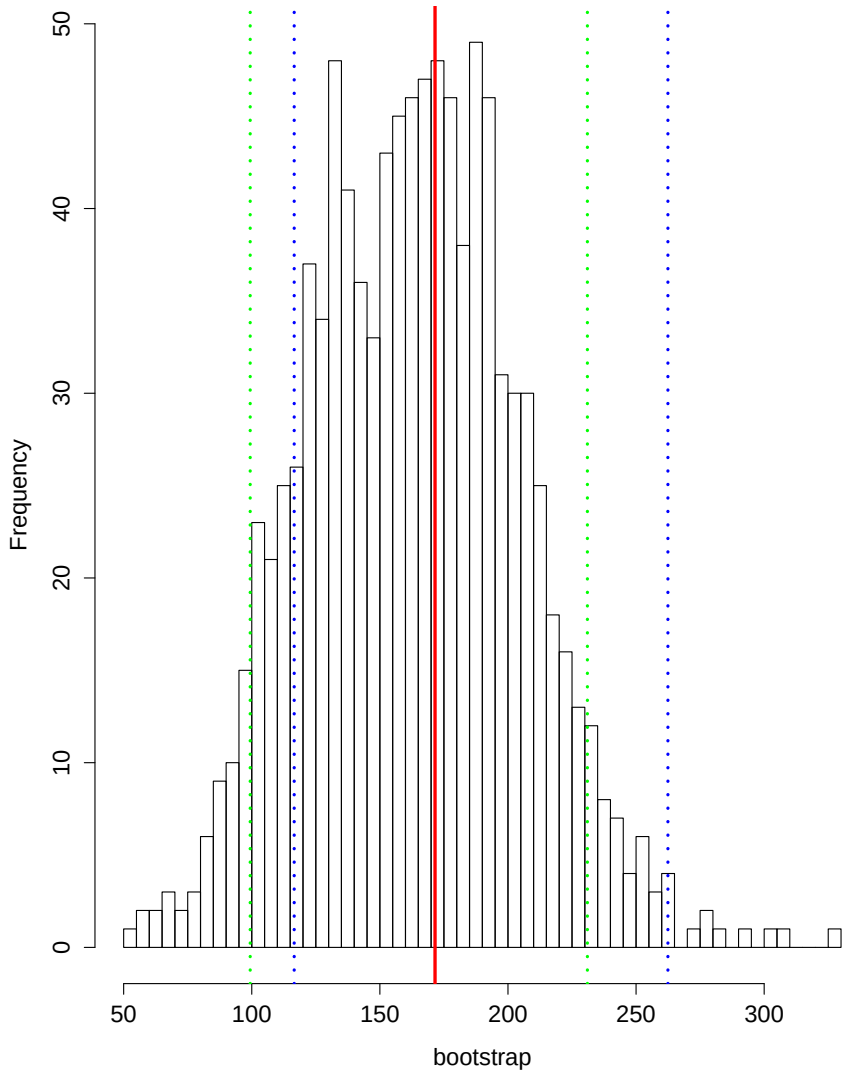


Figura 3.4: *Histograma de las replicaciones bootstrap de* $\widehat{\theta}$ *para el caso no paramétrico. La línea roja representa el valor de* $\widehat{\theta}$ *observado. Se han superpuesto los límites de los intervalos de confianza percentil (verde) y bca (azul).*

Nominal	Básico	Percentil	BC_a
1.0	0.8	0.0	1.0
2.5	2.5	0.0	2.5
5.0	4.8	0.0	5.0
95.0	35.0	1.6	91.5
97.5	36.7	4.4	100.0
99.0	38.3	6.9	100.0

Tabla 3.1: *Niveles de cobertura (%) exactos de intervalos de confianza superiores para la varianza de una normal estimada por máxima verosimilitud utilizando 10 muestras, cada una de tamaño 2.*

igual varianza θ. Se está, una vez más, interesado en estimar el parámetro θ y, para ello, se considera el estimador, que se sabe sesgado, $\widehat{\theta} = n^{-1} \sum_{i=1}^{k} \sum_{j=1}^{m} (x_{ij} - \bar{x}_i)^2$, donde $n = mk$ y $\bar{x}_i$ es el promedio de $x_{i1}, ..., x_{im}$. Se sabe además que $\widehat{\theta}$ tiene distribución $n^{-1}\theta\chi^2_d$, donde $d = k(m-1)$. Este resultado exacto permite obtener niveles de cobertura exactos para los distintos métodos bootstrap. Si el cuantil α de la distribución χ^2_d se denota por $c_{d,\alpha}$, utilizando que $\widehat{\theta}^*$ es $n^{-1}\widehat{\theta}\chi^2_d$, es fácil ver que los límites superiores del intervalo de confianza de nivel α para el método básico bootstrap y para el método percentil son, respectivamente $2\widehat{\theta} - n^{-1}\widehat{\theta}c_{d,1-\alpha}$, y $n^{-1}\widehat{\theta}c_{d,\alpha}$. Los niveles de cobertura se calculan de forma exacta y están dados por

$$P\left(\theta \leq 2\widehat{\theta} - n^{-1}\widehat{\theta}c_{d,1-\alpha}\right) = P\left(\chi^2_d \geq \frac{n}{2 - n^{-1}c_{d,1-\alpha}}\right),$$

y

$$P\left(\chi^2_d \geq \frac{n^2}{c_{d,\alpha}}\right).$$

Para el método BC_a, Davison y Hinkley (1997) muestran que $z_0 = \Phi^{-1}\left\{P\left(\chi^2_d \leq n\right)\right\}$, $a = (1/3)2^{1/2}n^{-1/2}$ y que el límite superior del intervalo de confianza BC_a de nivel α es $n^{-1}\widehat{\theta}c_{d,\widetilde{\alpha}}$, donde $\widetilde{\alpha} = \Phi\left(z_0 + \widehat{z}_\alpha/(1 - a\widehat{z}_\alpha)\right)$, y $\widehat{z}_\alpha = z_0 + z_\alpha$. Se tiene entonces que el nivel de cobertura para ese límite es

$$p\left(\chi^2_d \geq \frac{n^2}{c_{d,\widetilde{\alpha}}}\right).$$

En el caso de tomar, $k = 10$ y $m = 2$ se tienen los resultados de la Tabla 3.1. Los resultados son definitivamente malos para los métodos básico y percentil y acentúan el peligro de usar dichos métodos sin el cuidado de los supuestos.

n	Percentil	BC_a
20	0.76	0.82
35	0.83	0.87
100	0.88	0.89

Tabla 3.2: *Niveles de cobertura obtenidos para intervalos de confianza nominales de nivel 90% para la varianza de una normal estándar a partir de 2000 simulaciones con diferentes tamaños muestrales y* $B = 1000$ *replicaciones bootstrap no paramétricas.*

Simulaciones no paramétricas

Para esta última parte, se han querido comparar los niveles de cobertura de los intervalos de confianza Bc_a y percentil en un estudio de simulación. Se han considerado, respectivamente, 2000 muestras independientes de tamaño 20, 35 y 100 de normales estándar y se han calculado los intervalos de confianza bootstrap mencionados de nivel 90% para la varianza, utilizando la estimación plug-in, mediante $B = 1000$ replicaciones no paramétricas a partir de cada muestra. Los niveles de cobertura obtenidos pueden observarse en la Tabla 3.2. En este caso, ninguno de los métodos es realmente satisfactorio con tamaño muestral $n = 20$, si bien en todo los casos, como era de esperarse, el método BC_a tiene mejores niveles de cobertura.

3.4 Valores de influencia y jacknife-after-bootstrap

En la construcción de intervalos de confianza bootstrap, reviste especial interés el estudio de la distribución de $\widehat{\theta}^* - \widehat{\theta}$. En esta sección se propone una técnica de análisis de sensibilidad denominada $jacknife\text{-}after\text{-}bootstrap$ (ver Efron 1993). Se busca, a partir de dicha técnica, estudiar la influencia que pocos datos tienen en la distribución global del estadístico previamente mencionado.

Se requiere para ello cierta formalidad que se detalla a continuación.

3.4.1 Valores de influencia

De forma general, se define la función de influencia de la siguiente manera:

$$L_t(y, F) = \lim_{\epsilon \to 0} \frac{t\{(1-\epsilon)F + \epsilon\delta_y\} - t(F)}{\epsilon}$$

donde se entiende aquí al parámetro de interés como un funcional de la distribución F, es decir $\theta = t(F)$ y δ la masa puntual en y. Por ejemplo, $\theta = E(X)$ corresponde al funcional

$t(F) = \int x dF(x)$, de modo que

$$t\{(1-\epsilon)F + \epsilon\delta_x\} = (1-\epsilon)\theta + \epsilon x.$$

Se obtiene entonces que la función de influencia para la media θ de la distribución F es:

$$L_t(x) = x - \theta.$$

Por otro lado, es posible definir también la función de influencia empírica $l(y) = L_t(y, \widehat{F})$ donde $\widehat{F}$ es la distribución empírica. Los valores particulares $l_i = l(y_i)$ se denominan *valores de influencia empíricos*. En el caso de la media, la función de influencia empírica y los valores de influencia empírcos resultan ser $l(x) = x - \bar{x}$ *y* $l_i = x_i - \bar{x}$.

3.4.2 Jacknife-after-bootstrap

Otro enfoque para aproximar los valores de influencia empíricos es el enfoque jacknife. En este caso l_i es aproximado por:

$$l_{jack,i} = (n-1)(\widehat{\theta} - \widehat{\theta}_{(-i)})$$

donde $\widehat{\theta}_{(-i)}$ es la estimación de θ cuando x_i ha sido omitido del conjunto de datos. Por ejemplo, para el caso de la media, $\widehat{\theta} = \bar{x}$ y $\widehat{\theta}_{(-i)} = (n\bar{x} - x_i)/(n-1)$, por ende, $l_{jack,i} = x_i - \bar{x}$ al igual que ocurría con los valores l_i.

El análisis de sensibilidad, es decir el análisis de la influencia que pequeños cambios tienen en el resultado global, es importante para entender las implicancias de un cálculo estadístico. En general, una conclusión que repose fuertemente en pocos valores suele ser menos interesante que una que dependa de todo el conjunto de datos. Bajo el ángulo de modelos paramétrico, existen una serie de medidas de diagnóstico que permiten este análisis pues se conoce una distribución con la que comparar resultados. El caso no paramétrico exige ideas distintas pues el modelo viene dado por la distribución empírica de los datos y no hay, por ende, base con la que comparar los resultados, incluyendo la detección de valores atípicos, por ejemplo.

Una pregunta que se intenta responder es qué hubiese ocurrido si los cálculos hubiesen sido realizados sin la observación x_i por ejemplo. Los límites del intervalo de confianza habrían cambiado notablemente? Una forma de estudiar este problema es comparando resultados con toda la información disponible contra los mismos resultados omitiendo alguna de las observaciones. Como la probabilidad de que un dato dado no aparezca en una

muestra bootstrap es de $(1-\frac{1}{n})^n \simeq e^{-1} \simeq 0.368$, si n es grande, el número de muestras bootstrap que no incluyen al dato en B remuestreos es de aproximadamente $0.368B$. Sea $\mathcal{B}$ el conjunto de todas las muestras bootstrap obtenidas, es decir, $\#\mathcal{B} = B$, y se define $\mathcal{B}_{-(i)}$ como el subconjunto de $\mathcal{B}$ formado por aquellas muestras que no contienen a x_i, es decir, $\mathcal{B}_{-(i)} = \left\{(x_1^*, ..., x_n^*) : x_j^* \neq x_i \ \forall j\right\}$. Se indica por $B_{(-i)} = \#\mathcal{B}_{-(i)}$. Enonces, el efecto de x_i en el sesgo puede ser analizado por:

$$Sesgo_i = n\left\{\frac{1}{B_{(-i)}}\sum_{b\in\mathcal{B}_{-(i)}}(\widehat{\theta}_b^* - \widehat{\theta}_{(-i)}) - \frac{1}{B}\sum_{b=1}^{B}(\widehat{\theta}_b^* - \widehat{\theta})\right\},$$

donde $Sesgo_i$ es la estimación del sesgo sin la observación x_i y donde la primera suma se realiza sólo sobre las muestras que no incluyen esta observación. Si $B = 1000$, habrá entonces aproximadamente 368 muestras que cumplan con este requisito. Queda claro que para este tipo de análisis se exige un valor importante de remuestreos. En el siguiente ejemplo se describirá el gráfico que toma el nombre de esta sección, el gráfico del *jacnkinfe-after-bootstrap*.

3.4.3 Ejemplo: Jacknife en el conjunto de datos de la escuela de leyes

Se hace uso en este ejemplo de un conjunto de datos extraído de la librería *bootstrap* de R con el comando *law*. Se tienen 15 datos de resultados académicos en dos tipos de exámenes mientras que el parámetro de interés es la correlación entre los mismos. Los exámenes del conjunto de datos son el LSAT, que es el promedio sobre todas las clases de cada escuela en un test nacional sobre leyes, y el GPA, el promedio de la licenciatura realizado sobre todas las clases de cada escuela, ambos pertenecientes a la escuela de leyes en Estados Unidos en el año 1973. El valor de la correlación muestral $\widehat{\theta}$ entre los dos conjuntos de datos es $\widehat{\theta} = 0.776$. La Figura 3.5 realiza un análisis de sensibilidad del modelo y tiene por interés comprender como varían los cuantiles empíricos de la distribución de $\widehat{\theta}^* - \widehat{\theta}$ cuando un dato es eliminado. El análisis de esta distribución es de vital interés en las metodologías bootstrap. Es, por ejemplo, la base con la que se realizan los intervalo de confianza bootstrap y es la manera que se tiene de estimar la distribución del estimador centrada.

R posee una función que realiza el gráfico de forma directa. Para realizar este ejemplo se ha hecho uso del siguiente código (la función *corr* es una función de la librería boot que calcula la correlación eventualmente con pesos asignados):

```
library(boot)
```

```
library(bootstrap)
law.boot <- boot(law, corr, R=1000, stype="w")
law.L <- empinf(data=law, statistic=corr)
split.screen(c(1,2))
screen(1)
split.screen(c(2,1))
screen(4)
attach(law)
plot(LSAT,GPA,type="n")
text(LSAT,GPA,round(law.L,2))
text(LSAT,GPA,round(law.L,2))
screen(3)
plot(LSAT,GPA,type="n")
text(LSAT,GPA,1:nrow(law))
screen(2)
jack.after.boot(boot.out=law.boot,useJ=F,stinf=F, L=law.L)
par(mfrow=c(1,1))
```

A través de los gráficos presentes se puede leer información diversa. Por ejemplo, el valor $l_{jack,1} = -1.51$, en rojo, indica que la ausencia del dato 1, en rojo, en el cálculo de $\widehat{\theta}$ (la correlación estimada) tiene por efecto el aumento del valor del mismo. Mientras que $l_{jack,13} = 0.43$ indica el efecto contrario con la ausencia de la décimo tercera observación.

Por otro lado, gracias al gráfico jacnkife-after-bootstrap se puede ver como la ausencia del dato 1, el primer elemento desde la izquierda, tiende a comprimir la distribución de $\widehat{\theta}^* - \widehat{\theta}$ pues los primeros cuantiles son más grandes que los cuantiles con toda la información mientras que los últimos cuantiles son más pequeños. De hecho, es el dato con valor de influencia más grande en valor absoluto. Es fácil distinguir su importancia en la estimación del parámetro contrastando el valor de los cuantiles de la distribución de $\widehat{\theta}^* - \widehat{\theta}$ sin este dato con los cuantiles de los demás casos.

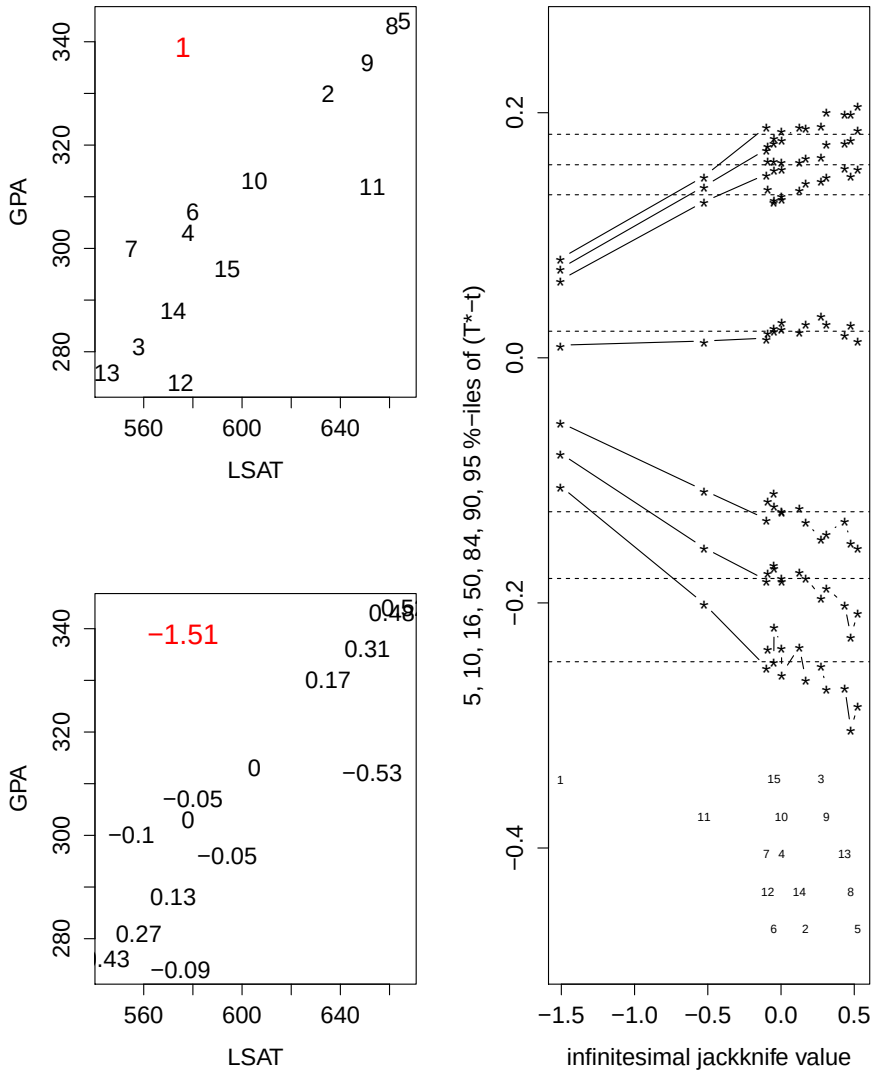


Figura 3.5: *La figura arriba a la izquierda realiza el gráfico de dispersión de los datos indicando a qué observación corresponde cada posición. La figura de abajo a la izquierda realiza el mismo gráfico de dispersión aunque esta vez se indica sobre la misma el valor de los valores de influencia empíricos. Por último, la figura de la derecha corresponde al gráfico jacknife-after-bootstrap. Éste analiza el cambio de los cuantiles de la distribución de* $\widehat{\theta}^* - \widehat{\theta}$ *cuando algún dato fue eliminado y se grafica contra los valores de influencia jacknife.*

Capítulo 4

Bootstrap en regresión lineal

El análisis de modelos de regresión ocupa un papel muy importante en el análisis estadístico. Se estudian los efectos de variables explicativas o covariables sobre una variable respuesta. El modelo de regresión lineal puede ser descripto de la siguiente manera:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_{p-1} x_{i,p-1} + \epsilon_i, \quad i = 1, ..., n, \tag{4.1}$$

donde $E(\epsilon_i) = 0$, ϵ_i son independientes y usualmente $Var(\epsilon_i) = \sigma^2$ con σ^2 un valor constante. Es común considerar a los x_{ij} como valores fijos, interpretados como parte del diseño del modelo aunque pueden ser valores observados de $p-1$ variables aleatorias. Si bien la esperanza de los errores se define siempre de esta forma, la varianza de los mismos podría no ser constante, es decir, el modelo podría no ser homoscedástico. El problema de heteroscedasticidad se analizará en la Sección 4.5 y por lo pronto se asumirá varianza común en los errores del modelo lineal. Para el caso en que $\epsilon_i \sim N(0, \sigma^2)$ existe una amplia bibliografía al respecto, ver por ejemplo Neter (2004), Scheffé (1959) y Seber (1977). La presentación que se dará tiene especial interés cuando no es posible aseverar normalidad de los errores o se tenga creencia sobre la no-normalidad de los mismos.

El modelo (4.1) también puede escribirse en notación vectorial como:

$$y_i = x_i^t \beta + \epsilon_i \quad i = 1, ..., n, \tag{4.2}$$

con $x_i^t = (1, x_{i1}, x_{i2}, ..., x_{i,p-1})$ y $\beta^t = (\beta_0, ..., \beta_{p-1})$ o, aún en notación matricial, como

$$Y = X\beta + \epsilon, \tag{4.3}$$

donde $Y^t = (y_1, ..., y_n)$, X es la matriz de $n \times p$ cuyas filas están dadas por x_i^t y ϵ es el vector de los residuos con componentes ϵ_i.

Se destacan algunas características particulares de la regresión múltiple:

1. El problema de selección de variables que se analizará en el Capítulo 6.
2. El estudio de problemas de predicción que se desarrollará en el Capítulo 5.

Este capítulo focaliza particularmente su interés en las técnicas de remuestreo bootstrap que permitan obtener réplicas bootstrap de los estimadores de los coeficientes de regresión del modelo de regresión lineal. Esto permitirá estimar características del estimador $\widehat{\beta}$ en relación al parámetro β. En primer lugar, se presenta a continuación la metodología clásica para el cálculo de $\widehat{\beta}$ a partir del cual se focalizará el análisis bootstrap.

4.1 Estimación por mínimos cuadrados

Los estimadores clásicos de los parámetros en un modelo de regresión se obtienen por mínimos cuadrados. Por simplicidad, se supondrá que la matriz de diseño, es decir la matriz X en (4.3), es de rango completo aunque también es posible trabajar con la inversa generalizada. El estimador de mínimos cuadrados de β, que bajo normalidad es equivalente al estimador de máxima verosimilitud, es igual a:

$$\widehat{\beta} = (X^t X)^{-1} X^t Y.$$

Definición 4.1. El vector de valores predichos, es decir, el vector de valores en la variable respuesta que permiten estimar la respuesta teórica de una observación a partir del modelo ajustado, se define como

$$\widehat{Y} = (\widehat{y_1}, ..., \widehat{y_n})^t = (x_1^t \widehat{\beta}, ..., x_n^t \widehat{\beta})^t = X\widehat{\beta}.$$

Además, $\widehat{Y}$ es la proyección de Y en el subespacio de las columnas de X, es decir:

$$\widehat{Y} = PY, \quad P = X(X^t X)^{-1} X^t,$$

donde la matriz P se denomina matriz de proyección o *hat matrix*.

Definición 4.2. Se define el residuo $i-$ésimo del modelo como la diferencia entre el valor predicho $i-$ésimo y el valor observado $i-$ésimo, de forma que el vector de residuos es

$$r = (y_1 - \widehat{y}_1, ..., y_n - \widehat{y}_n)^t.$$

Por último, por razones que se explicarán en la Sección 4.2, se definen los residuos estandarizados como

$$\widetilde{r}_i = \frac{r_i}{(1-p_{ii})^{1/2}},$$

donde p_{ii} es la componente $i-$ésima de la diagonal de la matriz de proyección P notados h_i y denominados *leverages* o palanca. Se puede observar que $Var(\widetilde{r}_i) = \sigma^2$, pues $\Sigma_r = \sigma^2(I-P)$ donde Σ_r es la matriz de covarianzas de r.

4.2 Métodos de remuestreo Bootstrap

Existen dos métodos de remuestreo bootstrap clásicos en modelos de regresión que se presentan a continuación. El uso de uno por sobre el otro dependerá de la naturaleza de las covariables del modelo y de la interpretación de éste. En la Sección 4.2.3 de este mismo capítulo se pueden encontrar razones que justifiquen el uso de uno u otro método. Para un análisis respecto de las características de los estimadores de los coeficientes de regresión cuando la teoría normal no se justifica es de gran utilidad el proceso de remuestreo y la metodología de los mismos es presentada a continuación.

4.2.1 Remuestreo a partir de los residuos

Este método supone que se cumplen los supuestos del modelo lineal, es decir que se cumplen los supuestos en (4.1) y que la varianza de los errores es una constante desconocida σ^2. Cuando la teoría normal no es aplicable, en general, es difícil tener precisión respecto de las características de $\widehat{\beta}$ tales desvío o intervalos de confianza. Por ello, aquí se presenta un primer método de remuestreo bootstrap que permita aproximar dichas características. La idea consiste en, considerando fijas las observaciones x_i^t, remuestrear a partir de los errores que en definitiva son quienes dan estructura al modelo. Desafortunadamente, no se tiene acceso al valor de los últimos. Por esta razón, se definieron los residuos que aproximan a los errores del modelo. Es natural entonces pensar en remuestrear los residuos. Aún así, no es exactamente esto lo que se hará y para entender por qué se detallan a continuación ciertas características de los residuos.

$$r = (I-P)Y,$$

y por ende, $E(r) = (I-P)E(Y) = 0$. Además, $\Sigma_r = (I-P)\sigma^2$ la matriz de covarianzas de los residuos. De aquí se tiene que $Var(r_i) = (I-P)_{ii}\sigma^2$.

Los métodos bootstrap deben replicar el proceso de generación de datos original lo mejor posible. Si se remuestrease a partir de los residuos no se estaría respetando esta condición:

los errores se saben independientes y homoscedásticos mientras que, como se vio, los residuos tienen covarianzas distintas de 0 y varianzas distintas. Por ello, se definieron anteriormente los denominados residuos estandarizados

$$\widetilde{r}_i = \frac{r_i}{(1-p_{ii})^{1/2}},$$

que tienen igual varianza. Si se llama G a la distribución de los errores, la independencia de los residuos se logrará al tomar muestras aleatorias de la distribución aproximada $\widehat{G}$ en el proceso bootstrap. Como última observación, una vez estandarizados los residuos, es importante centrarlos de forma que tengan esperanza igual a 0. El algoritmo bootstrap por remuestreo por residuos para modelos de regresión lineal en el caso no paramétrico se define de la siguiente manera:

Algoritmo 4.1

Para $b = 1, ...B$:

Para $i = 1, ..., n$:

1) Tome $x_i^* = x_i$

2) Tome r_i^* de forma aleatoria a partir de los residuos estandarizados y centrados $\widetilde{r}_1$-$\bar{\widetilde{r}}$,...,$\widetilde{r}_n$-$\bar{\widetilde{r}}$

3) Considere $y_i^* = x_i^t \widehat{\beta} + r_i^*$

Utilizando $(x_1^*, y_1^*), ..., (x_n^*, y_n^*)$ calcule por mínimos cuadrados $\widehat{\beta}^*$.

Los vectores bootstrap $r^* = (r_1^*, ..., r_n^*)^t$ recuperan la propiedad de independencia que gozaban los errores del modelo al haber sido tomados de forma aleatoria a partir de $\widehat{G}$. En el algoritmo se ha utilizado $\widehat{G}$ como la distribución empírica de $\widetilde{r}_1$-$\bar{\widetilde{r}}$,...,$\widetilde{r}_n$-$\bar{\widetilde{r}}$, en el que se usa además el mismo diseño original del problema. Esto es, $x_i^* = x_i$ para todo $1 \leq i \leq n$ y es porque se han considerado a los x_i como valores fijos. Esta será una gran diferencia con el siguiente método respecto del cálculo de las réplicas bootstrap de $\widehat{\beta}$.

Algunas propiedades

Es fácil ver que $E(\widehat{\beta}) = \beta$ y como $\Sigma_{\widehat{\beta}} = \sigma^2(X^tX)^{-1}$, entonces $Var(\widehat{\beta}_i) = \sigma^2((X^tX)^{-1})_{ii}$. Se recuerda que $\widehat{\beta}^*$ es el minimizador de:

$$\sum (y_i^* - x_i^t\widehat{\beta})^2.$$

Entonces, se puede ver que $\widehat{\beta}^* = (X^tX)^{-1}X^tY^* = (X^tX)^{-1}X^t(\widehat{Y}+r^*) = \widehat{\beta}+(X^tX)^{-1}X^tr^*$. Como $E_{\widehat{F}}(r_i^*) = \frac{1}{n}\sum r_j = 0$, se deduce entonces que $E_{\widehat{F}}(\widehat{\beta}^*) = \widehat{\beta}$. Con respecto a la varianza se puede ver que $\Sigma_{\widehat{\beta}^*} = (X^tX)^{-1}X^t\Sigma_{Y^*}X(X^tX)^{-1} = (X^tX)^{-1}\widehat{\sigma}^2$ donde $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (r_i - \bar{r})^2$. Este es un caso en donde no se necesita simulación para obtener la estimación bootstrap de la varianza de $\widehat{\beta}$.

4.2.2 Remuestreo por pares

Un enfoque distinto consiste en imaginar a los datos como muestras aleatorias e independientes de una distribución multivariada F, es decir, $\{(x_i, y_i)\} \sim F$. En el caso no paramétrico, el estimador $\widehat{F}$ de F debe ser la distribución empírica de los datos y por ende, el algoritmo para obtener las replicaciones bootstrap no es muy distinto de los algoritmos propuestos en Capítulos previos.

Algoritmo 4.2

Para $b = 1, ...B$:

1) Tome $1^*, ..., n^*$ de forma aleatoria y con reemplazo a partir de $\{1, ..., n\}$.
2) Para $j = 1, ..., n$, tome $x_j^* = x_{j^*}$, $y_j^* = y_{j^*}$
3) Por último, obtenga $\widehat{\beta}^*$ por mínimos cuadrados a partir de $(x_1^*, y_1^*), ..., (x_n^*, y_n^*)$.

En el caso de tener pocas observaciones es probable que la matriz $(X^{*t}X^*)$ resulte singular. En ese caso no se podrán calcular los estimadores de los coeficientes de regresión con el método de mínimos cuadrados por lo que se deberá generar una nueva muestra bootstrap. Se tiene además que la estimación bootstrap de la varianza viene dada por:

$$\Sigma_{\widehat{\beta}^*} = \sum_{b=1}^{B} (\widehat{\beta}_b^* - \widehat{\beta}_{(\cdot)}^*)^t(\widehat{\beta}_b^* - \widehat{\beta}_{(\cdot)}^*)/B$$

con $\widehat{\beta}_{(\cdot)}^* = 1/B\sum_{b=1}^B \widehat{\beta}_b^*$.

4.2.3 Diferencias entre métodos

Dos diferencias importantes sobresalen en las dos formas de obtener replicaciones bootstrap. En primer lugar, el segundo método no asume homoscedasticidad. Es más, en ningún momento se asume que la esperanza de la respuesta es lineal en x. Esto permite que el método pueda aplicarse aún en casos de heteroscedasticidad aunque resulta ineficiente si los supuestos del modelo lineal son correctos. En segundo lugar, el diseño no está fijo. El método de pares replica la información suponiendo aleatoriedad en la covariable mientras que el método de residuos fija la información original. En este sentido, ninguno es *mejor* que el otro. Respecto a este segundo ítem, dependerá de cada caso y de la interpretación de las variables la preferencia de un método frente a otro.

El método de remuestreo por pares es frágil cuando se tienen pocas observaciones pues, como se dijo, no será siempre posible utilizar el método de mínimos cuadrados en cada muestra bootstrap. El ejemplo siguiente subraya este inconveniente en un ejemplo extraído de Davison y Hinkley (1997).

4.3 Evaluación de diagnóstico del *Six-Minute Walk Test (6MWT)* en pacientes adultos con enfermedad cardíaca congénita (GUCH)

El ejemplo de Kehmeier *et al* (2014), propone el análisis de una nueva metodología para el diagnóstico de personas con enfermedad cardíaca congénita. El seguimiento a largo plazo y la toma de decisiones para la intervención en los pacientes gana importancia y las estrategias para identificar condiciones de deterioramiento son una cuestión central en este contexto. El método de mayor uso para el testeo de las funciones cardíacas es el *ejercicio de testeo cardiopulmonar (CPX)* que propone el análisis del pico de consumo de oxígeno (peakVO2) en el esfuerzo físico. De hecho se ha podido observar que la variable peakVO2 obtenida a partir del CPX es un predictor de la hospitalización y fallecimiento en GUCH. De esta forma, CPX es recomendado para testear pacientes GUCH en la actual guía de la Sociedad Europea de Cardiología. El problema del CPX es que exige un esfuerzo considerable lo que para pacientes con problemas cardíacos puede ser muy contraproducente. En contraste, el método del six-minute walk es simple, de fácil acceso, de poco costo y sin riesgo en el esfuerzo para los pacientes. El método calcula la cantidad de metros recorridos por el paciente en 6 minutos de caminata.

El objetivo del estudio es comparar el 6MWT con el CPX, el último estimado como el gold standard. Se ha considerado entonces un modelo de regresión lineal simple en el

que la cantidad de metros recorridos en el 6MWT es la variable explicativa x y el pico de consumo de oxígeno, $peakVO2$, calculado a partir del CPX, es la variable respuesta. Los datos han sido aportados por el hospital de Duesseldorf entre junio de 2013 y julio de 2014 y se cuentan $n = 103$ pacientes. El análisis y los resultados son originales de esta tesis y la información es manejada en R utilizando las siguientes variables :

```
> head(dat)
  x.6MWD peakVO2
1    572    14.5
2    402    18.4
3    485    18.6
4    646    28.5
5    665    29.9
6    480    11.3
```

Por ejemplo, se tiene que el primer paciente ha caminado 572 metros en el 6MWT y ha tenido un pico de consumo de oxígeno de 14.5ml/kg/min en el CPX. En la Figura 4.1 se muestra el diagrama de dispersión de los datos en el que se ha superpuesto la recta de regresión por mínimos cuadrados. El resultado del comando summary del ajuste lineal parece confirmar la significatividad de la relación entre ambas variables que puede contemplarse a simple vista con el diagrama de dispersión.

```
summary(m1)

Call:
lm(formula = peakVO2 ~ x.6MWD, data = dat)

Residuals:
    Min      1Q  Median      3Q     Max
-8.9022 -3.0895 -0.5912  3.3634 11.6964

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.668112   2.078366  -0.321    0.749
x.6MWD       0.042081   0.004013  10.487   <2e-16 ***

Residual standard error: 4.784 on 100 degrees of freedom
Multiple R-squared:  0.5238,Adjusted R-squared:  0.519
F-statistic:   110 on 1 and 100 DF,  p-value: < 2.2e-16
```

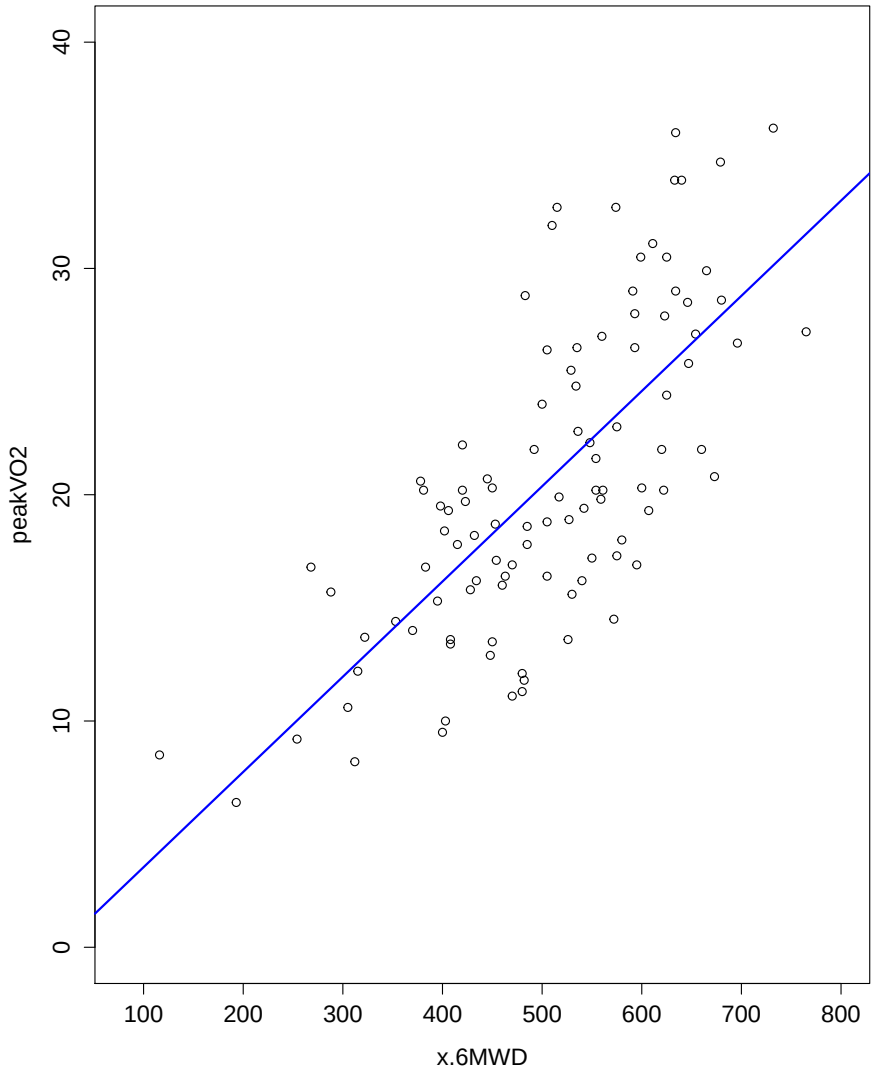


Figura 4.1: *Diagrama de dispersión de los datos de los 103 pacientes sometidos al 6MWT y al CPX.*

Se propone aquí el uso del método bootstrap para estimar la distribución de los coeficientes de regresión y comparar los resultados aportados por los métodos de residuos y pares. En una primera instancia, se propone el método por residuos y se calculan 20 réplicas de las rectas de mínimos cuadrados. El siguiente código permite lo deseado.

```
m1 <- lm(peakVO2~x.6MWD, data = dat)
m1.boot.residuals <- Boot(m1, R = 1000,
                          method = "residual")
a1.r <- m1.boot.residuals$t[, "(Intercept)"]
b1.r <- m1.boot.residuals$t[, "x.6MWD"]
for(i in 1:20)
+ {
+   curve(a1.r[i] + b1.r[i]*x, from = 100, to = 750,
 add = TRUE, col = "green", lty =2)
+ }
```

La función `lm` realiza un ajuste por mínimos cuadrados entre la variable dependiente peakVO2 y la variable regresora x.6MWT. En este caso se ha utilizado la función `Boot` de la librería *car* de R que genera las réplicas bootstrap de los coeficientes de regresión de forma más directa. Es posible especificar, en el comando, el método a usar (pares o residuos). Por último, la función curve permite graficar en este caso las rectas de regresión generadas lo que puede apreciarse en la Figura 4.2.

Lo mismo puede realizarse con la metodología de pares. El código a usar es esencialmente el mismo, especificando en la función `Boot` la metodología adecuada. El gráfico con las rectas por el método de pares puede apreciarse en la Figura 4.3 mientras que en la Figura 4.4 se han superpuesto al diagrama de dispersión las rectas generadas por ambos métodos. Se puede ver que la diferencia entre ambos métodos es insignificante a niveles prácticos si bien la teoría establece claras diferencias. De hecho, este ejemplo sería afín al uso del método de pares por la naturaleza de la covariable pero los resultados no muestran diferencias significativas.

El tamaño muestral, $n = 103$ es grande y podría ser la causa principal en la falta de distinción entre los métodos. Aún así, repitiendo los cálculos para una muestra aleatoria de tamaño 30 se recuperan las mismas particularidades. Esto puede observarse en la Figura 4.5. La mayor variabilidad de las rectas en los valores pequeños tiene que ver en este caso con la casi nulidad de datos para dicho sector. Por último, siguiendo con el mismo esquema, en la Figura 4.6 se exhiben las densidades estimadas de las réplicas bootstrap de los coeficientes de regresión en donde se observa nuevamente la similitud entre ambos métodos.

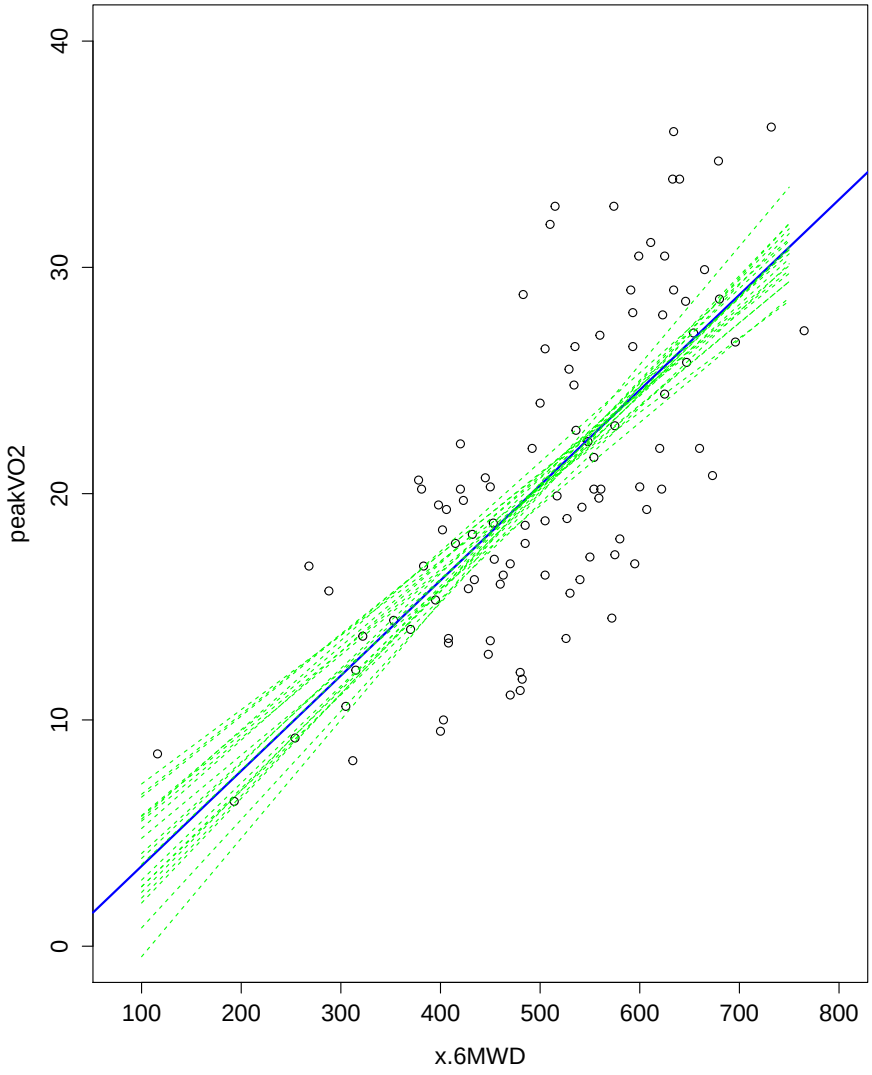


Figura 4.2: *Diagrama de dispersión de los datos de los 103 pacientes sometidos al 6MWT y al CPX en el que se han superpuesto la recta de regresión por mínimos cuadrados y las rectas de regresión bootstrap bajo el método de residuos en verde.*

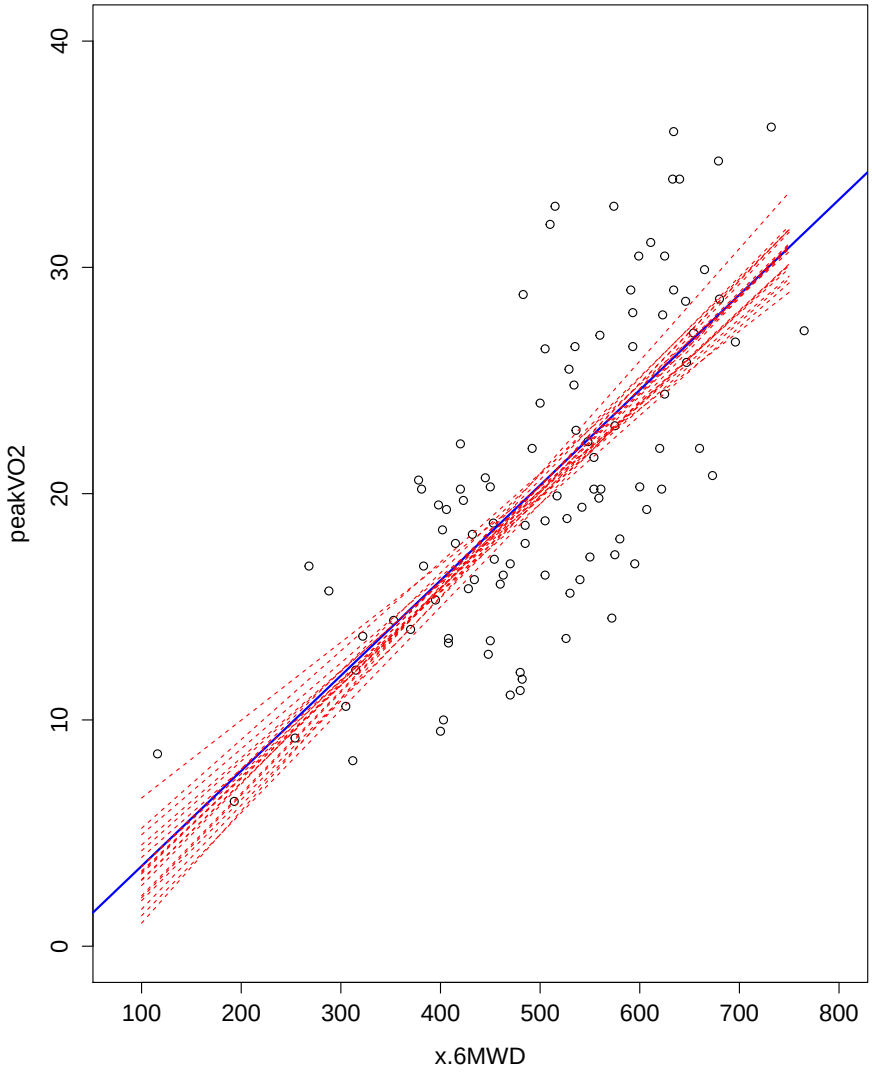


Figura 4.3: *Diagrama de dispersión de los datos de los 103 pacientes sometidos al 6MWT y al CPX en el que se han superpuesto la recta de regresión por mínimos cuadrados y las rectas de regresión bootstrap bajo el método de pares en rojo.*

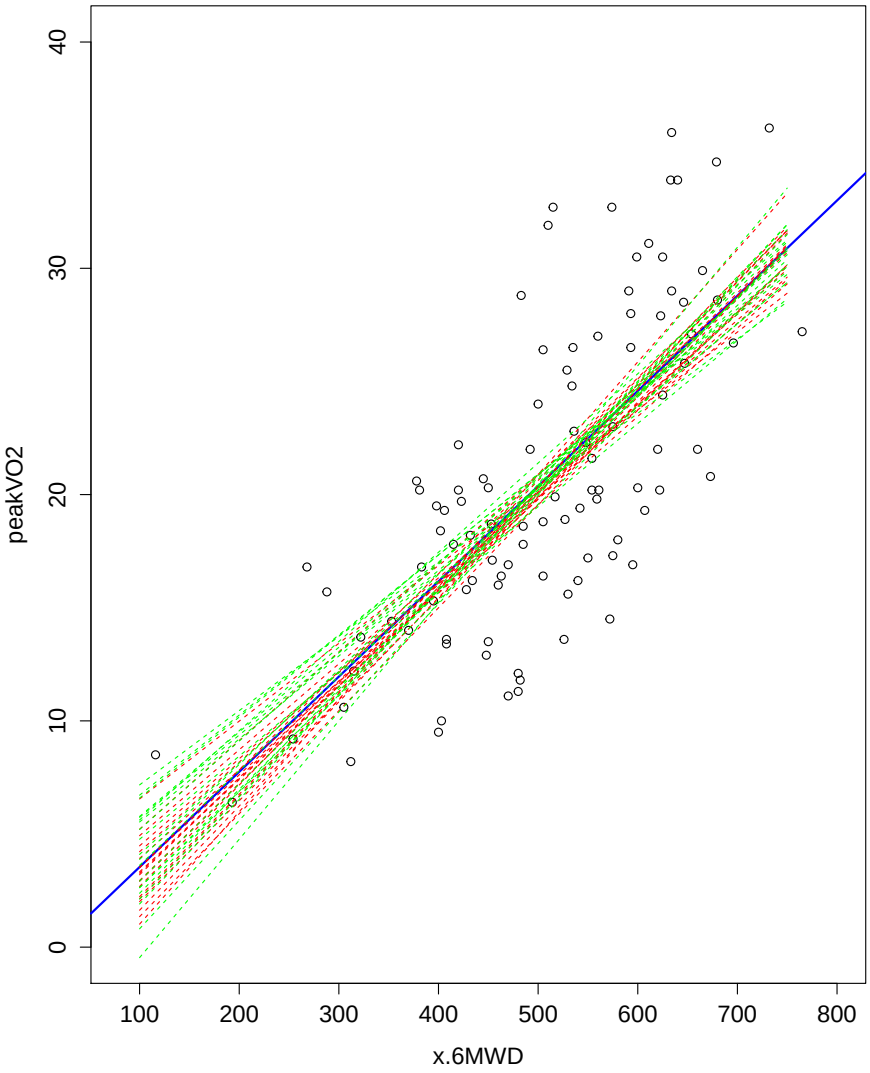


Figura 4.4: *Diagrama de dispersión de los datos de los 103 pacientes sometidos al 6MWT y al CPX en el que se han superpuesto la recta de regresión por mínimos cuadrados y las rectas de regresión bootstrap bajo el método de residuos en verde y bajo el método de pares en rojo.*

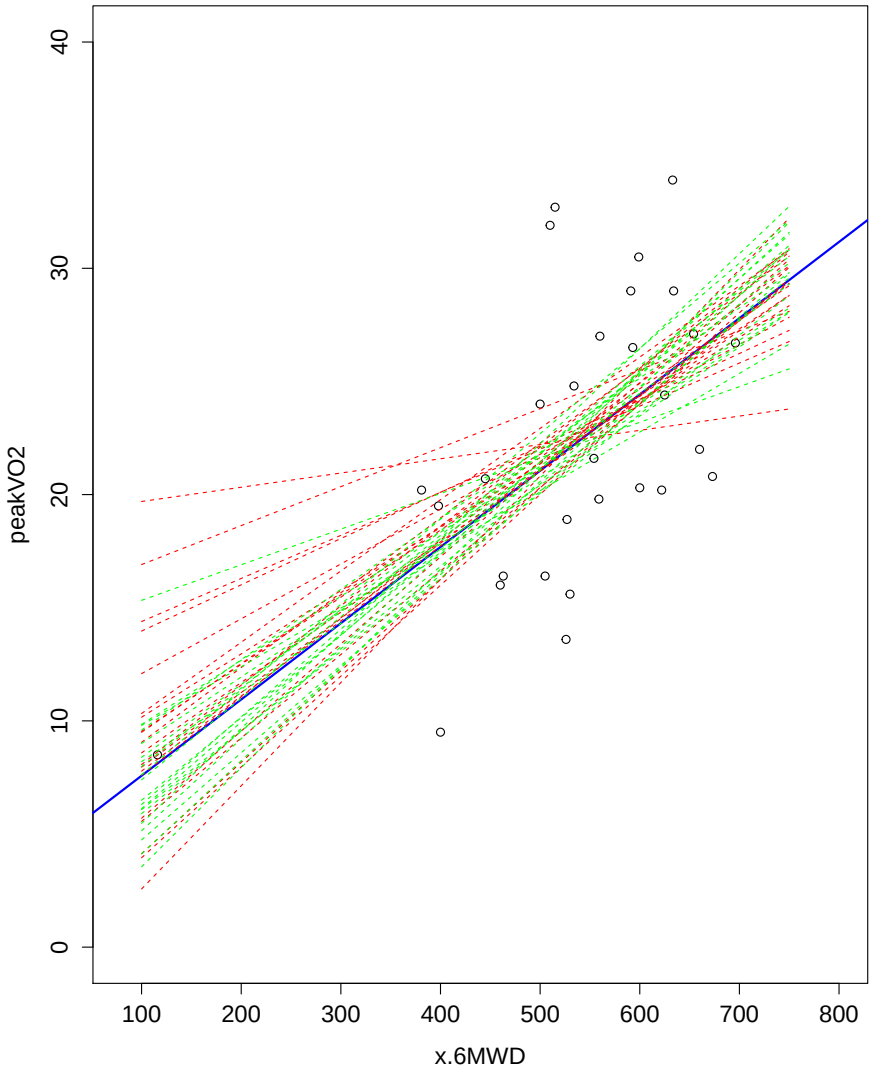


Figura 4.5: *Diagrama de dispersión de una muestra aleatoria de tamaño 30 de los datos de los 103 pacientes sometidos al 6MWT y al CPX en el que se han superpuesto la recta de regresión por mínimos cuadrados y las rectas de regresión bootstrap bajo el método de residuos en verde y bajo el método de pares en rojo.*

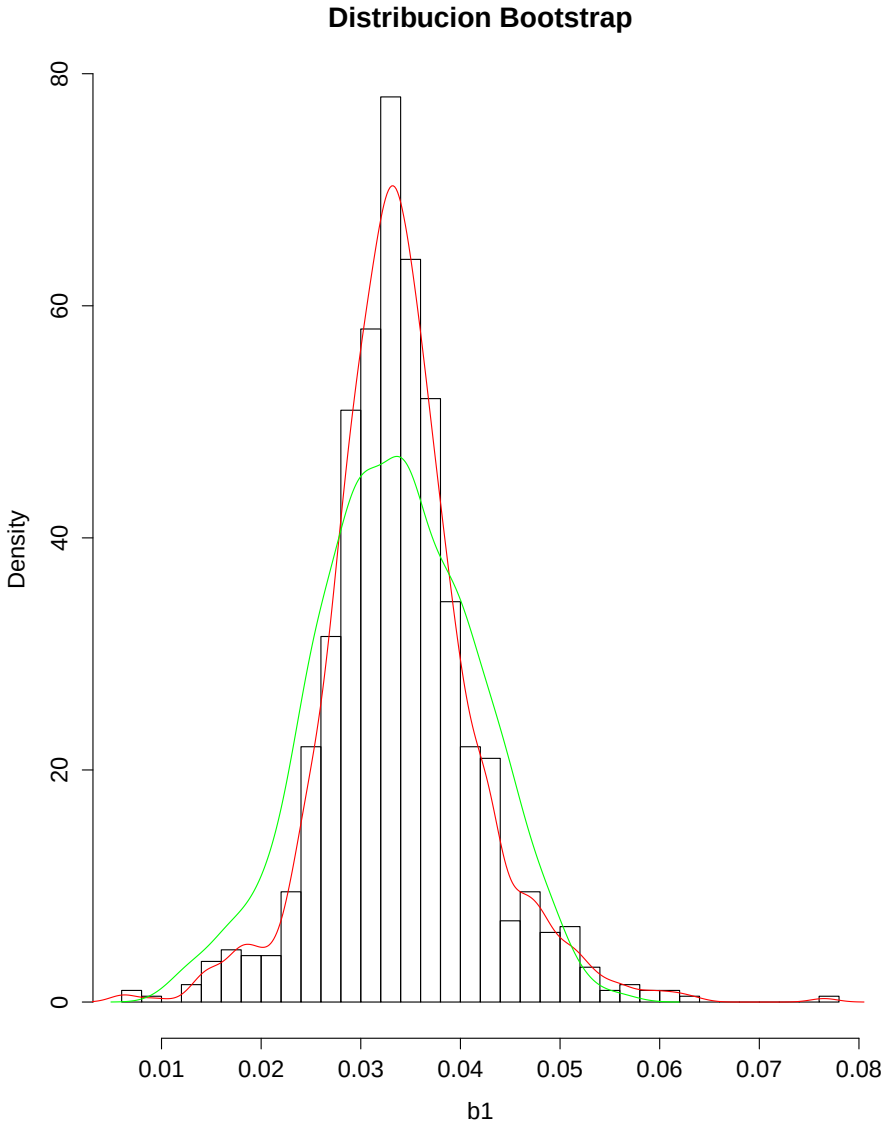


Figura 4.6: *Histograma del coeficiente de regresión bootstrap $\hat{\beta}_1^*$ bajo el método de pares. Se ha superpuesto además la densidad del mismo en rojo y la densidad del bootstrap del mismo estimador con el método de residuos en verde.*

	x_1	x_2	x_3	x_4	y
1	7	26	6	60	78.5
2	1	29	15	52	74.3
3	11	56	8	20	104.3
4	11	31	8	47	87.6
5	7	52	6	33	95.9
6	11	55	9	22	109.2
7	3	71	17	6	102.7
8	1	31	22	44	72.5
9	2	54	18	22	93.1
10	21	47	4	26	115.9
11	1	40	23	34	83.8
12	11	66	9	12	113.3
13	10	68	8	12	109.4

Tabla 4.1: *Datos extraídos de Woods, Steinour and Starke, 1932. La respuesta y es el calor (calorías por gramo de cemento) en un conjunto de muestras de cemento. Las variables explicativas son el porcentaje por peso de cuatro constituyentes del cemento.*

4.4 Problemas en el caso de remuestreo por pares

Las estimaciones bootstrap pueden no ser buenas en el caso de matrices de diseño bootstrap con autovalores muy pequeños. Al muestrear las filas de X se pueden obtener matrices $X^{t*}X^*$ casi no inversible lo que genera engrosamiento de algunos de los desvíos estimados bootstrap de $\widehat{\beta}$. Una posible solución a este incoveniente podría ser eliminar las matrices $X^{t*}X^*$ con mínimo autovalor l_1^* más chico que una cierta proporción del mínimo autovalor de X^tX. Davison y Hinkley (1997) proponen, por otro lado, quedarse con las matrices X^* con autovalores mínimos l_1^* incluidos en el 50% central del total de los valores de los mínimos autovalores obtenidos. Se presenta a continuación un ejemplo en el que se analiza este problema. La Tabla 4.1 presenta una variable respuesta y correspondiente a las calorías por gramo de cemento y 4 variables explicativas que representan el porcentaje por peso de 4 constituyentes del cemento. Los datos se han extraído de la librería *boot* de R. El dato interesante aquí es que el mínimo autovalor l_1 de X^tX es igual a 0.0012 de forma que $l_{min}/l_{max} = 2.73e-08$.

En la Figura 4.7 se observan los histogramas de $\widehat{\beta}_0^*, \widehat{\beta}_1^*, \widehat{\beta}_2^*$ y $\widehat{\beta}_3^*$. Se han superpuesto densidades normales ajustadas con estimadores de máxima verosimilitud en cada caso.

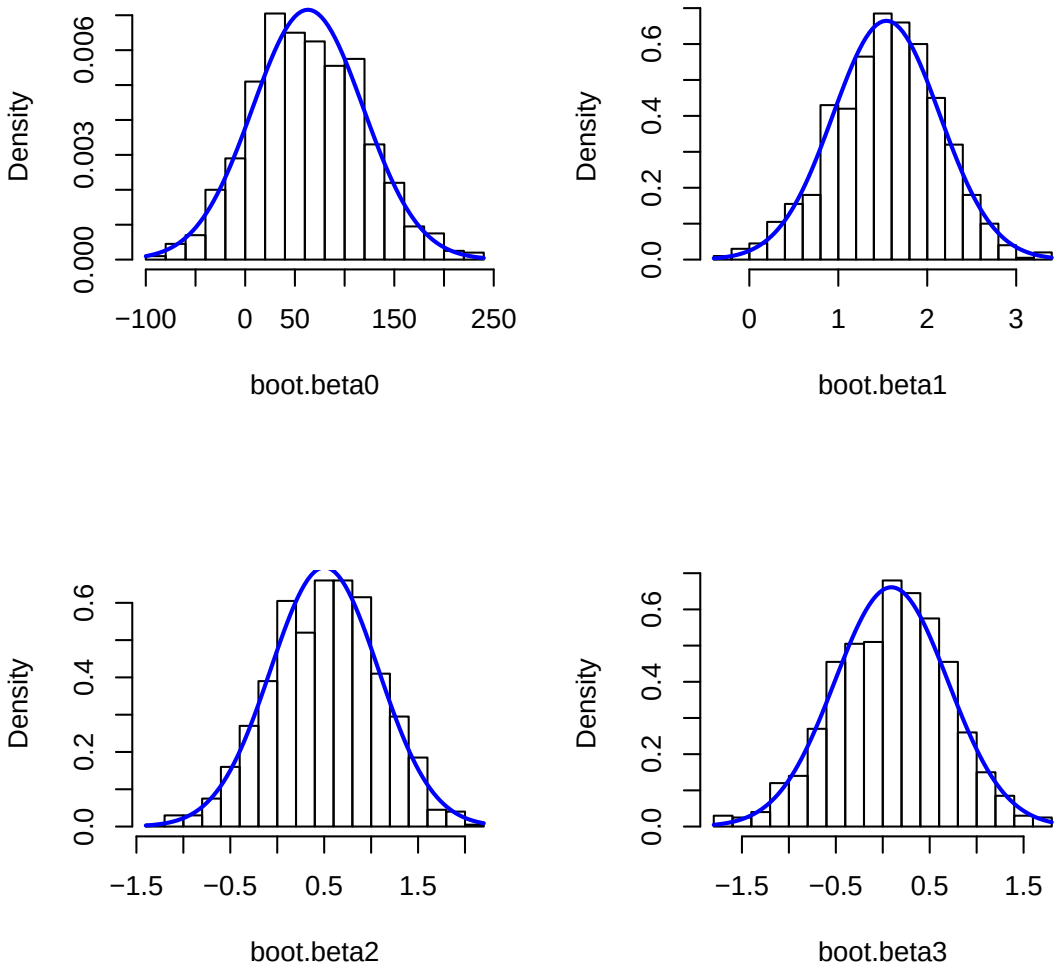


Figura 4.7: *Histogramas de las replicaciones bootstrap de los regresores con densidades normales ajustadas por máxima verosimilitud superpuestas.*

	$\widehat{\beta}_0$	$\widehat{\beta}_1$	$\widehat{\beta}_2$	$\widehat{\beta}_3$	$\widehat{\beta}_4$
Teoría normal	70	0.74	0.72	0.75	0.71
Residuos, R=1000	55.8	0.60	0.57	0.60	0.56
Pares, R=1000	104.48	1.08	1.08	1.14	1.07
Pares, 500 del medio	67.32	0.77	0.69	0.78	0.68

Tabla 4.2: *Desvíos de los estimadores $\widehat{\beta}$ según la teoría normal, el método bootstrap por residuos con B=1000 replicaciones y el método de pares con B=1000 replicaciones. Se han calculado los mismos desvíos con el método de pares pero esta vez considerando únicamente los regresores provenientes del 50% central de las matrices ordenadas por tamaño del autovalor más pequeño.*

Las replicaciones bootstrap fueron obtenidas por el método de pares por la naturaleza de los datos y parecen distribuirse normalmente. En la Tabla 4.2 se muestran los desvíos de $\widehat{\beta}$ calculados por la teoría normal (que parece ajustar bien en este caso) y por el método bootstrap en los dos casos estudiados (se han incluido en el cuadro los desvíos obtenidos con el método de residuos, aunque éste no parece ajustarse bien al ejemplo por el hecho de que las covariables no corresponden a un diseño fijo). En los desvíos obtenidos a partir del método de pares se puede ver el efecto de engrosamiento que se anticipó por el valor límite del mínimo autovalor de X^tX.

En las Figuras 4.9 y 4.11 se pueden observar los diagramas de dispersión de los autovalores l_1^* contra los estimadores $\widehat{\beta}_0^*$ y $\widehat{\beta}_1^*$. En las Figuras se aprecia como valores pequeños de l_1^* aumentan la variabilidad de los $\widehat{\beta}_i^*$. De esta forma se explica en parte el interés de Davison y Hinkley (1997) en eliminar las matrices de diseño que generen autovalores tan pequeños. Como se observa en la Tabla 4.2, los desvíos se aproximan más a los desvíos aportados por la teoría normal al considerar sólo las matrices de diseño que generan el 50% central de mínimos autovalores.

4.5 El caso de heteroscedasticidad

En muchas aplicaciones, el modelo lineal debe asumir heteroscedasticidad en los errores aleatorios. En este ejemplo se verá que, si la heteroscedasticidad puede ser modelada, entonces se puede modificar el método bootstrap descripto en el Algoritmo 4.1. En un modelo heteroscedástico se supone que las observaciones (x_i, y_i), $i = 1, ..., n$ son tales que

$$y_i = x_i^t\beta + \sigma_i\, u_i,$$

donde $E(u_i) = 0$, $Var(u_i) = 1$ en el caso de diseño fijo y $E(u_i|x_i) = 0$, $Var(u_i|x_i) = 1$ en el caso aleatorio. Sea $\epsilon_i = \sigma_i u_i$, luego $E(\epsilon_i|x_i) = 0$ y $Var(\epsilon_i|x_i) = \sigma_i^2$.

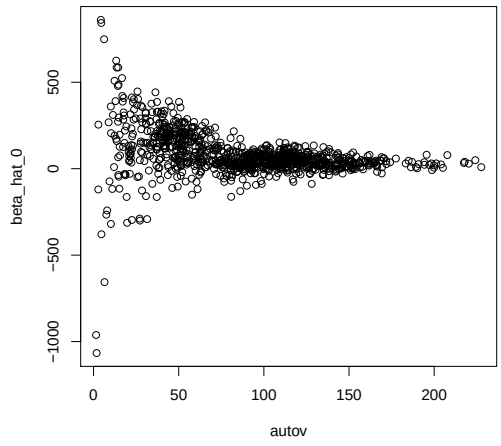


Figura 4.8:

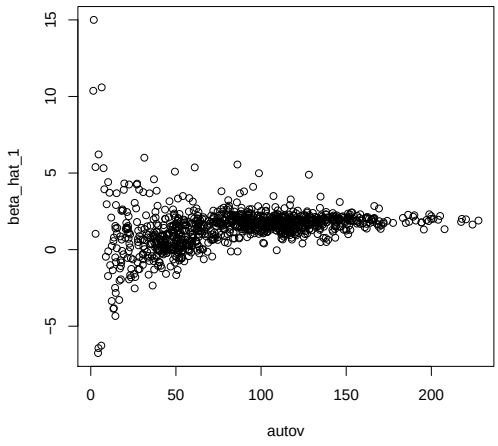


Figura 4.10:

Figura 4.9: *Diagrama de dispersión de los autovalores l_1^* contra los estimadores $\widehat{\beta}_0^*$*

Figura 4.11: *Diagrama de dispersión de los autovalores l_1^* contra los estimadores $\widehat{\beta}_1^*$*

Varianza conocida

Se supone aquí que la distribución de $\epsilon_i|x_i$ es tal que $E(\epsilon_i|x_i) = 0$ y su varianza es igual a $\sigma_j^2 = kV(\mu_j)$ donde $V(.)$ es una función conocida y μ_i es la media de la respuesta $y_i|x_i$. En este caso, los residuos estandarizados deben estimarse por:

$$\widetilde{r}_i = \frac{r_i}{(1-h_i)^{1/2}},$$

donde h_i es el i-ésimo elemento de la diagonal de la matriz de proyección $P = X(X^tV^{-1}X)^{-1}X^tV$ y V es la matriz diagonal de valores $kV(\mu_i)$. La distribución empírica de estos nuevos residuos habiéndoles sustraído su media, es un estimador de la distribución G de los errores homoscedásticos u_i del modelo. Se puede definir entonces el siguiente algoritmo no paramétrico:

Algoritmo 4.3

Para $b = 1, ..., B$:

Para $j = 1, .., n$:

i) Tome $x_j^* = x_j$

ii) Seleccione de forma aleatoria u_j^* a partir de $\widetilde{r}_1 - \bar{\widetilde{r}}, ..., \widetilde{r}_n - \bar{\widetilde{r}}$

iii) Tome $y_j^* = x_j^{*t}\widehat{\beta} + \widehat{\sigma}_j u_j^*$

Finalmente, ajuste por mínimos cuadrados al conjunto de datos $(x_1^*, y_1^*), ..., (x_n^*, y_n^*)$ y obtenga $\widehat{\beta}^*$

Lamentablemente, generalmente en la práctica no se tiene conocimiento de la función $V(\mu)$. En algunas circunstancias, si la cantidad de datos es suficiente, se pueden encontrar patrones en la varianza de los residuos. Esto mismo puede lograrse en el análisis del gráfico de residuos contra los valores predichos. Se puede también utilizar el gráfico de Tukey-Mosteller en búsqueda de una relación entre la varianza y la esperanza de y_i o aún a través de funciones estabilizadoras de la varianza. Para el caso en que no se encuentre un patrón específico, se describe a continuación otro enfoque bootstrap para la generación de réplicas de los estimadores de los coeficientes de regresión.

4.5.1 Wild Bootstrap

Ante la falta de homoscedasticidad, el algoritmo de re-muestreo del bootstrap clásico es inadecuado. Ya no se dispone de n datos *iid* de un único error, sino de un dato único para cada uno de los n errores correspondientes. El *Wild Bootstrap* introducido por Härdle y Mammen (1991,1993) permite remuestrear el modelo a partir de un único dato observado y mejora el comportamiento del bootstrap clásico. En vez de considerar la distribución empírica de los residuos centrados para remuestrear se definen nuevos residuos r_i^* de tal manera que

$$E(r_i^*) = 0, \quad Var(r_i^*) = 1 \quad y \quad E^3(r_i^*) = 1.$$

Definición 4.3. Para que cumplan lo pedido, se definen los residuos del método Wild bootstrap de forma que

$$P\{r_j^* = r_j(1 - \sqrt{5})/2\} = \frac{5 + \sqrt{5}}{10} \quad P\{r_j^* = r_j(1 + \sqrt{5})/2\} = 1 - \frac{5 + \sqrt{5}}{10}.$$

4.6 Ejemplo de aplicación del Wild Bootstrap

Para escenificar el método Wild Bootstrap se han generado 1000 observaciones con errores heteroscedásticos. Para ello se ha considerado el siguiente modelo:

$$y_i = 10 + 3x_i + 2x_i^2 + \sqrt{e^{x_i}/2.5 - 0.4}\, u_i,$$

donde $u_i \sim N(0,1)$ y $x_i \sim U(0,1)$. Las replicaciones se han generado con el comando *wild.boot* del paquete $fANCOVA$.

```
# install.packages("fANCOVA")

library(fANCOVA)
n <- 1000
x <- runif(n, min=0, max=1)

## se generan los errores heteroscedasticos
sig.x <- sqrt(exp(x)/2.5-0.4)
err <- sapply(sig.x, function(x) rnorm(1, sd=x))
x2 <- x^2
y <- 10+3*x+2*x2 +err
```

La Figura 4.12 exhibe el diagrama de dispersión de los valores simulados. En la Figura 4.13 se puede observar también el gráfico de valores predichos versus residuos del modelo ajustado por mínimos cuadrados en donde queda clara la heteroscedasticidad del modelo. Por otro lado, en la Figura 4.14 se ha realizado el mismo gráfico de valores predichos contra residuos pero para un esquema obtenido por re-muestreo a partir de los residuos según la metodología bootstrap clásica dada por el Algoritmo 4.1 que no es adecuado en este caso. Es normal haber perdido el patrón observado con anterioridad pues el método considera errores *i.i.d* lo que contradice la naturaleza de los datos. En la Figura 4.15 se ha realizado una vez más el mismo gráfico pero esta vez usando el Wild Bootstrap. En este caso, se observa como el método conserva los patrones de la heterogeneidad de la varianza. Se propone el cálculo de los coeficientes a partir del método wild bootstrap con el siguiente código:

```
fit <- lm(y ~ x + x2)
## obtain 499 samples of the wild bootstrap residuals
res.boot <- wild.boot(fit$res, nboot=499)
## obtain 499 samples of the wild bootstrap responses
y.boot <- matrix(rep(fit$fit,time=499), ncol=499) + res.boot
```

```
coef.wild0<-numeric(499)
coef.wild1<-numeric(499)
coef.wild2<-numeric(499)
for (i in 1:499){
 ajuste.wild<-lm(y.boot[,i]~x+x2)
 coef.wild0[i]<-coef(ajuste.wild)[1]
 coef.wild1[i]<-coef(ajuste.wild)[2]
 coef.wild2[i]<-coef(ajuste.wild)[3]
 }
```

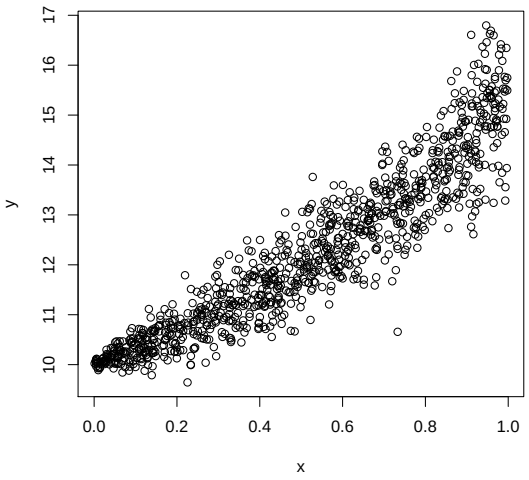


Figura 4.12: *Diagrama de dispersión de los datos simulados. Se entiende la necesidad de realizar un ajuste con x y x^2.*

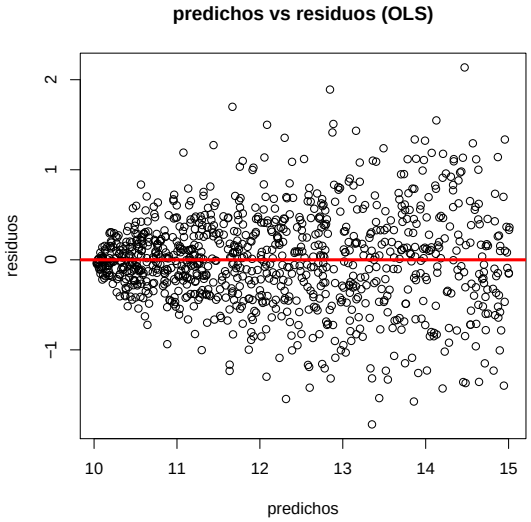


Figura 4.13: *Residuos vs valores predichos en el ajuste por mínimos cuadrados. La heterogeneidad de los errores se hace visible.*

Por otro lado, en la Tabla 4.3 se muestran los desvíos de los coeficientes de regresión multiplicado por 10^2 para el método de remuestreo por residuos, pares y el wild bootstrap. El wild bootstrap y el método de pares coinciden en los resultados mientras que el de residuos exagera notablemente el valor de los desvíos para la ordenada al origen. Este comportamiento era de esperarse en un caso de heteroscedasticidad ya que el método de residuos usa fuertemente los supuestos del modelo mientras que el de pares es más estable frente a la falta de homoscedasticidad. Por otro lado, el wild bootstrap resulta estable bajo heteroscedasticidad ya que supone errores provenientes de distribuciones en principio distintas.

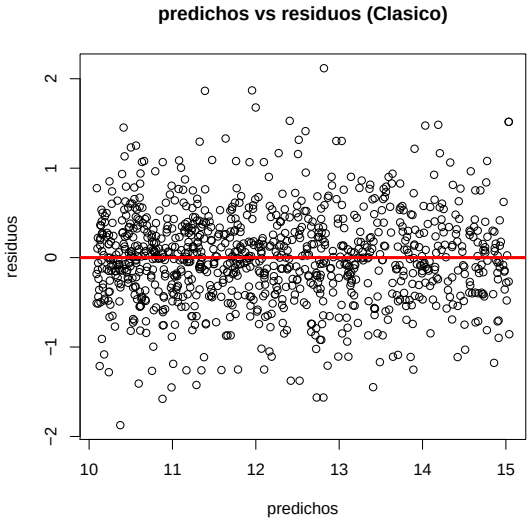


Figura 4.14: *Residuos vs valores predichos en el ajuste por mínimos cuadrados bajo la metodología bootstrap clásica.*

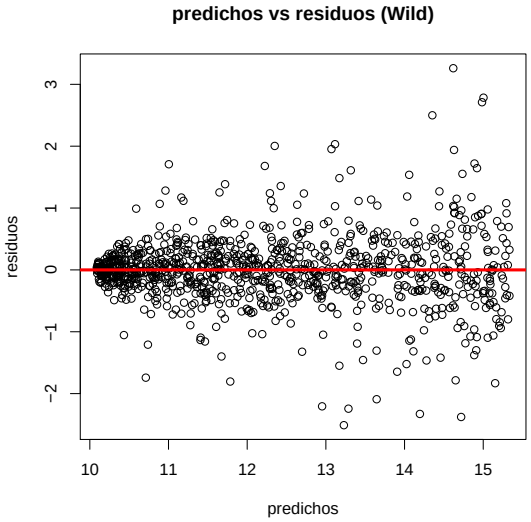


Figura 4.15: *Residuos vs valores predichos en el ajuste por mínimos cuadrados bajo la metodología Wild Bootstrap.*

Método	$\widehat{\beta}_0$	$\widehat{\beta}_1$	$\widehat{\beta}_2$
Residuos	4.96	22.93	21.61
Pares	2.7	19.26	22.21
Wild	2.7	19.5	21.93

Tabla 4.3: *: Desvíos de los coeficientes ($\times 10^2$) de regresión en el ejemplo simulado mediante metodología Wild bootstrap, bootstrap por residuos y por pares.*

Los sesgos y desvíos de la metodología wild bootstrap pueden hallarse directamente a partir del desvío y el sesgo de los coeficientes hallados anteriormente y se describe el programa que permite su cálculo:

```
#sesgos y desvios del metodo wild
sd.0<-sd(coef.wild0)
bias.0<-10-mean(coef.wild0)

sd.1<-sd(coef.wild1)
bias.1<-3-mean(coef.wild1)

sd.2<-sd(coef.wild2)
```

```
bias.2<-2-mean(coef.wild2)
```

Capítulo 5

Bootstrap en problemas de predicción en modelos lineales

En este capítulo, se desarrolla el problema de predicción para modelos lineales y se estudia el uso del Bootstrap en el análisis del error de predicción agregado. En el Capítulo 7, se generaliza el estudio de predicción para el modelo de regresion logística que en el caso de dos poblaciones es análogo al de clasificación.

5.1 Predicción por Validación Cruzada

Predecesora a la metodología bootstrap, la denominada Validación Cruzada y notada CV es una herramienta útil para el cálculo del error de predicción. En modelos de regresión lineal, así como en modelos de regresión logística que se analizarán en el Capítulo 7, reviste especial interés la estimación del error de predicción de un modelo M.

Idealmente se necesitaría una nueva colección de datos con el fin de estimar un error de predicción global. Esa colección habría de ser una muestra de testeo, distinta a la muestra usada para ajustar el modelo. A partir de ella se podrían calcular nuevos residuos y considerar, por ejemplo, el promedio del cuadrado de éstos como estimador del error de predicción. Más precisamente, si $(\widetilde{x}_1, \widetilde{y}_1), ..., (\widetilde{x}_m, \widetilde{y}_m)$ fuese una nueva muestra independiente de la muestra original $(x_1, y_1), ..., (x_n, y_n)$ y $(\widehat{\widetilde{y}}_1, ..., \widehat{\widetilde{y}}_m)$ fuesen los valores predichos de la muestra de testeo se podría definir el error de predicción como:

$$\widehat{EP} = (1/m) \sum_{i=1}^{m} (\widetilde{y}_i - \widehat{\widetilde{y}}_i)^2.$$

Usualmente, la obtención de nuevos datos se dificulta por los costos que esto implica, por los tiempos requeridos o por otros factores limitantes. El método de Validación Cruzada permite obtener un estimador del error de predicción sin usar nuevas muestras. El método utiliza parte de la información disponible para ajustar el modelo y la otra para testearlo. En esta sección, se desarrollará primero el caso *leave-one-out* del método de Validación Cruzada que deja una única observación disponible para testear: para cada observación i, se ajusta el modelo lineal con las $n-1$ observaciones restantes y se considera el cuadrado de la diferencia entre y_i y $\widehat{y}_i^{(-i)}$ que es el valor predicho de la i-ésima observación en el modelo ajustado sin esta observación.

Definición 5.1. Se define la estimación del error de predicción bajo el método de Validación Cruzada leave-one-out por:

$$CV = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i^{(-i)})^2.$$

De hecho, a partir de la expresión de $\widehat{y}_i^{(-i)}$ se puede demostrar la siguiente propiedad.

Proposición 5.1. *Los residuos del método Validación Cruzada son siempre mayores o iguales en valor absoluto a los residuos del modelo original.*

Demostración. Se recuerda que la estimación de β sin la $i-$ésima observación está dada por:

$$\widehat{\beta}_{(-i)} = \widehat{\beta} - \frac{(X^tX)^{-1}X_i r_i}{1-p_{ii}}.$$

Por ende,

$$\widehat{y}_i^{(-i)} = \widehat{y}_i - \frac{p_{ii}(y_i - \widehat{y}_i)}{1-p_{ii}},$$

de donde se deduce que

$$\widehat{y_i}^{-i} - y_i = y_i(1-p_{ii}) + \widehat{y}_i - p_{ii}y_i = \frac{\widehat{y}_i - y_i}{1-p_{ii}}.$$

Recordando que p_{ii} es el $i-$ésimo elemento de la diagonal de la matriz de proyección P y que $0 < p_{ii} < 1$ queda claro el resultado buscado. □

En general, no hay razón para tomar conjuntos de tamaño $n-1$ para ajustar el modelo en el método de Validación Cruzada. Una implementación posible del método por grupos es considerar un subconjunto de ajuste o muestra de entrenamiento C_a y un subconjunto de testeo o de predicción C_p del conjunto original $\{(y_i, x_i)\}_{i=1}^n$ de modo que sean conjuntos disjuntos y que juntos tengan el total de datos originales. El conjunto C_a se utilizaría para

ajustar el modelo por mínimos cuadrados mientras que el conjunto C_p permitiría estimar el error de predicción del mismo. El método *K-fold* de Validación Cruzada propone repetir este procedimiento K veces, con K diferentes particiones C_a y C_p de los datos. Se puede pensar que si n es el número de datos del modelo, entonces n_p es el número de elementos en C_p y por ende, C_a cuenta con $n - n_p$ datos.

Definición 5.2. Se define el método de Validación Cruzada Generalizada por:

$$KCV_g = (1/K)\sum_{k=1}^{K}(1/n_p)\sum_{i\in C_{p,k}}(y_i - X_i^t\widehat{\beta}_{a,k})^2,$$

donde $\widehat{\beta}_{a,k}$ es el estimador de β utilizando mínimos cuadrados sobre el conjunto C_a en la partición k-ésima.

Aún así, en la práctica, se suele utilizar una versión más eficiente de este método en el que se considera $\{C_1, ..., C_K\}$ una partición aleatoria del conjunto de datos original de forma que cada C_k represente un conjunto de predicción y que el conjunto $\cup_{j\neq k}C_j$ represente el conjunto de ajuste en cada caso (ver Davison y Hinkley (1997)). Para cada (x_i, y_i), $i = 1, ..., n$ se analiza el error de predicción en el $i-$ésimo elemento bajo el modelo ajustado con la unión de los C_k que no contengan a la observación $i-$ésima, obteniéndose como estimador del error:

$$KCV_K = (1/n)\sum_{i=1}^{n}(y_i - X_i^t\widehat{\beta}_{-k(i)})^2,$$

con $\widehat{\beta}_{-k(i)}$ el estimador calculado con el subconjunto de datos donde el grupo $C_{k(i)}$ conteniendo la $i-$ésima observación fue eliminado.

Se observa que si $K = n$ se obtiene el método de Validación Cruzada leave-one-out. Por lo general, se suele tomar $K = \min\{n^{1/2}, 10\}$.

Como se verá en el Capítulo 6, el método de Validación Cruzada juega un papel importante en el problema de selección de variables y su metodología es utilizada en los métodos bootstrap aplicados a este problema.

5.2 La estimación bootstrap del error de predicción

El primer enfoque bootstrap, el más simple, genera B muestras bootstrap, calcula sus respectivas replicaciones $\widehat{\beta}_b^*$ y aplica cada modelo ajustado a la muestra original dando lugar a B estimaciones del error de predicción. El promedio de esto último es la estimación

	$err(z^*, \widehat{F})$	$err(z^*, \widehat{F}^*)$	$err(z^*, \widehat{F}) - err(z^*, \widehat{F}^*)$
Promedio :	14105.44	12733.4	818.2533

Tabla 5.1: *En la primera columna se encuentra el promedio de 1000 estimaciones del error de predicción bootstrap. En la segunda columna se ha calculado el promedio de 1000 promedios de la suma de los cuadrados de los residuos de cada modelo ajustado bootstrap. La última columna es la resta de ambos promedios.*

de este enfoque. Como un ejemplo de esto, en la primera columna del la Tabla 5.2 se ha calculado el promedio de 1000 estimaciones bootstrap del error de predicción para los datos del 6MWT del Capítulo 4.2.

El promedio calculado da un valor de 14105.44 en comparación al valor de 13287.19 obtenido por el promedio de la suma de los cuadrados de los residuos del modelo ajustado original (RRS/n = 13287.19). RSS/n puede entenderse como una estimación del error de predicción que tiende a subestimar el verdadero error pues es, de alguna manera, demasiado *optimista* ya que usa la misma información tanto para predecir como para ajustar. En la segunda columna de la Tabla 5.2 se ha calculado el promedio de 1000 predicciones del error cuando el modelo ajustado bootstrap se aplica a la misma muestra bootstrap. No es sorprendente notar que este valor es menor al de la primera columna pues como se dijo, es una estimación que tiende a subestimar el error real. Efron (1993) propone otro método de mayor precisión en la estimación. Llama *optimismo* a la diferencia entre la primera y la segunda columna de la Tabla 5.2 y propone que la estimación, a la que denomina error de predicción Bootstrap mejorado, sea la suma del promedio de la suma de los cuadrados de los residuos del modelo original y el *optimismo* (RSS/n + *optimismo*). En el ejemplo de datos del 6MWT se obtiene 13287.19+818.2533 = 14105.44. Esencialmente se ha agregado una corrección del sesgo a RSS/n, también conocido como la *tasa del error aparente*.

Se tratará de dar un poco más de formalidad y de generalidad en el caso del estudio de predicción en modelos de regresión lineal. Sea $z = \{(x_i, y_i)\}_{i=1}^n$ el conjunto de datos al cual se ajusta un modelo de regresión lineal. Las observaciones x_i pueden ser vectores multi-dimensionales como en el caso de regresión múltiple mientras que los valores y_i son las respuestas teóricas de dichas observaciones. La idea del cálculo del error de predicción nace a partir del interés de saber que tan bien un modelo puede predecir nuevas observaciones.

Definición 5.3. Si (x_0, y_0) es una nueva observación, se define el error de prediccón en esa nueva observación para el modelo de regresión ajustado por el conjunto de datos z por

$$err(z, F) = E_F[c(y_0, \widehat{y}_0)]. \tag{5.1}$$

donde la esperanza se toma sobre la nueva observación a partir de la población F y donde F es la función de distribución a partir de la cual se han generado los datos del modelo. Esto quiere decir que el conjunto de observaciones originales $z = \{x_i, y_i\}_{i=1}^n$ es una muestra i.i.d. a partir de la distribución multi-dimensional F. Además, c es la denominada función de costo que mide la distancia entre el valor observado y el valor predicho.

En modelos de regresión lineal suele utilizarse la función de costo c definida por:

$$c(y, \widehat{y}) = (y - \widehat{y})^2,$$

que es de hecho la función que se utilizó en el análisis del método de Validación Cruzada. En general, el problema de predicción focaliza su análisis en otra medida: el *error de predicción medio* definido por $E_F(err(z, F))$, donde en este caso la esperanza se toma sobre el conjunto de observaciones $(x_i, y_i) \sim F$. Para el caso de regresión lineal, utilizando:

1. la función de costo $c(y, \widehat{y}) = (y - \widehat{y})^2$,
2. la estimación de los coeficientes de regresión por mínimos cuadrados $\widehat{\beta} = (X^t X)^{-1} X^t Y$,

se tiene que (5.1) se escribe como

$$A_{ag} = E\left[(y_0 - x_0^t \widehat{\beta})^2\right].$$

Si se aplica el *error de predicción* a la misma muestra con la que se ha ajustado el modelo de regresión, se obtiene lo que se denomina *tasa del error aparente*,

$$\Delta_{ap} = err(z, \widehat{F}) = (1/n) \sum_{i=1}^{n} (y_i - x_i^t \widehat{\beta})^2 = RSS/n,$$

donde $\widehat{F}$ es la distribución empírica de los datos. Como se dijo, ésta, es una estimación del error de predicción que tiende a subestimar los datos ya que utiliza la misma muestra tanto para ajustar el modelo como para predecir el error. Se destacan dos estimaciones bootstrap del error de predicción. Una de ellas propone el cálculo de una aproximación del sesgo cometido al utilizar el error de predicción aparente para corregir el mismo error de predicción aparente, mientras que la otra es la estimación bootstrap más simple que se detalla a continuación.

Para construir la estimación bootstrap más simple del error de predicción, se aplica el principio plug-in al error de predicción medio.

Definición 5.4. Si $z^* = \{(x_1^*, y_1^*), ..., (x_n^*, y_n^*)\}$ es una muestra bootstrap, se define el error de predicción bootstrap como:

$$E_{\widehat{F}}[err(z^*, \widehat{F})] = E_{\widehat{F}}[\sum_{i=1}^{n} c(y_i, \widehat{y}_i^*)/n]. \tag{5.2}$$

Dado que la última esperanza se toma sobre la distribución ideal bootstrap, en la práctica, se aproxima este valor utilizando una cantidad finita de réplicas bootstrap, obteniéndose la siguiente expresión:

$$\widehat{E}_{\widehat{F}}[err(z^*, \widehat{F})] = \frac{1}{B} \sum_{b=1}^{B} \sum_{i=1}^{n} c(y_i, \widehat{y}_{i,b}^*)/n.$$

El error de predicción bootstrap mejorado propuesto por Efron (1993) (que agrega una estimación del sesgo al error de predicción aparente) propone otro enfoque que ya fue explicado informalmente al principio de esta sección. El autor considera la suma del error aparente con el denominado optimismo. Como ya se ha dicho, el error de predicción aparente viene dado por $err(z, \widehat{F})$ mientras que el sesgo de esta estimación del error (el optimismo) tiene la siguiente expresión:

$$\omega(F) = E_F[err(z, F) - err(z, \widehat{F})],$$

mientras que su estimación bootstrap debe ser entonces,

$$\omega(\widehat{F}) = E_{\widehat{F}}[err(z^*, \widehat{F}) - err(z*, \widehat{F}^*)],$$

donde $\widehat{F}^*$ es la distribución empírica de una muestra bootstrap. Una vez más, en la práctica, se considera una aproximación de esta cantidad ideal, dado que la última considera un número infinito de muestras bootstrap (es la estimación de $\omega(F)$ para conjuntos de tamaño n tomados aleatoriamente de $\widehat{F}$). Utilizando B muestras bootstrap se obtiene:

$$\widehat{\omega}(\widehat{F}) = \frac{1}{Bn} \left\{ \sum_{b=1}^{B} \sum_{i=1}^{n} c(y_i, \widehat{y}_{i,b}^*) - \sum_{b=1}^{B} \sum_{i=1}^{n} c(y_{i,b}^*, \widehat{y}_{i,b}^{**}) \right\},$$

donde $\widehat{y}_{i,b}^*$ es el valor predicho en la $i-$ésima observación original con el modelo ajustado por la $b-$ésima muestra bootstrap, $y_{i,b}^*$ es el valor de la respuesta de la $i-$ésima observación de la $b-$ésima muestra bootstrap y $\widehat{y}_{i,b}^{**}$ es el valor predicho de la $i-$ésima observación de la $b-$ésima muestra bootstrap con el modelo ajustado por la $b-$ésima muestra bootstrap. Finalmente el error de predicción bootstrap mejorado se calcula con la siguiente expresión:

$$err(z, \widehat{F}) + \widehat{\omega}(\widehat{F}).$$

Capítulo 6

Bootstrap en problemas de selección de variables

Se considera en este capítulo, el problema de regresión descripto en el Capítulo 4. Los datos del modelo vienen dados por $z_i = (x_i, y_i)$, donde $x_i = (1, x_{i,1}, ..., x_{i,p-1})^t$ es la variable regresora $i-$ésima e y_i es la respuesta de la observación $i-$ésima. Para estudiar y analizar el modelo lineal se ha hecho uso hasta aquí del método de mínimos cuadrados. Aún así, Tibshirani (1996) subraya dos problemas mayores:

1. La precisión en la predicción no es buena dado que la varianza de los estimadores suele ser grande.
2. Modelos con una gran cantidad de coeficientes de regresión dificultan la interpretación de los resultados.

Para ambos inconvenientes, el problema de selección de variables propone trabajar con un subconjunto de predictores. Se conocen distintos enfoques aunque todos convergen en una misma cuestión: Qué covariables utilizar en el modelo final y cuáles descartar? En este capítulo, se supondrá que el modelo lineal es el modelo correcto para ajustar los datos y se buscará eliminar información de variables redundantes.

Existen diversos métodos para realizar esta tarea que pueden categorizarse en dos grandes grupos: Los métodos *APR* (all possible regression) que recorren todos los candidatos posibles en búsqueda de un óptimo y los métodos de a pasos o *stepwise* que son menos intensivos computacionalmente aunque no garantizan la obtención del modelo óptimo. En las secciones siguientes se describirán algunos de ellos y se verá como usar el método bootstrap en el problema de selección de variables.

6.1 Criterios de selección de variables

Un enfoque posible para la aplicación del criterio de selección de variables es restringir el problema a una única clase de modelos como la clase de modelos lineales con errores normales y luego intentar seleccionar el *mejor* subconjunto de variables entre los 2^p posibles. Este enfoque convierte el problema de selección de variables en un problema de selección de modelo (ver George (2012)).

Para describir los procedimientos de selección, se empezará fijando la notación a usar. Se indica por X_γ a la matriz de diseño para la cual, las q_γ columnas son el subconjunto γ de $x_1, ..., x_p$, $\gamma = 1, ..., 2^p$. Más precisamente, γ es el índice de los subconjuntos de $x_1, ..., x_p$, y q_γ es el número de covariables en el subconjunto γ. El modelo γ es, por ende: $Y = X_\gamma \beta_\gamma + \epsilon$ donde $E(\epsilon) = 0$, $Var(\epsilon) = \sigma^2$ y el estimador de mínimos cuadrados basado en él será:

$$\widehat{\beta}_\gamma = (X_\gamma^t X_\gamma)^{-1} X_\gamma^t Y.$$

La suma del cuadrado de los residuos se indicará por:

$$SS_\gamma = Y^t Y - \widehat{\beta}_\gamma^t X_\gamma^t X_\gamma \widehat{\beta}_\gamma.$$

Un criterio de selección de variables es elegir el modelo que maximiza el criterio de suma de cuadrados penalizado:

$$SS_\gamma / \widehat{\sigma}^2 - F\, q_\gamma, \tag{6.1}$$

donde $\widehat{\sigma}^2$ es el estimador de σ^2 basado en el modelo completo y F es un valor prefijado. En particular, la elección de $F = 2$ se corresponde con el denominado criterio de información de Akaike, de ahora en más AIC. De forma general, el mismo criterio puede considerarse en modelos que no sean lineales con residuos normales. La expresión general para el valor de AIC es $AIC{=}2ln(L) - 2\, q_\gamma$ donde L es la función de verosimilitud y la expresión (6.1) es el valor del criterio en el caso particular considerado en este capítulo. Para modelos con muchos datos o muchas covariables, puede ser interesante usar un valor de F distinto. Por ejemplo, $F = \log\, n$ corresponde al criterio llamado BIC mientras que cuando $F = 2 \log\, p$ se obtiene el criterio denominado de riesgo aumentado. El comando *stepAIC* de R, que se usará en el próximo ejemplo, utiliza el criterio (6.1) con $F = 2$. Aún así es posible elegir otros valores de F.

6.2 Datos quine y uso de AIC en R

Se analiza a continuación un ejemplo en el que se aplica el criterio de Akaike para selección de variables de un modelo lineal. En este ejemplo, se usará el conjunto de datos *quine* de la librería *MASS* de R en el cual 146 niños de Walgett, Australia han sido clasificados según la cultura, la edad, el sexo y el nivel de estudio. A través de estos factores, usados en este ejemplo como covariables, se busca comprender: el número de ausencias al colegio en un año, que será la variable respuesta. Con este fin se ajusta a los datos un modelo de regresión múltiple y se estiman los coeficientes de regresión por mínimos cuadrados. La instrucción summary del ajuste devuelve lo siguiente:

```
attach(quine)
summary(fm1 <- lm( log(Days + 0.5) ~ Eth+Sex+Lrn+Age , data= quine))

Call:
lm(formula = log(Days + 0.5) ~ Eth + Sex + Lrn + Age, data = quine)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5352 -0.6039  0.1418  0.7527  2.1653

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.51776    0.27872   9.033 1.26e-15 ***
EthN        -0.73353    0.18710  -3.921 0.000138 ***
SexM         0.08975    0.19496   0.460 0.645963
LrnSL        0.17285    0.22606   0.765 0.445777
AgeF1       -0.20911    0.29071  -0.719 0.473151
AgeF2        0.18448    0.28901   0.638 0.524319
AgeF3        0.32429    0.30387   1.067 0.287728

Residual standard error: 1.127 on 139 degrees of freedom
Multiple R-squared:  0.1295,Adjusted R-squared:  0.0919
F-statistic: 3.446 on 6 and 139 DF,  p-value: 0.003333
```

Como se puede apreciar, prácticamente la totalidad de las variables independientes parecen no ser informativas respecto de la variable respuesta. El valor de R^2 es muy cercano a 0 y todos los coeficientes de regresión con excepción de la covariable Cultura (Eth) no rechazan el valor 0 en el test de t. En la Figura 6.1 se muestran los intervalos de confianza de nivel 95% para los estimadores de los coeficientes de regresión calculados con

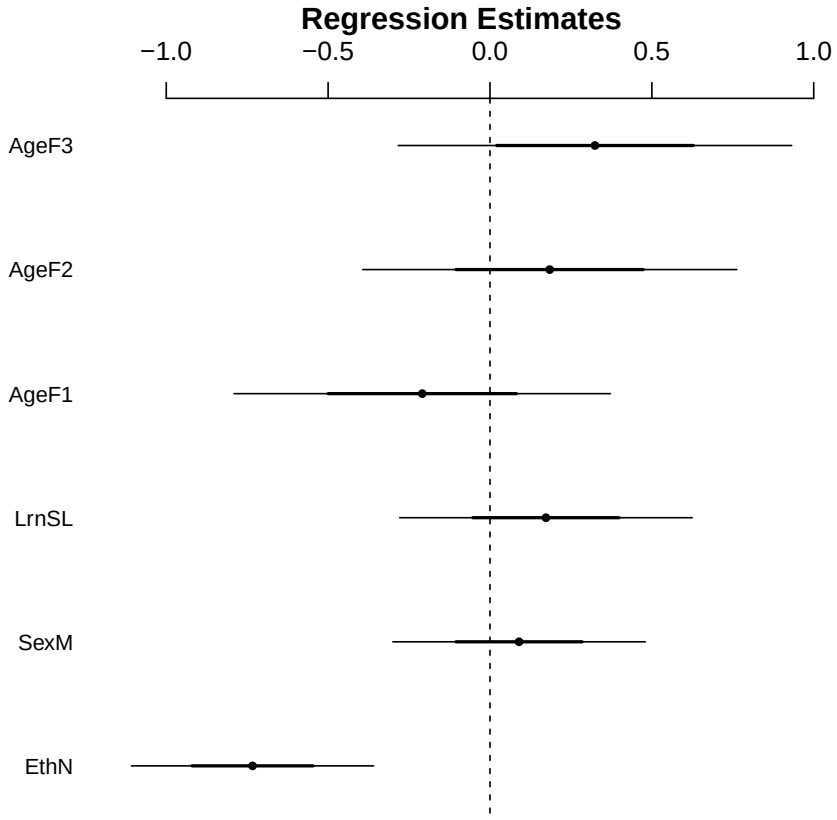


Figura 6.1: *Intervalos de confianza de nivel 95% usando el comando coefplot de R.*

el comando *coefplot* de la librería *arm* de R en los que se puede apreciar lo recientemente enunciado. Por otro lado, los supuestos del modelo lineal parecen ajustarse razonablemente (ver Figura 6.2). La verificación de estos supuestos reviste especial interés si se tienen en cuenta medidas de bondad como R^2 que usan fuertemente que los errores son normales y homoscedásticos. Se usará entonces el criterio de Akaike con el fin de simplificar el modelo originalmente propuesto. El comando *stepAIC* de R, que por defecto utiliza el método *backward stepwise selection*, simplifica la tarea. El método backward es un método iterativo que comienza con todos los candidatos a predictores del modelo y elimina covariables hasta obtener un conjunto lo más significativo posible respecto a la respuesta. El criterio para eliminar variables es justamente el valor de AIC en este caso. Se eliminan las variables con menor AIC. Las instrucciones y resultados obtenidos se exhiben a continuación:

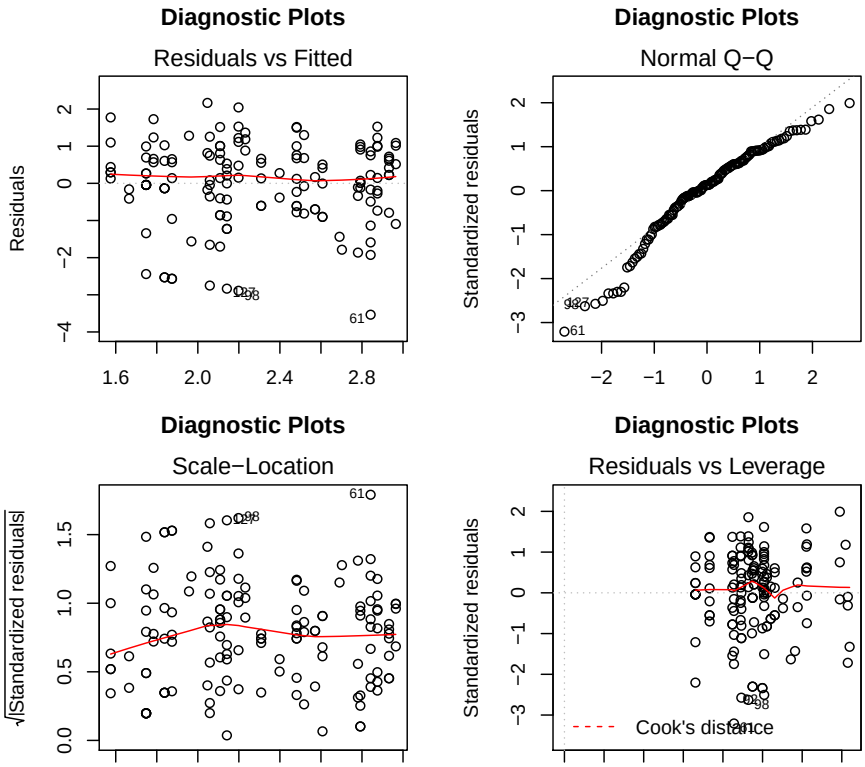


Figura 6.2: *Diagnostico del ajuste de regresión múltiple de los datos quine.*

```
fm2 <- stepAIC(fm1, trace = F)
summary(fm2)

Call:
lm(formula = log(Days + 0.5) ~ Eth, data = quine)

Residuals:
    Min      1Q  Median      3Q     Max
-3.3919 -0.6941  0.0647  0.8293  2.2911

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.6988     0.1355  19.921  < 2e-16 ***
EthN         -0.7485     0.1865  -4.013 9.62e-05 ***

Residual standard error: 1.125 on 144 degrees of freedom
Multiple R-squared:  0.1006,Adjusted R-squared:  0.09432
F-statistic:  16.1 on 1 and 144 DF,  p-value: 9.616e-05
```

El resultado era quizás anticipable: el criterio optó por eliminar del modelo los coeficientes de todas las variables que no eran significativas en la instancia anterior y ajustó el valor del estimador del coeficiente de la covariable restante.

6.3 Validación cruzada en selección de variables

Respetando el mismo objetivo, es decir, eliminar información redundante de un modelo de regresión lineal múltiple, resulta de utilidad el método de Validación Cruzada que se describirá en esta sección. Por simplicidad, se considerarán modelos ajustados por mínimos cuadrados y errores de predicción obtenidos por errores cuadráticos medios, supuestos que ya se han usado en el Capítulo 5.

A diferencia del método AIC, éste no pretende penalizar el ajuste de mínimos cuadrados. La propuesta consiste en estimar errores de predicción, como se realizó en el Capítulo 5, en todos los posibles modelos y optar por el de menor error estimado. Se fijará la notación a utilizar. El objetivo es elegir un modelo M entre los 2^p posibles (por las formas de quitar o dejar alguno de los coeficientes). Si el modelo M es usado se denotará por X_M a la matriz de diseño asociada, es decir, a la matriz con p_M covariables y se llamará $\widehat{\beta}_M$ y P_M al estimador de los coeficientes de regresión por mínimos cuadrados bajo el modelo M y

a la matriz de proyección en el subespacio de las columnas de X_M respectivamente. Se indica por $\widehat{y}_M$ a $P_M y$. Se ha decidido cambiar la notación inicalmente propuesta en este capítulo para facilitar la explicación del método.

Davison y Hinkley (1997), proponen el uso del método Validación Cruzada Generalizada. Más precisamente, para el análisis de selección de variables, los autores proponen el uso del siguiente estimador del error en el modelo M:

$$CV(M) = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_p}\sum_{i\in C_{p,k}}(y_i - X_{Mi}^t\widehat{\beta}_M(C_{a,k}))^2,$$

con $\widehat{\beta}_M(C_{a,k})$ el estimador de mínimos cuadrados en el modelo M respecto del $k-$ésimo conjunto de ajuste de K particiones previas, es decir, el estimador basado en el conjunto $C_{a,k}$ de observaciones y n_p la cantidad de observaciones en el conjunto $C_{p,k}$. Como en el Capítulo 5, se consideran K particiones en conjuntos $\{(C_{a,j}, C_{p,j})\}_{j=1}^K$ para ajustar y para predecir el modelo. Se puede ver que si se elige n_p de forma tal que $n - n_p \to \infty$ y $n_p/n \to 1$ cuando $n \to \infty$ entonces minimizar $CV(M)$ genera una elección consistente del verdadero modelo cuando $n \to \infty$ y $K \to \infty$. Cuando $C_{p,j} = \{(y_j, x_j)\}$ y $C_{a,j} = \{(y_i, x_i) : i \neq j\}$ se obtiene el método de Validación Cruzada leave-one-out que se utilizará en la Sección 6.5.

6.4 Método Bootstrap

Un enfoque posible sería considerar el método bootstrap mejorado propuesto por Efron (1993) en predicción, aplicarlo a cada modelo y optar por el de menor error. Esto no es más que considerar el modelo M que minimize la siguiente expresión:

$$\Delta_B(M) = \frac{1}{n}RSS_M + (1/B)\sum_{b=1}^{B}\frac{1}{n}\left(\sum_{i=1}^{n}(y_i - x_{M,i,b}^{*t}\widehat{\beta}_{M,b}^*)^2 - RSS_{M,b}^*\right),$$

donde el segundo término de la expresión de la derecha es la estimación bootstrap del *optimismo*. El método de re-muestreo puede ser cualquiera de los dos estudiados. Aún así, la minimización de Δ_B no conduce a una elección consistente del modelo real cuando $n \to \infty$. Davison y Hinkley (1997), proponen, para ajustar el modelo, de forma similar a lo propuesto en Validación Cruzada Generalizada, tomar subconjuntos de tamaño $n - m$ en vez de n de forma que $m/n \to 1$ y $n - m \to \infty$ cuando $n \to \infty$. Además, gracias a esta modificación es posible utilizar el siguiente estimador más simple del error de predicción:

$$\Delta_B(M) = \frac{1}{B}\sum_{i=1}^{B}\frac{1}{n}\sum_{i=1}^{n}(y_i - x_{M,i}^t\widehat{\beta}_{M,b}^*)^2.$$

Si el método por pares es usado para remuestrear, $n - m$ datos son tomados de forma aleatoria del conjunto total. Si, por otro lado, el método por residuos es usado, X_M^* es una selección aleatoria de $n-m$ filas de X_M y los $n-m$ residuos r_i^* son tomados aleatoriamente de los n residuos estandarizados y centrados del modelo M.

6.5 Aplicación del método de Validación Cruzada y Bootstrap en un problema con datos de una central nuclear

En este ejemplo, se estudia el comportamiento de los métodos de Validación Cruzada y Bootstrap en selección de variables en un ejemplo de datos reales aportados por una central nuclear. Los datos están disponibles en la librería boot de R con el comando nuclear. Se considera el costo (en millones de dólares) de 32 reactores de agua ligera así como 10 covariables (4 continuas y 6 discretas) asociadas a cada dato. Las covariables continuas son: *date* que representa la fecha en que se emitió el permiso de construcción (medido en años desde el primero de enero de 1990 y redondeado al mes más próximo), $t1$, el tiempo entre la aplicación para el permiso de construcción y la aceptación del mismo, $t2$, el tiempo entre la emisión de la licencia operativa y el permiso de construcción y *cap*, la capacidad neta del poder de la central (MWe). Las variables discretas son: *pr*, una variable binaria para la cual el valor 1 indica la pre-existencia de una central LWR en el mismo sitio, *ne*, una variable binaria para la cual el valor 1 indica que la central fue construída en la región noreste de Estados Unidos, *ct*, una variable binaria para la cual el valor 1 indica el uso de una torre de refrigeración en la central, *bw*, una variable binaria para la cual el valor 1 indica que el sistema de suministro de vapor nuclear fue construido en Babcock-Wilcox, *cum.n*, el número de centrales construidas por cada arquitecto, *pt*, una variable binaria para la cual el valor 1 indica aquellas centrales con garantías parciales.

Se tiene a continuación el encabezado de los datos:

```
head(nuclear)
    cost  date t1 t2  cap pr ne ct bw cum.n pt
1 460.05 68.58 14 46  687  0  1  0  0    14  0
2 452.99 67.33 10 73 1065  0  0  1  0     1  0
3 443.22 67.33 10 85 1065  1  0  1  0     1  0
4 652.32 68.00 11 67 1065  0  1  1  0    12  0
5 642.23 68.00 11 78 1065  1  1  1  0    12  0
6 345.39 67.92 13 51  514  0  1  1  0     3  0
```

Suponiendo que los errores del modelo son normales, es razonable el estudio de los $p-$valores del test de t respecto de cada coeficiente de regresión en el ajuste por mínimos cuadrados. Pocas covariables son realmente significativas en el modelo total y el gran número de posibilidades ($2^{10} = 1024$) en el armado del modelo obligan el uso de algún método de selección de variables. Los siguientes son los comandos y resultados para el análisis de estos datos suponiendo que el modelo lineal es válido.

```
ajuste<-lm(log(cost)~log(cap)+log(cum.n)+pt+pr+ne+ct+
                         bw+t1+t2+date,data=nuclear)
summary(ajuste)

Call:
lm(formula = log(cost) ~ log(cap) + log(cum.n) + pt + pr + ne +
    ct + bw + t1 + t2 + date, data = nuclear)

Residuals:
     Min       1Q   Median       3Q      Max
-0.29493 -0.10322  0.02751  0.09738  0.25297

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -13.356981   5.197581  -2.570  0.01785 *
log(cap)      0.693986   0.136313   5.091 4.84e-05 ***
log(cum.n)   -0.081716   0.046680  -1.751  0.09462 .
pt           -0.228019   0.126595  -1.801  0.08605 .
pr           -0.091766   0.082211  -1.116  0.27693
ne            0.256140   0.077732   3.295  0.00345 **
ct            0.117225   0.067336   1.741  0.09633 .
bw            0.034221   0.104354   0.328  0.74621
t1            0.005009   0.021948   0.228  0.82169
t2            0.004220   0.004700   0.898  0.37942
date          0.212140   0.079716   2.661  0.01461 *

Residual standard error: 0.1658 on 21 degrees of freedom
Multiple R-squared:  0.8696,Adjusted R-squared:  0.8075
F-statistic: 14.01 on 10 and 21 DF,  p-value: 3.622e-07
```

En la Figura 6.3 se grafican los valores del error de predicción utilizando el método de Validación Cruzada leave-one-out (azul) y el Bootstrap (verde, B=100) con $m = 0$ respecto de la cantidad de variables en el modelo. Se ha utilizado el método de pares en las replicaciones bootstrap por la naturaleza de los datos. Además, en rojo se observan algunos errores calculados por el método bootstrap con $n - m = 16$. Como ilustración, los datos han sido introducidos según la siguiente secuencia: date, log(capacity), NE, CT, log(N), PT, T1, T2, PR, BW. El método de Validación Cruzada leave-one-out puede calcularse utilizando el código de R que aquí se presenta:

```
loocv<-function(fit){
+ h=lm.influence(fit)$h
+ mean((residuals(fit)/(1-h))^2)
+ }
> loocv(ajuste)
[1] 0.04454992
```

La misma función debe aplicarse a cada uno de los 6 modelos propuestos. El código para el bootstrap se dará más adelante. En ambos casos, el mínimo error de predicción se alcanza en el modelo con 6 covariables. Se puede ver que el método de Validación Cruzada leave-one-out aplicado a todos los posibles subconjuntos del modelo originalmente planteado conduce a la misma elección. No es así para el método bootstrap aplicado a todos los posibles valores. Las Figuras 6.4 y 6.5 representan los valores del error de predicción bajo ambos métodos respecto de todos los modelos posibles. Una dificultad mayor en la aplicación del método bootstrap en este caso es la elección del cociente m/n. Davison y Hinkley (1997) proponen, en base a evidencia empírica, tomar el valor 2/3 para este cociente. Siendo así, varios de los B subconjuntos utilizados pueden generar matrices de diseño X_M^* singulares. En el ejemplo particular estudiado aquí, modelos con más de 5 covariables no son identificables si se usan subconjuntos de ajuste de tamaño 20 o menos. La naturaleza desbalanceada de las covariables, en conjunción con la naturaleza binaria de alguna de ellas, propician frecuentemente diseños singulares. Esto es lo que ocurre con la estimación del error bootstrap con subconjuntos de ajuste de tamaño 16. En el gráfico se han calculado sólo algunos valores pues los demás conducen de forma regular a matrices singulares.

El cálculo de los errores de predicción Bootstrap y Validación Cruzada para todos los posibles modelos puede ser complicado ya que deben evaluarse 1024 modelos distintos y no hay forma obvia de uso del comando for para lograr esto. Se presenta aquí una técnica con uso del código binario para lograr este fin. Con la codificación binaria, se entiende el uso de una u otra variables en el modelo y para cada caso se calcula tanto el error bootstrap como el error del método Validación Cruzada.

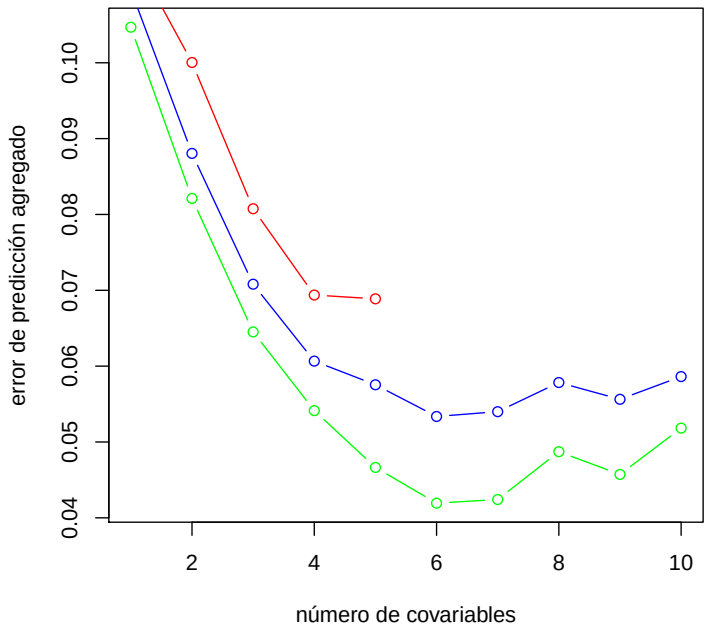


Figura 6.3: *Estimación de los errores de predicción por Validación cruzada(azul) y bootstrap(verde) con* $m = 0$ *respecto del número de covariables en el modelo. En rojo se tienen 5 errores de predicción calculados con el método bootstrap con* $m = 16$.

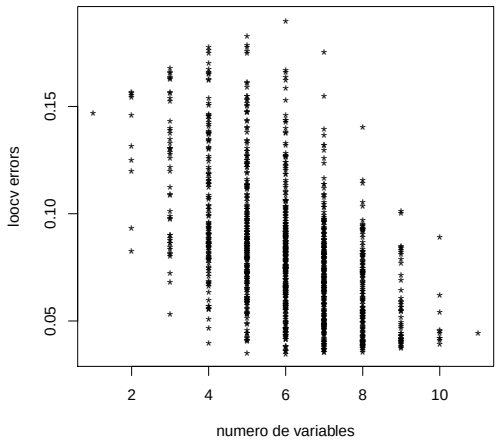


Figura 6.4: *Error de predicción con el método Validación Cruzada leave-one-out para todos los posibles modelos según el número de variables.*

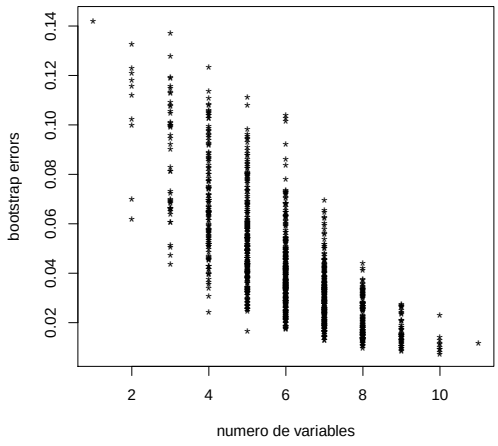


Figura 6.5: *Error de predicción Bootstrap con m=8 y B=100 para todos los posibles modelos según el número de variables.*

```
todo<-matrix(NA,1024,14)
n<-nrow(nuclear)
m<-8
Nboot<-10
for (x in 0:1023){
  binario<-as.integer(intToBits(x))[1:10]
  s1<-ifelse(binario[1]==1,"+date","")
  s2<-ifelse(binario[2]==1,"+log(cap)","")
  s3<-ifelse(binario[3]==1,"+ne","")
  s4<-ifelse(binario[4]==1,"+ct","")
  s5<-ifelse(binario[5]==1,"+log(cum.n)","")
  s6<-ifelse(binario[6]==1,"+pt","")
  s7<-ifelse(binario[7]==1,"+t1","")
  s8<-ifelse(binario[8]==1,"+t2","")
  s9<-ifelse(binario[9]==1,"+pr","")
  s10<-ifelse(binario[10]==1,"+bw","")

  X<-rep(1,n)
  X<-cbind(X)
  s11<-if(binario[1]==1) X<-cbind(X,date)
```

```
  s22<-if(binario[2]==1) X<-cbind(X,log(cap))
  s33<-if(binario[3]==1) X<-cbind(X,ne)
  s44<-if(binario[4]==1) X<-cbind(X,ct)
  s55<-if(binario[5]==1) X<-cbind(X,log(cum.n))
  s66<-if(binario[6]==1) X<-cbind(X,pt)
  s77<-if(binario[7]==1) X<-cbind(X,t1)
  s88<-if(binario[8]==1) X<-cbind(X,t2)
  s99<-if(binario[9]==1) X<-cbind(X,pr)
  s1010<-if(binario[10]==1) X<-cbind(X,bw)


  laformula<-paste("log(cost)~1",s1,s2,s3,s4,s5,s6,
                   s7,s8,s9,s10,sep="")
  ajuste<-lm(laformula,data=nuclear)
  r<-numeric(Nboot)
  rr<-numeric(Nboot)
  for (j in 1:Nboot){
    muestra<-sample(1:n,n-m,replace=TRUE)
    X2<-X[muestra,]
    datos.m<-nuclear[muestra,]
    laformula.boot<-paste("log(cost)~1",s1,s2,s3,s4,s5,s6,
                          s7,s8,s9,s10,sep="")
    ajuste.boot<-lm(laformula.boot,data=datos.m)
    r[j]<-mean(ajuste.boot$res^2,na.rm=TRUE)
    ifelse (length(X2)==n-m,
            rr[j]<-mean((log(cost)[muestra]-X2*ajuste.boot$coef)^2,
                        na.rm=TRUE),
            rr[j]<-mean((log(cost)[muestra]-X2%*%ajuste.boot$coef)^2,
                        na.rm=TRUE))

    }
  todo[x+1,]<-c(sum(binario)+1,binario,loocv(ajuste),
                mean(ajuste$res^2,na.rm=TRUE)-mean(rr-r,na.rm=TRUE),
                mean(rr,na.rm=TRUE))
}
```

El código puede parecer complicado pero es de aplicación directa. La matriz *todo* guarda la información de qué variables se han usado en el modelo respectivo en codificación binaria entre la segunda columna y la columna 11, mientras que en la primera se indica la cantidad de variables usadas. Las últimas columnas tienen el valor del error por Validación Cruzada y el error Bootstrap. Esta información se detalla en la última parte del código siguiente:

```
todo[x+1,]<-c(sum(binario)+1,binario,loocv(ajuste),
                mean(ajuste$res^2,na.rm=TRUE)-mean(rr-r,na.rm=TRUE),
                mean(rr,na.rm=TRUE))
```

Capítulo 7

Bootstrap en regresión logística

7.1 Introducción

El análisis de regresión ha sido el eje central de lo descripto hasta el momento. Aún así se ha limitado el análisis a regresiones lineales, un caso particular, aunque de gran importancia, en los posibles tipos de regresión. Es común, sobre todo en el ámbito clínico, que la regresión de interés considere una variable dependiente respuesta discreta, tomando dos o más posibles valores.

Los modelos lineales que se han desarrollado en los capítulos previos consideran respuestas continuas, tales como el costo de cierto material o la concentración de alguna solución. Los *modelos lineales generalizados* pretenden generalizar situaciones del, a veces simplista, modelo lineal a contextos menos claros. En el mundo de la salud es de vital importancia, en ciertos casos, poder predecir por ejemplo la presencia o la no presencia de una cierta infección en un sujeto. Este es el caso de variables respuestas binarias. En esta situación se opta especialmente por la *regresión logística*. Hosmer y Lemeshow (1989) consideran dos razones primarias en la elección de la regresión logística en casos de variables dependientes dicotómicas:

1. Desde un punto de vista matemático, la función de distribución utilizada es extremadamente flexible y simple.
2. La elección conduce naturalmente a una buena interpretación clínica.

En cualquier problema de regresión la cantidad clave es la esperanza de la variable respuesta condicionada al valor de las covariables independientes. En el caso de regresión

lineal esta cantidad, $E(Y|X)$, que se denotará μ es lineal en X. Se recuerda que $E(Y|X) = X^t\beta$ permite a la esperanza tomar cualquier valor mientras el rango de X varíe de $-\infty$ a ∞. En el caso de variables respuesta dicotómicas, la esperanza condicional debe tomar valores en el intervalo $[0, 1]$. Para variables binarias tomando valores en $\{0, 1\}$ la esperanza condicional es igual a $P(Y = 1|X = x)$. Por esta razón, se relaciona la esperanza $E(Y|X)$ con $X\beta$ a través de una función, llamada función de vínculo que en el caso de regresión logística es

$$h(t) = \frac{e^t}{1 + e^t}.$$

De esta forma,

$$\mu(x) = \frac{e^{\beta_0+\beta_1 x_1+...+\beta_{p-1}x_{p-1}}}{1 + e^{\beta_0+\beta_1 x_1+...+\beta_{p-1}x_{p-1}}}. \tag{7.1}$$

Por otro lado, una transformación de $\mu(x)$ central en el estudio de este tipo de regresión es la transformación *logit* que se define como:

$$g(x) = \log\left(\frac{\mu(x)}{1 - \mu(x)}\right) = \beta_0 + \beta_1 x_1 + ... + \beta_{p-1}x_{p-1}.$$

Se observa que $g = h^{-1}$. En modelos lineales generalizados se conoce a esta función como *función de enlace* y ocupa un lugar entre muchas funciones posibles que relacionan a la variable dependiente con el *predictor lineal* $\eta = \beta_0 + \beta_1 x_1 + ... + \beta_{p-1}x_{p-1}$. En el caso de modelos lineales, la función de enlace es la función identidad. La gran ventaja de esta transformación es que, a diferencia de μ, $g(\mu)$ es lineal en los parámetros y su rango de valores puede abarcar todo dependiendo del rango de valores de x. Se puede escribir también:

$$E(Y|X) = \mu(X) = g^{-1}(X^t\beta),$$

donde se hace abuso de notación y cada elemento en las igualdades es un vector.

Otra observación importante en relación a las diferencias con la regresión lineal tiene que ver con la distribución condicional de la respuesta. La teoría clásica suponía normalidad en el caso estudiado previamente mientras que aquí se supone distribución binomial con probabilidad condicional $\mu(x)$ en la variable respuesta.

Odds ratio

Con el fin de interpretar resultados en el análisis de regresión logística se introduce una medida de asociación conocida como *odds ratio* o razón de chances. Los posibles valores de las probabilidades logísticas en un caso univariado con variable independiente y variable

dependiente dicotómicas tomando valores 0 o 1 pueden ser representadas en una tabla de 2×2 como se muestra en la Tabla 7.1.

Variable respuesta (Y)	Variable independiente (X)	
	$x = 1$	$x = 0$
$y = 1$	$\pi(1) = \dfrac{\exp(\beta_0 + \beta_1)}{1 + \exp(\beta_0 + \beta_1)}$	$\pi(0) = \dfrac{\exp \beta_0}{1 + \exp \beta_0}$
$y = 0$	$1 - \pi(1) = \dfrac{1}{1 + \exp(\beta_0 + \beta_1)}$	$1 - \pi(0) = \dfrac{1}{1 + \exp \beta_0}$
Total	1.0	1.0

Tabla 7.1: *Probabilidades logísticas en un caso univariado con variable independiente y variable respuesta dicotómicas.*

Se puede pensar en un ejemplo en el que la variable respuesta Y discrimine sobrevivientes y no sobrevivientes. De la misma forma, se puede considerar que la variable independiente determine el sexo del paciente. Por ejemplo, el par $(x = 1, y = 1)$ querrá decir que un paciente de sexo femenino ha fallecido. Las chances (odds) de que una persona no haya sobrevivido entre las que son mujeres se define como $\pi(1)/(1 - \pi(1))$.

Definición 7.1. La razón de chances u odds ratio, notada OR, se define como la razón de las chances para $x = 1$ sobre las chances para $x = 0$ y viene dada por la ecuación:

$$OR = \frac{\pi(1)/\{1 - \pi(1)\}}{\pi(0)/\{1 - \pi(0)\}}.$$

Se puede ver fácilmente, reemplazando los valores de $\pi(1)$ por $e^{\beta_0+\beta_1}/(1+e^{\beta_0+\beta_1})$ y $\pi(0)$ por $e^{\beta_0}/(1+e^{\beta_0})$ que $OR = e^{\beta_1}$. Esto puede generalizarse naturalmente a regresiones múltiples donde el OR respecto de una cierta covariable ha de ser la exponencial del coeficiente de regresión asociado. Además, para el caso univariado, el cálculo de la razón de chances puede aproximarse directamente mediante la razón del producto cruzado de una tabla de contingencia sin necesidad de pasar por el cálculo de un estimador de máxima verosimilitud.

De forma general, puede pensarse que si se tienen dos variables aleatorias independientes $Y_1 \sim B(n, p)$ e $Y_2 \sim B(m, q)$, donde $0 < p,\ q < 1$, la razón de chances queda definida por

$$OR = \frac{p/(1-p)}{q/(1-q)},$$

y puede estimarse por

$$\widehat{OR} = \frac{\widehat{p}/(1-\widehat{p})}{\widehat{q}/(1-\widehat{q})},$$

donde

$$\widehat{p} = Y_1/n = p + \delta_n = p + \sqrt{p(1-p)}\psi_n/\sqrt{n},$$

y

$$\widehat{q} = Y_2/m = q + \Delta_m = q + \sqrt{q(1-q)}\Psi_m/\sqrt{m},$$

con ψ_n y Ψ_m asintóticamente normales. Es importante que n y m sean valores grandes.

Proposición 7.1. *Si se cumplen las hipótesis mencionadas,* $\log(\widehat{OR})$ *es asintóticamente normal con media* $\log(OR)$ *y varianza asintótica*

$$\frac{1}{np} + \frac{1}{n(1-p)} + \frac{1}{mq} + \frac{1}{m(1-q)}. \tag{7.2}$$

Demostración.

$$\log(\widehat{OR}) - \log(OR) = \log(\widehat{p}/p) - \log(1-\widehat{p}/(1-p)) - \log(\widehat{q}/q) + \log(1-\widehat{q}/(1-q))$$

$$= \log\left(1+\frac{\delta_n}{p}\right) - \log\left(1+\frac{\delta_n}{1-p}\right) - \log\left(1+\frac{\Delta_m}{q}\right) + \log\left(1+\frac{\Delta_m}{1-q}\right).$$

Lo cual, aproximado por un desarrollo de taylor de orden 1 es igual a

$$\simeq \left(\frac{1}{p}+\frac{1}{1-p}\right)\delta_n - \left(\frac{1}{q}+\frac{1}{1-q}\right)\Delta_m$$

$$= \frac{1}{\sqrt{p\,(1-p)}}\frac{\psi_n}{\sqrt{n}} - \frac{1}{\sqrt{q\,(1-q)}}\frac{\Psi_m}{\sqrt{m}}.$$

La varianza de esto último es (7.2) y suele aproximarse por

$$\frac{1}{Y_1} + \frac{1}{n-Y_1} + \frac{1}{Y_2} + \frac{1}{m-Y_2}.$$

□

7.2 Ejemplo en R de uso de la regresión logística

Se presenta un breve ejemplo de regresión logística para que el lector pueda familiarizarse con la misma.

Se consideran 3 coeficientes de regresión, b_0, b_1, b_2 fijos y dos covariables x_1, x_2 simuladas a partir de 200 normales y 200 binomiales, respectivamente. A partir de la función $b_0 + b_1x_1 + b_2x_2$ se generan las respuestas binarias y. La inversa de la función de enlace *invlogit* permite establecer la probabilidad de éxito de las respuestas:

```
library(arm)
# Parametros
b0 <- 1
b1 <- 2.5
b2 <- 2
# variables de regresion
x1 <- rnorm(200)
x2 <- rbinom(200, 1, 0.5)
# respuesta binaria como funcion de "b0 + b1 * x1 + b2 * x2"
y <- rbinom(200, 1,
            invlogit(b0 + b1 * x1 + b2 * x2))
```

Una forma de visualizar la información es a través del siguiente código. El resultado puede apreciarse en la Figura 7.1 donde se grafican las observaciones obtenidas (x_{1i}, y_i) así como dos curvas que representan en rojo y azul las curvas de regresión logística para el modelo con el coeficiente b_2 y sin el coeficiente, respectivamente.

```
# Grafico y visualizacion de los datos
jitter.binary <- function(a, jitt = 0.05)
{
ifelse(a==0, runif(length(a), 0, jitt),
       runif(length(a), 1-jitt, 1))
}
plot(x1, jitter.binary(y), xlab = "x1",
     ylab = "Success probability")
curve(invlogit(b0 + b1*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2)
curve(invlogit(b0 + b1*x + b2),
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2)
legend("bottomright", c("b2 = 0", "b2 = 2"),
       col = c("blue", "red"), lwd = 2, lty = 1)
```

Una vez visualizados los datos, se realiza el ajuste del modelo logístico con la función *glm* de R y se muestra el resultado del ajuste:

```
> fn <- glm(y ~ x1 + x2, family = binomial())
> summary(fn)

Call:
glm(formula = y ~ x1 + x2, family = binomial())
```

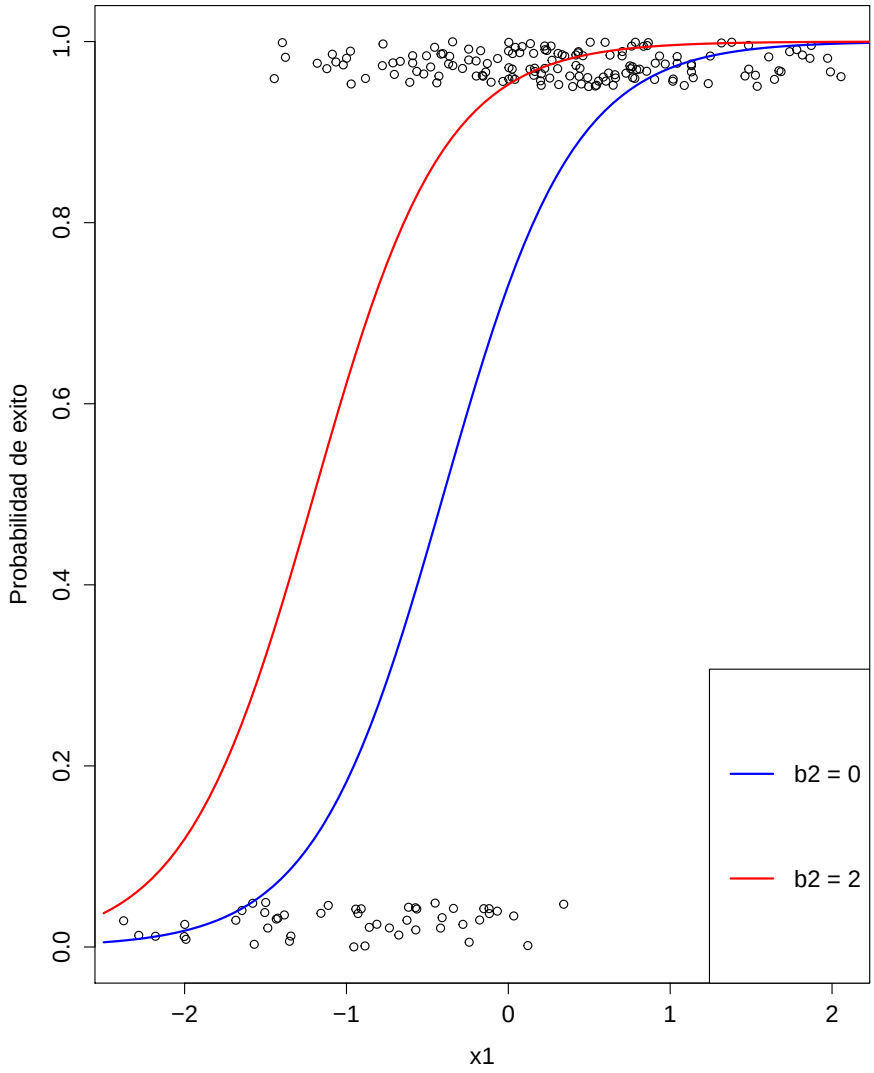


Figura 7.1: *Gráfico de la variables respuesta contra los valores de la variable de regresión x_1. Los puntos están ligeramente corridos para una mejor visualización de los datos. La curva azul representa la curva de regresión logística para el modelo sin el coeficiente b_2 mientras que la curva roja es la misma curva de regresión logística con el parámetro b_2 incluido.*

```
Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.73406  -0.09251   0.12030   0.43560   2.26211

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.0844     0.3205   3.383 0.000716 ***
x1            3.0260     0.5028   6.019 1.76e-09 ***
x2            2.4946     0.5716   4.364 1.27e-05 ***

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 233.30  on 199  degrees of freedom
Residual deviance: 112.54  on 197  degrees of freedom
AIC: 118.54

Number of Fisher Scoring iterations: 6
```

Para visualizar el resultado del ajuste, se realiza una vez más un gráfico como el de la Figura 7.1 donde se superponen en punteado las respectivas curvas logísticas ajustadas por el modelo propuesto (Figura 7.2). Las líneas representan las verdaderas curvas logísticas del modelo. Las siguientes instrucciones permiten obtener la Figura 7.2.

```
# Coeficientes y visualizacion

plot(x1, jitter.binary(y), xlab = "x1",
     ylab = "Success probability")
beta <- coef(fn)
b0.hat <- beta[1]
b1.hat <- beta[2]
b2.hat <- beta[3]
curve(invlogit(b0 + b1*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2)
curve(invlogit(b0.hat + b1.hat*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2,
      lty =2)
curve(invlogit(b0 + b1*x + b2),
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2)
curve(invlogit(b0.hat + b1.hat*x + b2.hat),
```

```
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2,
      lty =2)
legend("bottomright", c("b2 = 0", "b2 = 2"),
       col = c("blue", "red"), lwd = 2, lty = 1)
```

7.3 Métodos de remuestreo

Al igual que con modelos lineales, no hay una única técnica de remuestreo bootstrap con el fin de obtener intervalos de confianza para el análisis de los coeficientes de regresión ajustados.

Con el fin de abarcar estos procesos en el caso específico de regresión logística se usará cierta notación general propuesta por Davison y Hinkley (1997). Estos autores proponen pensar los modelos lineales generalizados bajo el siguiente modelo:

$$E(y_i) = \mu_i, \quad g(\mu_i) = x_i^t\beta, \quad Var(y_i) = kc_iV(\mu_i), \tag{7.3}$$

donde k puede ser desconocido, c_i son pesos conocidos, g es la función de enlace, que en el caso de regresión logística es la función logit, y μ_j como en (7.1) es la media condicional de $y_i|x_i$. Por otro lado, $V(.)$ ha de ser una función de varianza conocida. Para el caso de respuestas binomiales, si se considera $r_i|x_i \sim B(m_i, p(x_i))$ y se define $y_i = r_i/m_i$ (que es lo que suele realizarse en el caso de regresión logística) para cada i, se obtiene para y_i, $k = 1$ y c_i la cantidad de repeticiones en las binomiales (ver Sección 7.4). Aún así se mantendrá la notación completa para mayor generalidad de los métodos propuestos. En un próximo ejemplo se analizarán más en detalle estos parámetros.

En lo que sigue se estudiará la forma de de definir residuos, la forma de ajustar modelos y la definición de la medida estadística *desviación* antes de pasar propiamente a los modelos de remuestreo.

Residuos

La manera más simple de encarar la definición de residuos en este caso es imitando la definición usada en modelos lineales si bien la estructura del modelo ya no es la misma.

Definición 7.2. Se definen los *residuos Pearson* como $(y_j - \widehat{\mu}_j)/\{c_j\widehat{k}V(\widehat{\mu}_j)\}^{1/2}$. El mismo ajuste leverage se puede hacerse aquí de forma a obtener los *residuos Pearson estandarizados*, es decir:

$$r_{Pi} = \frac{y_i - \widehat{\mu}_i}{\{c_i\widehat{k}V(\widehat{\mu}_i)(1-h_i)\}^{1/2}},$$

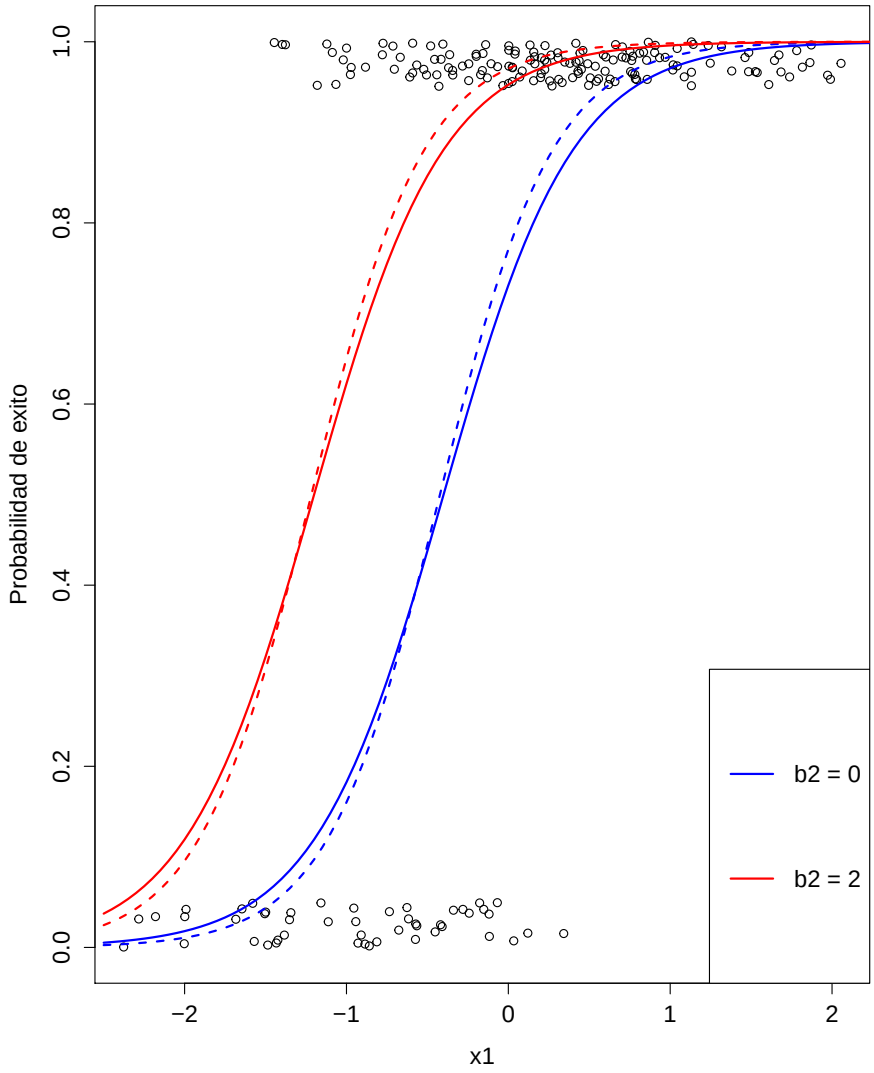


Figura 7.2: *Gráfico de la variables respuesta contra los valores de la variable de regresión x_1. Los puntos están ligeramente corridos para una mejor visualización de los datos. La curva azul (sólida) representa la curva de regresión logística para el modelo sin el coeficiente b_2 mientras que la curva roja (sólida) es la misma curva de regresión logística con el parámetro b_2 incluido. En punteado se han superpuesto las curvas de regresión logística ajustadas por el modelo propuesto con sus respectivos colores para los dos casos considerados.*

para $i = 1, ..., n$, donde h_i es el elemento $i-$ésimo de la diagonal de la matriz de proyección $P = X\left(X^tX\right)^{-1}X^t$.

La gran diferencia respecto de los residuos estandarizados del modelo lineal es que el denominador de los residuos Pearson estandarizados puede depender de los parámetros estimados. En el caso de regresión logística, que es el que interesa, ocurre esto último pues $V(\mu_i) = \mu_i(1-\mu_i)$ ya que ha de considerarse a la respuesta como variable aleatoria binomial. En general, los residuos Pearson heredan la asimetría de las variables respuesta, lo que puede resultar considerable en ciertos casos, y puede ser mejor estandarizar una respuesta transformada. Una manera de lograr esto es definiendo los residuos estandarizados en la escala del predictor lineal:

$$r_{Li} = \frac{g(y_i) - g(\widehat{\mu}_i)}{\{c_i\widehat{k}g'(\widehat{\mu}_i)V(\widehat{\mu}_i)(1-h_i)^{1/2}\}} \quad i = 1, ..., n.$$

Ajuste del modelo

El método de estimación de los coeficientes de regresión en un modelo logístico es por máxima verosimilitud. Un método que generaliza el de máxima verosimilitud y hace uso solo de los supuestos (7.3) y no de la distribución específica de los datos es el de quasi-verosimilitud. En este método, los estimadores minimizan $\sum r_i^2$ con $r_i = (y_i - \mu_i)/(c_iV(\mu_i))^{1/2}$ y se resuelven por mínimos cuadrados iterados ponderados. En cada paso de la iteración se hace una regresión a las respuestas ajustadas $z_i = \eta_i + (y_i - \mu_i)g'(\mu_i)$ en los x_i con pesos

$$w_i^{-1} = c_iV(\mu_i)g'(\mu_i)^2,$$

donde todas estas cantidad son evaluadas en los valores estimados hasta el momento (ver Davison y Hinkley (1997)). Por otro lado, la matriz de covarianzas asintótica para los coeficientes ajustados puede estimarse por:

$$\mathbf{\Sigma}_{\widehat{\beta}} \simeq k(X^tWX)^{-1},$$

con W la matriz diagonal de pesos evaluados en los últimos valores ajustados $\widehat{\mu}_i$. El vector de residuos $r = y - \widehat{\mu}$ tiene matriz de covarianzas aproximada $(I - H)\Sigma_Y$ con H la matriz de proyección de mínimos cuadrados ponderados. Por último, cuando el parámetro de dispersión k es desconocido, éste suele estimarse por :

$$\widehat{k} = \frac{1}{n-p-1}\sum_{i=1}^{n}\frac{(y_i - \widehat{\mu}_i)^2}{c_iV(\widehat{\mu}_i)}.$$

En el caso del modelo lineal, $V(\mu) = 1$ y $k = \sigma^2$ de donde $\widehat{k} = s^2$.

Desviación

La desviación, también conocida como *deviance* en inglés, es una medida muy importante en modelos de regresión logística. Esta medida juega un rol similar al de la suma del cuadrado de los residuos en el caso de modelos de regresión lineal, es decir que la desviación es una medida sobre la calidad del ajuste realizado. El principio es el mismo que el del test F en regresión lineal: comparar valores observados de la variable respuesta con los valores predichos de los modelos ajustados con y sin las variables consideradas.

Definición 7.3. Se define entonces la desviación para un modelo M dado como:

$$D(y) = -2[\log(p(y|\widehat{\theta})) - \log(p(y|\widehat{\theta}_s))],$$

donde p es la función de verosimilitud, $\widehat{\theta}$ es el conjunto de los valores ajustados de los coeficientes del modelo M y $\widehat{\theta}_s$ es el conjunto de valores ajustados de los coeficientes del modelo saturado. Un modelo saturado es un modelo que contiene igual cantidad de datos que de parámetros.

Esta definición tiene especial interés pues sabe que si los datos cumplen con los supuestos del modelo de regresión logística, entonces:

$$D \sim \chi^2_{n-(p+1)},$$

donde n es la cantidad de datos y p la cantidad de parámetros del modelo M (ver Hosmer y Lemeshow (1989)).

Con todo esto en mano ya se está en condiciones de hablar de métodos de remuestreo. La diagramación de éstos es la misma de siempre. Se pueden considerar simulaciones paramétricas con la particular desventaja de depender fuertemente de un buen ajuste, lo cual no siempre es posible con pocos datos. También se puede optar por simulaciones no paramétricas donde un método posible es el de remuestreo por pares que se aplica de forma idéntica al método de modelos lineales. Habiendo definido la noción de residuos en regresión logística parece conveniente definir un método de remuestreo por residuos. Nuevamente, el enfoque más simple imita el de modelos lineales si bien el esquema permite heteroscedasticidad en la varianza de las respuestas. Se definen las respuestas simuladas como:

$$y_i^* = \widehat{\mu}_i + \{c_i \widehat{k} V(\widehat{\mu}_i)\}^{1/2} \epsilon_i^* \quad i = 1, ..., n,$$

con $\epsilon_1^*, ..., \epsilon_n^*$ una muestra aleatoria de los residuos Pearson estandarizados y centrados $r_{Pi} - \overline{r_P}$. Otra propuesta consiste en realizar algo similar sobre el predictor lineal:

$$y_i^* = g^{-1}(x_i^t \widehat{\beta} + g'(\widehat{\mu}_i)\{c_i \widehat{k} V(\widehat{\mu}_i)\}^{1/2} \epsilon_i^*),$$

donde $g^{-1}(.)$ es la inversa de la función de enlace y los ϵ^* son tomados aleatoriamente de los residuos estandarizados en la escala del predictor lineal.

7.4 Ejemplo: Caña de azúcar

Los datos *cane* de la librería *boot* de R tienen por objetivo el análisis de una enfermedad común en las cañas de azúcar de algunas áreas de Brasil. Se tiene información sobre 45 variedades de caña de azúcar para las cuales se ha estudiado la cantidad r de brotes enfermos respecto de la cantidad m de brotes totales. Este mismo experimento se ha repetido un total de 4 veces por lo que se tienen cuatro bloques distintos con un total de 180 datos. La información puede ser pensada como una matriz de 4x45 de pares (m, r) y R los presenta de la siguiente forma:

```
> head(cane)
    n  r  x var block
1  87 76 19   1     A
2 119  8 14   2     A
3  94 74  9   3     A
4  95 11 12   4     A
5 134  0 12   5     A
6  92  0  3   6     A
```

Un modelo posible aunque simplista es el de considerar los brotes r_{ij} del i-ésimo bloque y la j-ésima variedad como una variable binomial con cantidad de repeticiones m_{ij} y probabilidad π_{ij}. Es decir,

$$r_{ij} \sim Binomial(\pi_{ij}, m_{ij}), \qquad E(r_{ij}) = \pi_{ij}\, m_{ij}.$$

Por ende, en un primer enfoque, se considera $y_{ij} = r_{ij}/m_{ij}$ las variables respuesta con media μ_{ij} igual a la probabilidad π_{ij} que un brote esté enfermo.

La función de varianza debe ser $V(\mu) = \mu(1-\mu)$ de forma tal que $c_{ij} = 1/m_{ij}$ con $k = 1$. El modelo se describe entonces por:

$$E(y_{ij}) = \mu_{ij}, \quad \mu_{ij} = \frac{e^{\alpha_i+\beta_j}}{1+e^{\alpha_i+\beta_j}},$$

$$Var(y_{ij}) = \frac{V(r_{ij})}{m_{ij}^2}, \quad V(rij) = m_{ij}\mu_{ij}(1-\mu_{ij}),$$

El análisis focaliza su interés en las variedades con menor β_j, es decir las más resistentes a la enfermedad. Dado que la deviance juega un papel similar al de la suma del cuadrado de los residuos en modelos lineales, este mismo valor permite generar confianza sobre la adecuación del ajuste. Por otro lado, suponiendo que π_{ij} está siendo correctamente modelado, la deviance debe entonces distribuirse como una χ^2_{132} (ver Hosmer y Lemeshow (1989)). Los grados de libertad de la deviance se calculan como la diferencia entre los grados de libertad del modelo saturado, es decir, un modelo con n parámetros para n datos, y los grados de libertad del modelo propuesto. En definitiva, se tiene que los grados de libertad para la deviance se calculan por:

$$df = n - (p + 1),$$

lo que es igual en este caso a

$$df = 180 - 48,$$

ya que se tiene que agregar a los 45 parámetros para cada variedad la influencia de cada bloque. El valor de la misma es de 1142.8 en el ejemplo con lo que se tiene una clara sobredispersión relativa al modelo. El panel izquierdo de la Figura 7.3 exhibe el valor de los predictores lineales del bloque A respecto de la variedad. Las variedades 1 y 3 son las menos resistentes mientras que la variedad 31 es la que muestra menos probabilidad de encontrar brotes enfermos. En el panel derecho se observa el gráfico de residuos versus predichos lineales y se ve como la asimetría a derecha tiende a desaparecer a medida que el η (el predictor lineal) es más grande.

En lo que sigue se busca un modelo que mejor ajuste a los datos. El primer enfoque propuesto fue el modelo paramétrico binomial aunque éste no parece reflejar la variabilidad de los datos. Por otro lado, siguiendo a Davison y Hinkley (1997), usando que una manera de modelar la función de varianza para datos binomiales sobredispersos está dada por

$$V(\pi) = \phi\pi(1 - \pi),$$

con $\phi > 1$ se puede realizar una simulación no paramétrica usando el primer método de remuestreo presentado en este capítulo y redondeando las respuestas al entero más cercano $0, 1, ..., m$. La Figura 7.4 muestra los boxplots de la relación deviance sobre sus grados de libertad en 200 simulaciones a partir del modelo binomial, a partir de simulaciones no paramétricas y para las mismas estratificadas según las 15 variedades con menor valor $\widehat{\beta}_j$, las 15 del medio y las 15 mayores. La línea punteada se corresponde con el valor observado de la deviance sobre sus grados de libertad. El modelo binomial es claramente no acertado mientras que la simulación no paramétrica estratificada es la mejor. Para realizar las simulaciones se utilizó el código de R que se presenta a continuación. En primer lugar, se define el modelo y se construyen los residuos Pearson estandarizados y centrados.

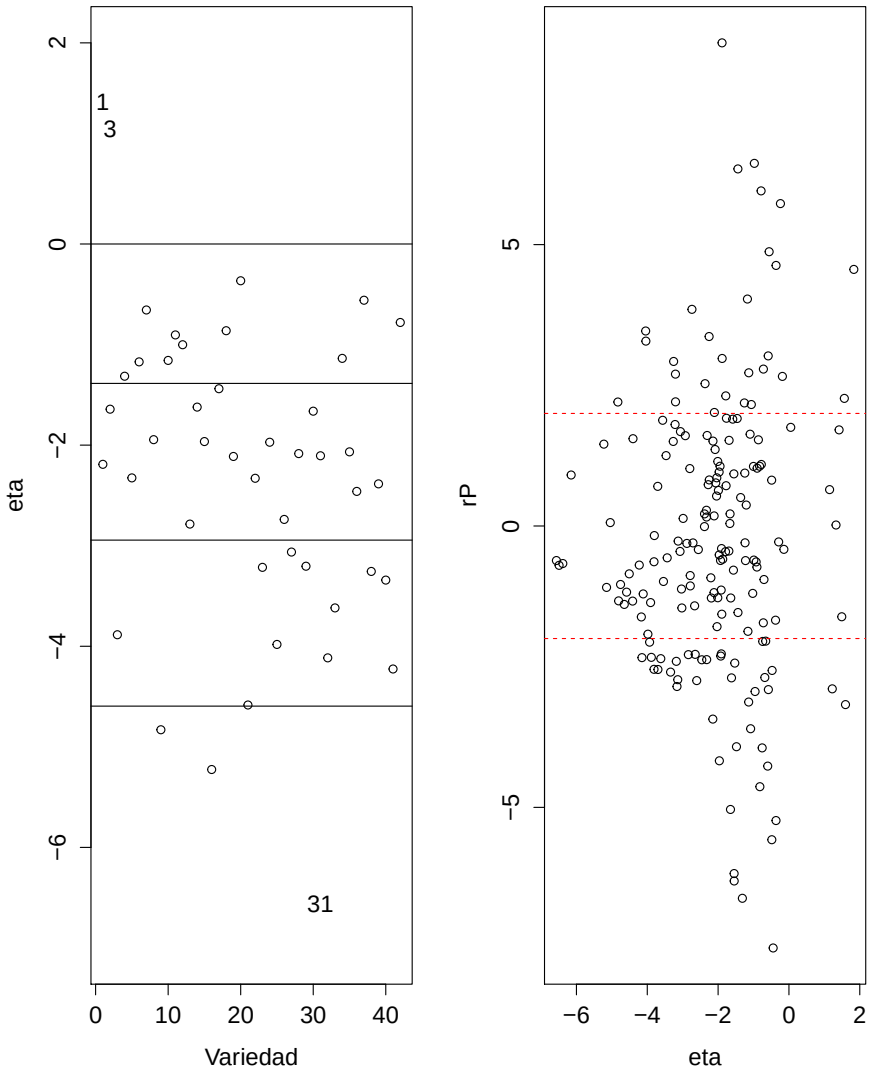


Figura 7.3: *A la izquierda se encuentra el diagrama de dispersión de los predictores lineales del bloque A contra la variedad. Se destacan las variedades 1 y 3 (menos resistentes) y la variedad 31 (más resistente). A la derecha se tiene el gráfico de residuos versus predichos lineales.*

```
 cane.glm<-glm(cbind(r, n-r) ~  var + block,
              family = binomial(link = "logit"),data=cane)
cane.diag<-glm.diag.plots(cane.glm,ret=T)
num<-r/n-cane.glm$fit
denom<-sqrt((1/n)*cane.glm$fit*(1-cane.glm$fit)*8.3*(1-cane.diag$h))
cane.res <- num/denom # residuos pearson estandarizados
cane.res <- cane.res - mean(cane.res)
cane.df <- data.frame(cane,res=cane.res,fit=fitted(cane.glm))
```

Las simulaciones paramétricas binomiales se pueden realizar de la siguiente manera:

```
cane.fun <- function(data)
{ tmp <- glm(cbind(r, n-r)~  var + block,
              family = binomial(link = "logit"),data=data)
deviance(tmp)}
cane.sim <- function(data, mle)
{
for (i in 1:nrow(data)){
data$r[i] <- rbinom(1,data$n[i], mle[i])
}
data
}
cane.mle <- fitted(cane.glm)
cane.boot.sim <- boot(cane, cane.fun, R=199,
sim="parametric", ran.gen=cane.sim, mle=cane.mle)
```

Las simulaciones no paramétricas se adecuan al siguiente código:

```
cane.model <- function(data, i)
{ d <- data
y<-d$fit + sqrt((1/(d$n))*8.3*d$fit*(1-d$fit))*d$res[i]
y2 <- y*d$n
for (i in 1:length(d$n)){

if(y2[i]<0) y2[i]<-0
if(y2[i]>cane.df$n[i]) y2[i]<-cane.df$n[i]
if (y2[i]-floor(y2)[i]>0.5) y2[i]<-floor(y2[i]+1)
               else y2[i]<- floor(y2[i])
}
```

```
d$r<-y2
tmp<-glm(cbind(d$r, d$n-d$r)~  var + block,
                family = binomial(link = "logit"),data=d)
deviance(tmp)
}
cane.boot<-boot(cane.df,cane.model,R=199,
                            strata=cane.df$block)
```

En este caso, lo más interesante es el proceso de simulación utilizado. Como se puede observar en el código, se redondean al entero más cercano los valores obtenidos en el remuestreo ya que las variables consideran valores enteros. Por último, se replica lo mismo pero para simulaciones estratificadas. El proceso de simulación es semejante.

```
cane2.df<-cane.df
attach(cane2.df)
est.var<-predict(cane.glm)
cane2.df$est<-est.var

bloqueD<-cane2.df[cane2.df$block=="D",]
ordenD<-order(bloqueD$est)
cane2.df$strata<-rep(1,180)
bloqueD$strata<-rep(1,45)
bloqueD$strata[ordenD]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="D",]<-bloqueD

bloqueA<-cane2.df[cane2.df$block=="A",]
ordenA<-order(bloqueA$est)
bloqueA$strata<-rep(1,45)
bloqueA$strata[ordenA]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="A",]<-bloqueA

bloqueB<-cane2.df[cane2.df$block=="B",]
ordenB<-order(bloqueB$est)
bloqueB$strata<-rep(1,45)
bloqueB$strata[ordenB]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="B",]<-bloqueB

bloqueC<-cane2.df[cane2.df$block=="C",]
ordenC<-order(bloqueC$est)
bloqueC$strata<-rep(1,45)
```

```
bloqueC$strata[ordenC]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="C",]<-bloqueC

cane.boot.2<-boot(cane2.df,cane.model,R=199,strata=cane2.df$strata)
```

Se ordenan los predictores dentro de cada bloque para generar los estratos adecuados.

Ranking de las variedades

Al principio se vio que para los datos del ejemplo, las variedades 1 y 3 eran las menos resistentes en el bloque A y que la variedad 31 era la más resistente. Las replicaciones bootstrap pueden generar distribuciones de los rankings asociados a las variedades más y menos resistentes. Por ejemplo, la variedad 1 tiene asociado el ranking 45 con los datos originales. En la Figura 7.5 se encuentra el histograma de los ranking de la variedad 1 en 1000 replicaciones bootstrap así como el análogo para la variedad 31. La tendencia se mantiene en todas las replicaciones. La variedad 1 es la peor rankeada un 60% de las veces y es la anteúltima el resto de las ocasiones mientras que la variedad 31 ocupa el primer lugar 550 veces sobre el total de repeticiones.

7.5 Predicción en clasificación

En esta sección, se estudia el problema de predicción para modelos de regresión logística. En el contexto de este tipo de modelos, en donde las variables respuesta representan dos o más categorías, al problema de predicción se lo conoce como problema de clasificación. Para una variable respuesta dicotómica y puede pensarse por ejemplo que $y_i = 1$ afirma la presencia de una enfermedad en el individuo x_i mientras que el valor $y_i = 0$ la descarta. Es natural que se cometan errores en la clasificación de ciertas observaciones pues el modelo que se usa para ajustar los datos no es exacto. Esto quiere decir que para una observación (x_0, y_0), donde y_0 es la categoría teórica de x_0, el valor predicho por el ajuste del modelo de regresión logística utilizado para el caso x_0, denominado $\widehat{y}_0$, es distinto de y_0. Como los valores predichos por el modelo ajustado no son siempre números enteros, se define una función, denominada función de costo c, que determine la calidad de la predicción. En el caso de regresión lineal se había considerado como función de costo $c(y, \widehat{y}) = (y - \widehat{y})^2$. En este trabajo, para el modelo de regresión logística, se considerará la función de costo

$$c(y, \widehat{y}) = I_{\{|y-\widehat{y}|>0.5\}},$$

de modo que si el valor absoluto de la diferencia entre la respuesta teórica y el valor predicho por el ajuste del modelo propuesto es menor a 0.5 se considerará que no ha habido error en la predicción correspondiente a esa observación.

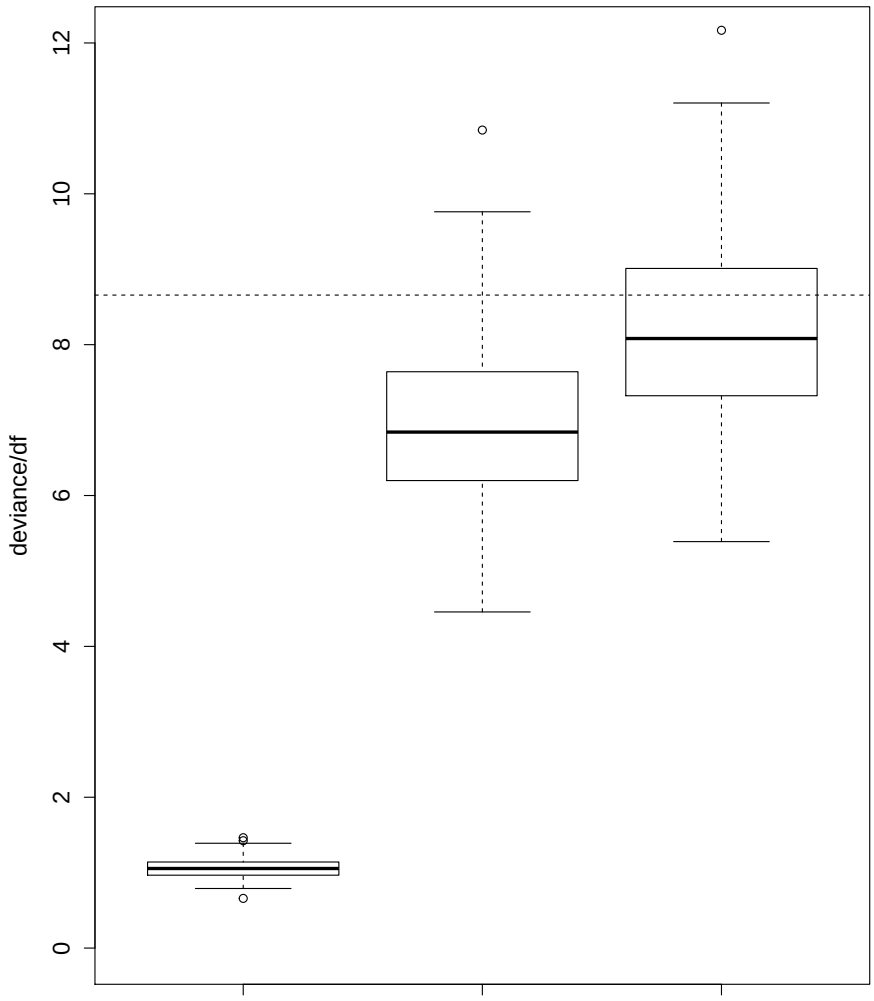


Figura 7.4: *A partir de $B = 200$ simulaciones se encuentran las distribuciones bootstrap (boxplot) de la deviance/df para remuestreo binomial (primer boxplot desde la izquierda), remuestreo no paramétrico sin estratificar (boxplot del medio) y remuestreo no paramétrico con estratificación (boxplot a derecha). Se ha superpuesto además el valor observado de la deviance sobre sus grados de libertad.*

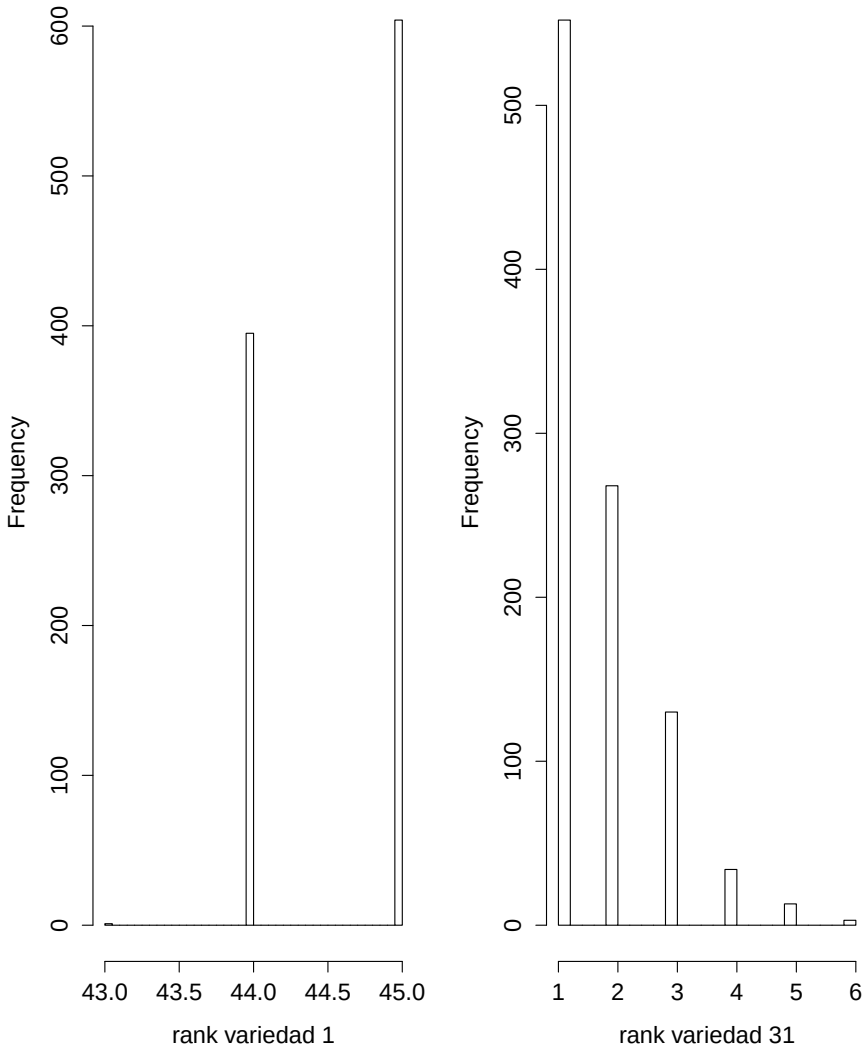


Figura 7.5: *A la izquierda se tiene el histograma de los ranking de la variedad 1 en el bloque A para 1000 replicaciones bootstrap. A derecha se repite el procedimiento para la variedad 31 del bloque A.*

En términos más formales, si se ha ajustado el modelo de regresión logística con el conjunto de datos $z = \{(x_i, y_i)\}_{i=1}^n$ y se quiere conseguir un valor predicho para y en una nueva observación x_0, el error de clasificación se define, de la misma manera que se ha hecho en el Capítulo 5, por

$$err(z, F) = E_F[c(y_0, \widehat{y}_0)], \tag{7.4}$$

donde la esperanza se toma sobre la nueva observación (x_0, y_0) obtenida a partir de la distribución F. Aquí se considera fijo el conjunto de datos con el que se ajustó el modelo. Por otro lado, una vez más como se hizo en el Capítulo 5, se define la tasa del error de predicción aparente como

$$err(z, \widehat{F}) = E_{\widehat{F}}[c(y_0, \widehat{y}_0)],$$

lo que para el caso en donde $\widehat{F}$ es la distribución empírica de los datos, se tiene que

$$err(z, \widehat{F}) = \#\{|y_i - \widehat{y}_i| > 0.5\}/n.$$

Estas últimas dos definiciones, análogas a las definiciones dadas en el caso de predicción en modelos de regresión lineal, permiten obtener el error de predicción bootstrap mejorado. En este caso, se propone una nueva estimación del error de predicción conocida como *error de predicción bootstrap 0.632*.

Se recuerda que la estimación del error de predicción bootstrap más simple propuesta por Efron (1993) se define por:

$$\widehat{E}_{\widehat{F}}(err(z^*, \widehat{F})) = \frac{1}{n}\sum_{i=1}^{n}\sum_{b=1}^{B} c(y_i, \widehat{y}_{ib}^*)/B, \tag{7.5}$$

donde $err(z^*, \widehat{F})$ es la estimación bootstrap (o plug-in) del error de predicción (7.4).

El estimador bootstrap 0.632

La ecuación (7.5) puede pensarse como el promedio sobre $i = 1, .., n$ de la estimación del error de predicción para cada punto individual (x_i, y_i). De hecho, es por eso que se han invertido los sumandos en la ecuación. El método que aquí se presenta pretende ajustar el *optimismo* en el método bootstrap mejorado. Con ese fin, y bajo la idea de que estimar el

error de predicción del dato (x_i, y_i) tiende a ser mayor cuando la muestra bootstrap usada para predecir no incluye al dato en cuestión se define la cantidad ϵ_0 como

$$\epsilon_0 = \frac{1}{n}\sum_{i=1}^{n}\sum_{b\in B_{(-i)}} c(y_i, \widehat{y}^*_{bi})/B_i,$$

con $B_{(-i)}$ el conjunto de índices de las muestras bootstrap que no contienen al dato $i-$ésimo y B_i el número de dichas muestras. Se define entonces el estimador 0.632 bootstrap del optimismo como:

$$opt_{0.632} = 0.632(\epsilon_0 - err(z, \widehat{F})),$$

donde $err(z, \widehat{F})$ es la tasa del error aparente del modelo. Agregando esta estimación al mismo error aparente se obtiene el estimador 0.632 del error de predicción

$$\begin{aligned}\widehat{err}^{0.632} &= err(z, \widehat{F}) + 0.632(\epsilon_0 - err(z, \widehat{F})) \\ &= 0.368 err(z, \widehat{F}) + 0.632\epsilon_0.\end{aligned}$$

El factor $0.632 \simeq (1 - e^{-1})$ es aproximadamente el límite de la probabilidad de que una observación dada aparezca en una muestra bootstrap de tamaño n cuando $n \to \infty$. Esto es así pues la probabilidad de que una observación dada no aparezca en la muestra bootstrap es $(1 - 1/n)^n$ que tiene por límite justamente e^{-1}. Ya se está en condiciones de presentar un ejemplo de error de clasificación (predicción) en el contexto de un modelo de regresión logística.

7.6 Ejemplo: Análisis de laboratorio. Predicción de presencia de cristales de oxalato

En este ejemplo, se estudia la variable respuesta binaria y que representa la presencia de cristales de oxalato de calcio en 79 muestras de orina. Los datos fueron extraídos del paquete *boot* de R y las covariables consideradas son *gravedad* (la densidad de la orina relativa al agua), *pH*, *osmolaridad* (mOsm), *conductividad* (mMho millimHo), *concentración de urea* (milimoles por litro) y *concentración de calcio* (milimoles por litro). Dos casos fueron eliminados por falta de información recuperando un restante de 77 datos. Los datos pueden visualizarse con el comando *parallelplot* de la librería *lattice* de R.

```
library(boot)
data<-urine[c(-1,-55),]
attach(data)
#visualizacion de la data

parallelplot(~data[, 2:7] | factor(r), data = data)
```

El gráfico resultante se presenta en la Figura 7.6. Estos gráficos pueden detectar patrones respecto de los valores tomados por las covariables en las distintas categorías consideradas. En este caso no se observa un patrón relevante, aún así, la figura puede ser interesante para vislumbrar comportamientos extraños como se verá al final del ejemplo.

El análisis del valor de la *desviación* (deviance en inglés) sugiere el uso del modelo que incluye las 4 covariables *gravedad*, *conductividad*, *log de la concentración de calcio* y *log de la concentración de urea*. Por ende, el análisis de predicción constará de la información aportada por dicho modelo. Como este ejemplo puede entenderse como un problema de clasificación es coherente el uso de la función de costo $c(y, \widehat{y}) = I_{\{y \neq \widehat{y}\}}$. En la práctica se utiliza una función equivalente que vale 1 si el valor absoluto de la diferencia entre valor predicho y valor observado es mayor a 0.5 y 0 en caso contrario. Esta función se definió en la Sección 7.5. En la Tabla 7.2 se muestran los valores obtenidos del error de predicción según distintos métodos. Se destacan el error aparente, el método bootstrap mejorado, el bootstrap 0.632 y diferentes valores para el Validación Cruzada K-fold.

```
data<-urine[c(-1,-55),]
attach(data)
cost <- function(r, pi=0) mean(abs(r-pi)>0.5)
urine.glm <- glm(r~gravity+cond+log(calc)+log(urea),
           binomial,data=data)
urine.diag <- glm.diag(urine.glm)
#error de prediccion aparente
app.err <- cost(r, fitted(urine.glm))
#error de pred K-fold cross-validation ajustado
cv.err <- cv.glm(data, urine.glm, cost, K=77)$delta
cv.38.err <- cv.glm(data, urine.glm, cost, K=38)$delta
cv.10.err <- cv.glm(data, urine.glm, cost, K=10)$delta
cv.7.err <- cv.glm(data, urine.glm, cost, K=7)$delta
cv.2.err <- cv.glm(data, urine.glm, cost, K=2)$delta
```

Se usa la función *glm* en vez de *lm* para el uso de regresiones lineales generalizadas y se especifica la familia de distribuciones a usar (para el caso de regresión logística el análisis está centrado en familias binomiales). La naturaleza discontinua del error de predicción puede dar resultados más variables que en el caso de regresión lineal. Por ejemplo, el método de Validación Cruzada es más sensible respecto de qué observaciones caen en cada grupo.

En la Figura 7.7 se pueden observar los boxplot de las cantidad $y_j - \mu(x_j^*, \widehat{F})$ que contribuyen a la estimación 0.632 bootstrap del error de predicción, graficados contra la obser-

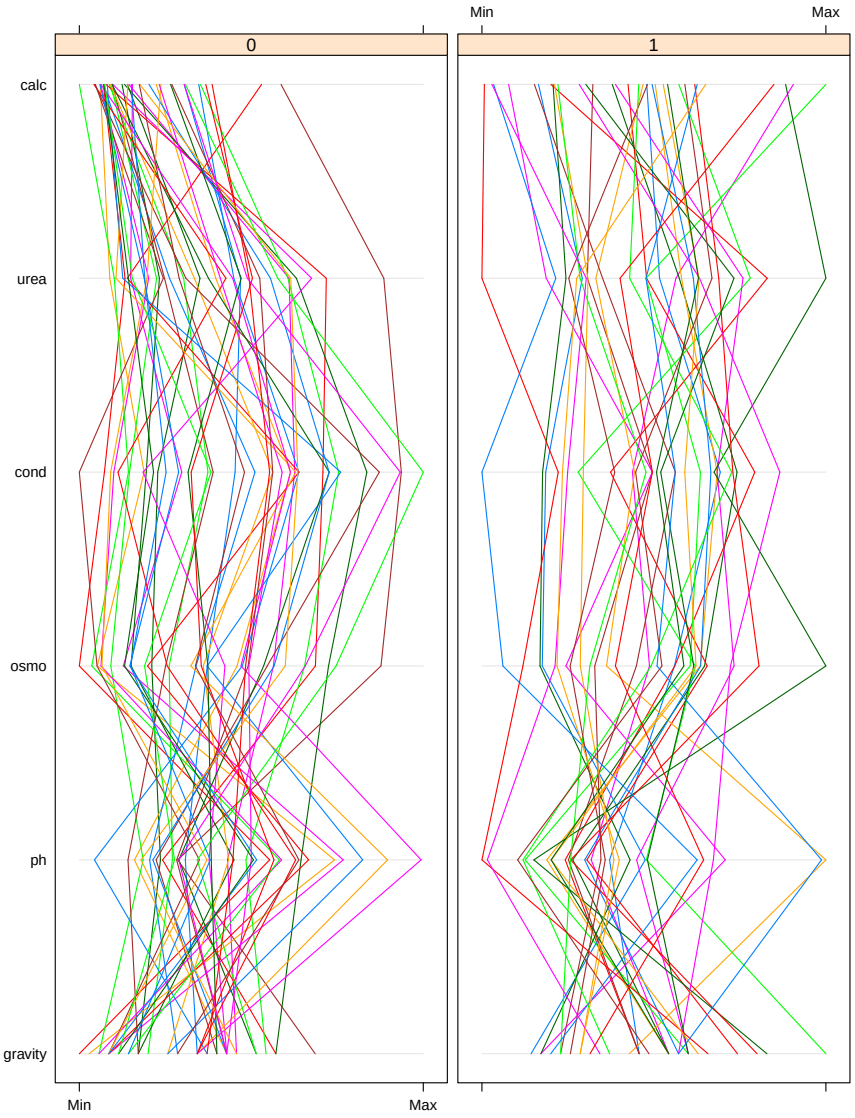


Figura 7.6: *Valores de las covariables para cada dato (segmentos con colores distintos) discriminando la información según la presencia o la ausencia de cristales.*

err ap	Bootstrap	0.632	77	38	10	7	2
20.8	24.3	21.8	23.6	24.8	24.6	21.6	23.3

Tabla 7.2: *Estimaciones del error de predicción agregado ($\times 10^2$ o error de clasificación para los datos orina del paquete boot de R. Las últimas 5 columnas son estimaciones usando el método K-fold de Validación Cruzada y la primera línea indica el valor de K en cada caso.*

vación j ordenadas según el valor de los residuos. Sólo los valores que sobrepasan las líneas marcadas contribuyen al error, dado que la función de costo elegida considera que una observación ha sido mal clasificada si el valor absoluto del error es mayor a 0.5. Como se puede notar, observaciones con residuos próximos a cero tienden a tener mejores predichos que observacion con residuos de valor absoluto grande. De hecho, los últimos dos boxplots (los de residuos más importantes), correspondientes a los casos 77 y 53 son siempre predecidos de forma incorrecta. El código de R para el cálculo del error bootstrap 0.632 se puede encontrar a continuación. Los detalles respecto de cada comando son especificados debajo de cada segmento respectivo.

```
#error de pred 0.632 bootstrap
#C\'alculo de los errores 0.632:

# proceso de simulacion S
urine.pred.fun <- function(data, i, model)
 {
 d <- data[i,]
 d.glm <- update(model,data=d)
 # vuelve a ajustar el modelo actualizado
 pred <- predict(d.glm,data,type="response")
 # type response devuelve predichos en la escala
 # de la variable dependiente
 # (por default lo daria en la escala del predictor lineal)
 D.F.Fhat <- cost(data$r, pred)
 # error de pred del modelo ajustado por
 #la muestra boot sobre la muestra original
 D.Fhat.Fhat <- cost(d$r, fitted(d.glm))
 # error aparente en cada muestra boot
 c(data$r-pred, D.F.Fhat - D.Fhat.Fhat)
 # la resta del primero con el segundo devuelve el optimismo
 # (eso aparece en la ultima columna del urine.boot$t)
 }
```

Una vez definido el proceso de simulación, que debe calcular el optimismo, se procede con el cálculo de las replicaciones:

```
urine.boot <- boot(data, urine.pred.fun, R=200,
               model=urine.glm)
 # 200 replicaciones bootstrap
```

Se calcula entonces el error de predicción bootstrap, y se ordenan los datos para el cálculo del error 0.632:

```
urine.boot$f <- boot.array(urine.boot)
 # dice cuantas veces aparece cada observacion
 #en cada muestra bootstrap
n <- nrow(data)
err.boot <- mean(urine.boot$t[,n+1]) + app.err
 # error bootstrap mejorado
ord <- order(urine.diag$res)
 # se ordenan los residuos
urine.pred <- urine.boot$t[,ord]
 # y se ordenan las predicciones seg\'un el orden
 # de los residuos
```

Además, para cada observación se analizan las muestras bootstrap en las que no ha aparecido esta misma y se calcula el error de predicción bootstrap 0.632:

```
err.632 <- 0
n.632 <- NULL
pred.632 <- NULL
for (i in 1:n) {
 # En el siguiente for uno considera
 # las muestras bootstrap en las que no aparece cada
 # observacion
 inds <- urine.boot$f[,i]==0
 err.632 <- err.632 + cost(urine.pred[inds,i])/n
 # se calcula el error de la obs i con las muestras
 # en las que no aparece i
 }
err.632 <- 0.368*app.err + 0.632*err.632
 # se calcula finalmente el error boot 0.632
```

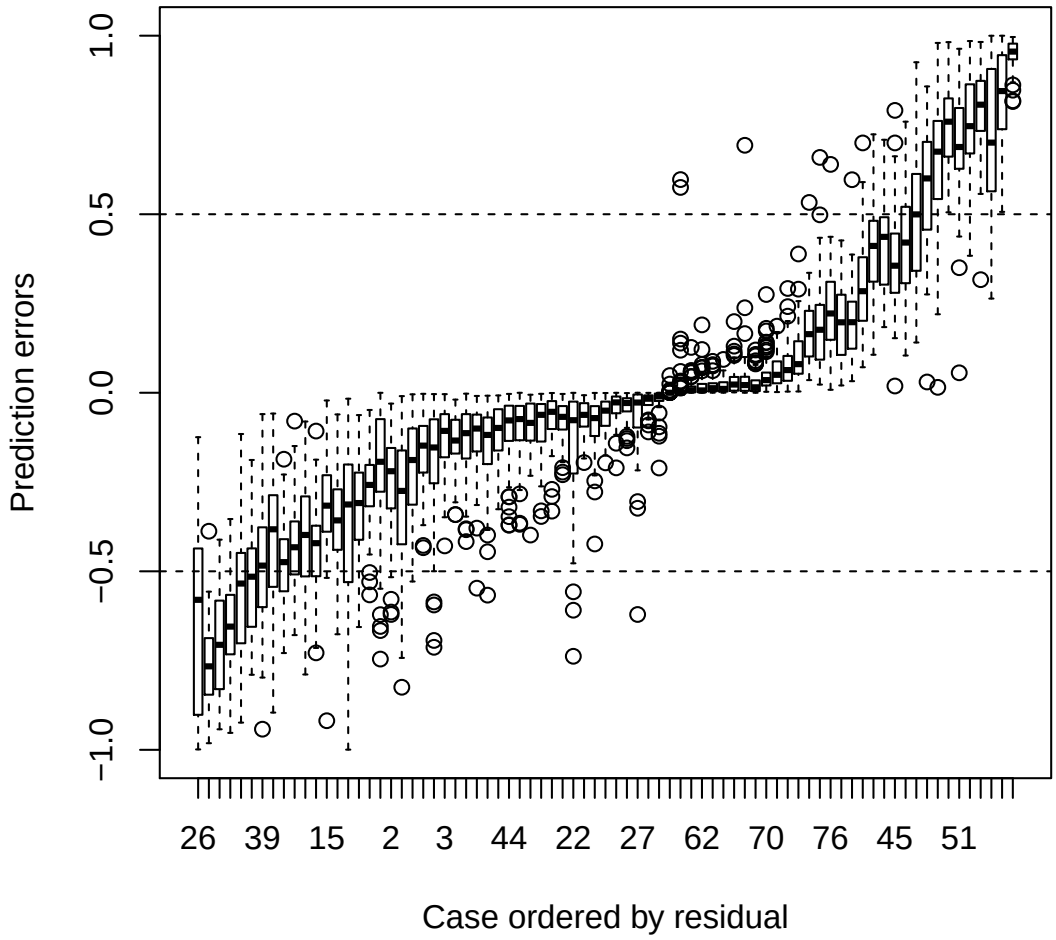


Figura 7.7: *Componentes de la estimación 0.632 bootstrap del error de predicción agregado para B=200 simulaciones.*

Es interesante tratar de entender por qué razón las dos observaciones destacadas son siempre mal predecidas. Uno esperaría que hubiese un comportamiento atípico en estos dos casos o tan simplemente una mala elección de ajuste. Para el caso de la observación 77 (observación con presencia de cristales (r=1)), uno podría esperar, por ejemplo que las observaciones sin presencia de cristales (r=0) tengan valores en las covariables más parecidos a los de esta observación que los datos correspondientes a las observaciones con presencia de cristales (r=1). Es decir, un comportamiento efectivamente atípico. De hecho, esto mismo puede observarse en el gráfico de la Figura 7.8. Los valores en las covariables de la observación 77 son más parecidos a la media de los valores tomados por las covariables de observaciones sin presencia de cristales. Es decir que el comportamiento de la observación 77 se asemeja más al comportamiento de la categoría opuesta.

En contrapartida se puede analizar el mismo gráfico para una observación siempre bien clasificada (Figura 7.9). Es el caso de la observación 27 del tipo r=0. Círculos rojos y verdes se encuentran a mayor proximidad que círculos rojos y puntos unidos por segmentos.

Se puede también realizar un análisis numérico un tanto más profundo. Cada dato posee 6 valores de las 6 covariables consideradas. Es posible analizar, para cada observación, la cantidad de valores de dichas covariables que se asemejan más en valor absoluto a la media de los valores de la covariable del tipo opuesto. Por ejemplo, para la observación 77, que es de tipo r=1, se tienen los siguientes valores en las covariables:

```
as.numeric(data[77,2:7])
[1]   1.015   6.030 416.000  12.800 178.000   9.390
```

Mientras que la media de los valores en las covariables para observaciones de tipo r=0 y r=1 respectivamente son las siguientes:

```
cero.1
[1]   1.015364   6.125682 561.659091  20.550000 232.431818
      2.628864
uno.1
[1]   1.021576   5.927273 682.878788  21.378788 302.363636
      6.202424
```

En este caso, en 5 oportunidades sobre 6, la observación 77 tiene un comportamiento más parecido al tipo opuesto (r=0). Se puede ver que, aproximadamente, un 22% de las observaciones tiene 5 o más valores que se asemejan más al tipo opuesto que al que pertenecen. Este porcentaje, es un valor común en el mundo médico y explica el número de boxplots en la Figura 7.7 que son generalmente mal clasificados.

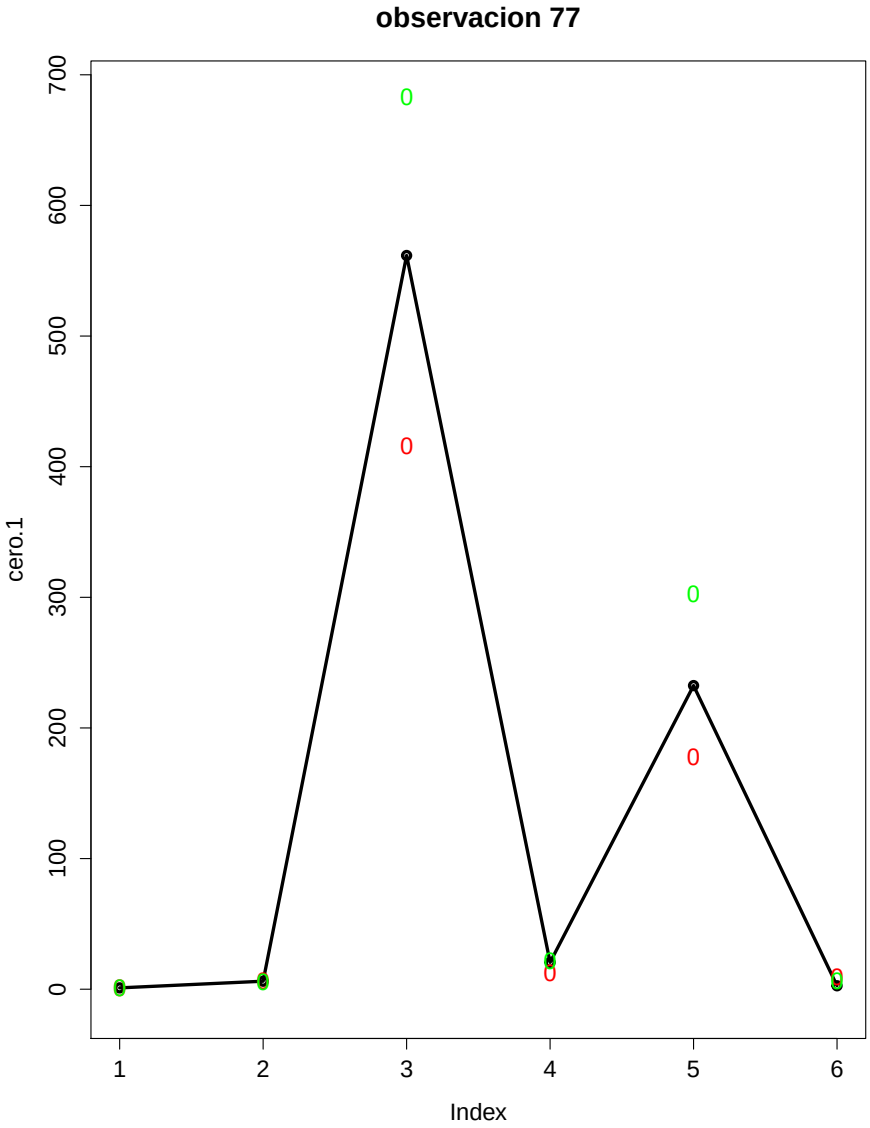


Figura 7.8: *Unidos por segmentos se encuentran los promedios de los valores de las covariables para observaciones con r=0 (sin presencia de cristales) Los círculos rojos se corresponden con los valores de las covariables de la observación 77 mientras que en verde se encuentran los promedios de los valores de las covariables para observaciones con r=1 (con presencia de cristales).*

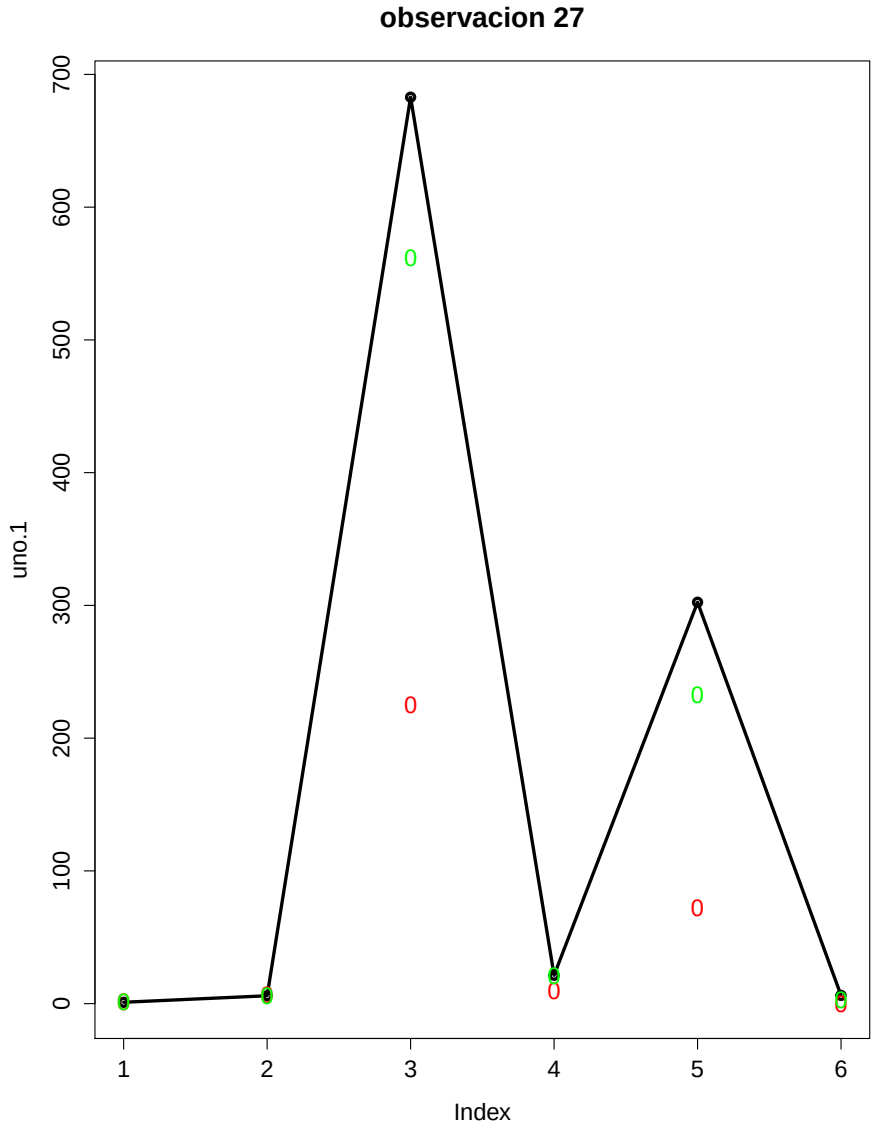


Figura 7.9: *Unidos por segmentos se encuentran los promedios de los valores de las covariables para observaciones con r=1 (con presencia de cristales) Los círculos rojos se corresponden con los valores de las covariables de la observación 27 mientras que en verde se encuentran los promedios de los valores de las covariables para observaciones con r=0 (sin presencia de cristales).*

Capítulo 8

El caso de anemia en pacientes mayores de 60 años

Este ejemplo toma datos amablemente aportados por el hospital de la Universidad de Duesseldorf que serán analizados en el contexto de un modelo de regresión lineal. El ejemplo propone un análisis novedoso en el que se estudia en pacientes mayores de 60 años la capacidad explicativa de 11 variables dependientes respecto de una variable respuesta Hgb_{Admit} que mide la cantidad de hemoglobina en sangre (g/dl). En un primer análisis, se consideran sólo las covariables *sex* (el sexo del paciente), única variable dicotómica y de relevancia médica a priori, *age* (edad del paciente), WBC_{Admit} (número de leucocitos, 10^9/l a la admisión), CRP_{Admit} (proteína C-reactiva medida en mg/l y a la admisión) y $Crea_{pre}$ (creatinina medidad en mg/dl y a la admisión) pues se ha querido trabajar con variables continuas. La relevancia médica de la variable *sex* en este caso ha obligado su inclusión en este primer estudio. Otras variables, todas ellas discretas, serán además consideradas y descriptas en el análisis de selección de variables para este mismo ejemplo.

Se propone ajustar los datos con un modelo de regresión lineal múltiple y para ello se analiza, en un principio, la influencia individual de las covariables en la variable respuesta mediante el gráfico de diagramas de dispersión cruzados que puede observarse en la Figura 8.1.

Es difícil apreciar un comportamiento particular dado que las variables WBC, CRP y $Crea$ son muy asimétricas. Al tomar el logaritmo de estas mismas covariables en la regresión múltiple se tiene una mejor descripción y visualización de la Figura anterior (ver Figura 8.2).

La función $scatterplotMatrix$ de la librería *car* de R realiza este mismo gráfico au-

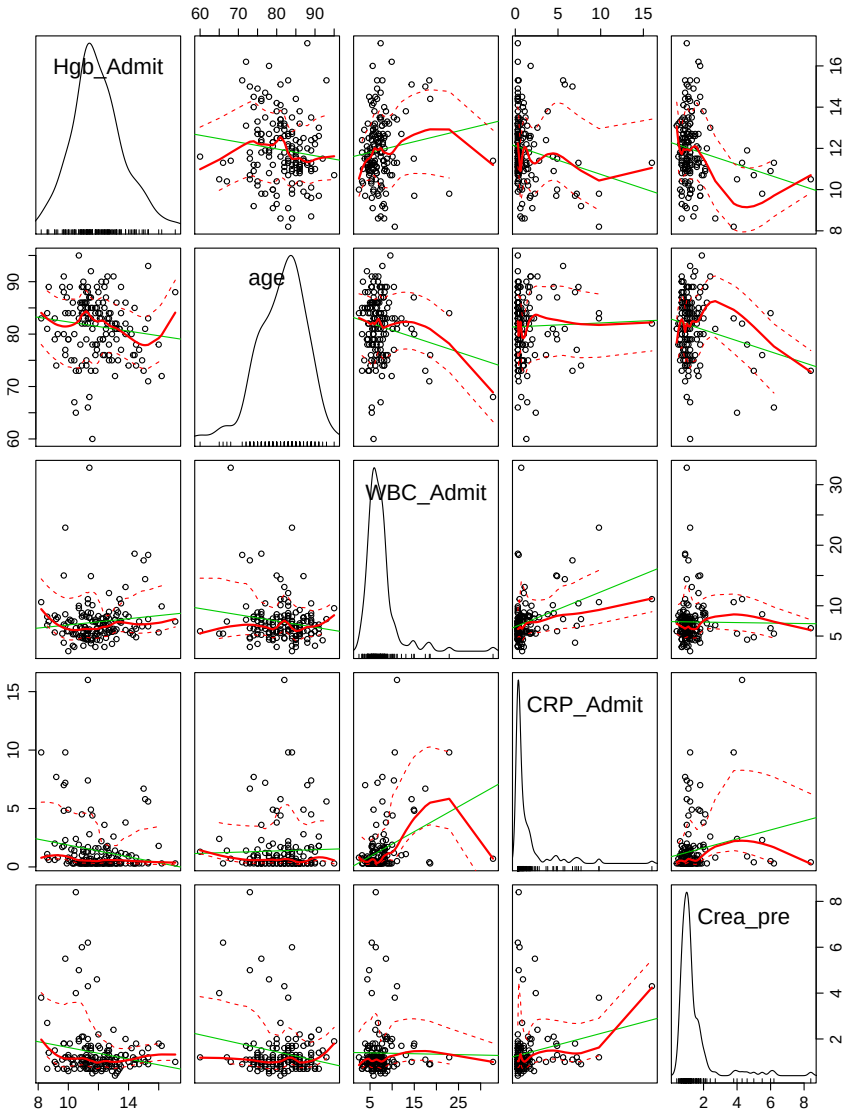


Figura 8.1: *Diagramas de dispersión cruzados de la regresión lineal múltiple.*

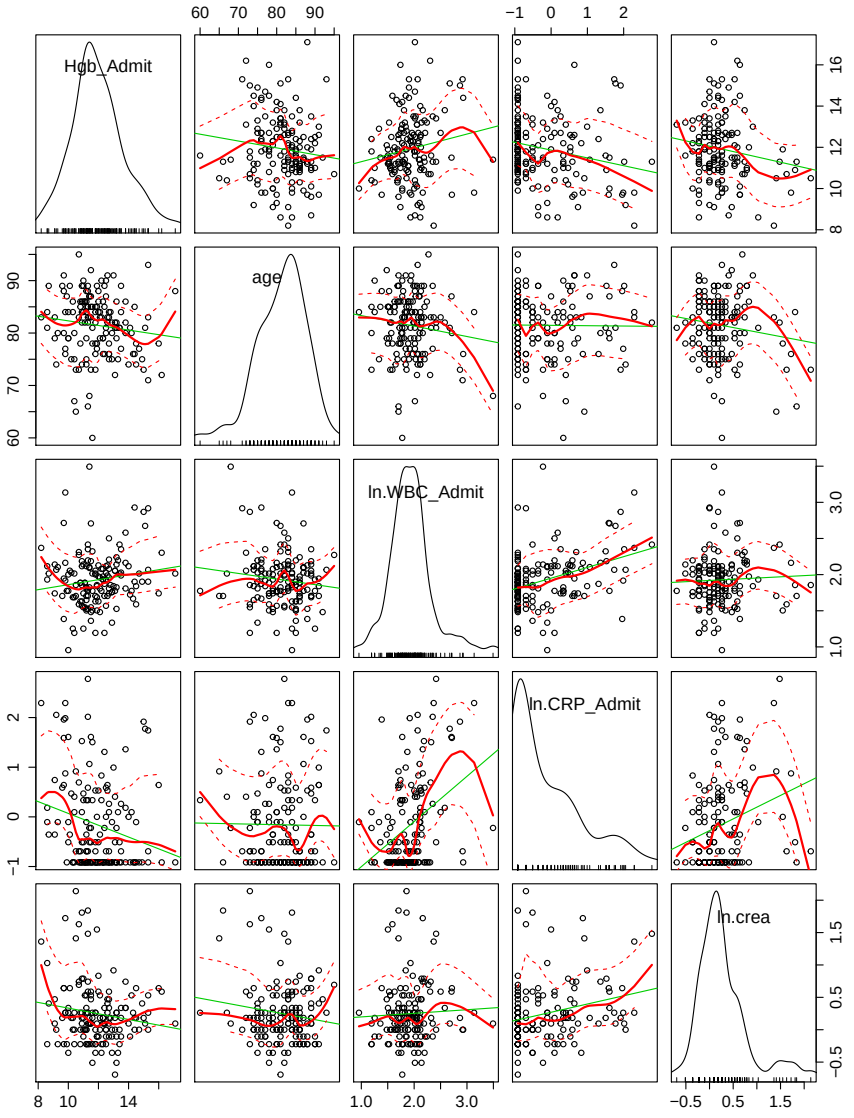


Figura 8.2: *Diagramas de dispersión cruzados de la regresión lineal múltiple donde WBC, CRP y Crea son consideradas en escala logarítmica.*

tomáticamente.

```
scatterplotMatrix( ~ Hgb_Admit + age + ln.WBC_Admit +
       ln.CRP_Admit + ln.crea, data = dat)
```

En la primera fila se pueden apreciar los diagramas de dispersión de las variables independientes con la variable dependiente. No es extraño en datos reales observar diagramas sin un patrón específico o fácil de analizar y éste no es un caso ajeno a esta particularidad. Aún así, una regresión múltiple no puede ser analizada únicamente por la explicación que aporta individualmente cada covariable. Generalmente, los aportes de cada variable independiente pueden apreciarse en el contexto global de la regresión. De hecho, aún cuando los gráficos no muestran una relación evidente entre covariable y variable respuesta, destacando que los supuestos del modelo lineal se cumplen razonablemente bien para el ajuste utilizado (ver Figura 8.3), el $p-$valor del estadístico F que se encuentra en el siguiente código parece justificar la elección del modelo.

```
model.anem1 <- lm(Hgb_Admit ~ sex + age + ln.WBC_Admit +
          ln.CRP_Admit + ln.crea, data = dat)
summary(model.anem1)

Call:
lm(formula = Hgb_Admit ~ sex + age + ln.WBC_Admit +
   ln.CRP_Admit + ln.crea, data = dat)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4202 -0.8779 -0.1377  0.8640  4.7443

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  12.88797    2.01160   6.407 1.98e-09 ***
sexwoman     -0.64978    0.27689  -2.347  0.02030 *
age          -0.02930    0.02144  -1.367  0.17389
ln.WBC_Admit  0.96662    0.37008   2.612  0.00996 **
ln.CRP_Admit -0.44934    0.15294  -2.938  0.00385 **
ln.crea      -0.66061    0.30073  -2.197  0.02964 *


Residual standard error: 1.532 on 144 degrees of freedom
Multiple R-squared:  0.1575,Adjusted R-squared:  0.1282
F-statistic: 5.382 on 5 and 144 DF,  p-value: 0.0001441
```

Una vez más, una forma simple de obtener las réplicas bootstrap de los estimadores de los coeficientes de regresión es mediante la función *Boot* de la librería *car* de R.

```
m1.boot <- Boot(model.anem1, R = 1000)
```

Para este ejemplo se opta por la metodología de pares para realizar el análisis por la naturaleza de las covariables (si bien ya se ha visto que la metodología puede no ser influyente en los resultados finales).

La Figura 8.4 exhibe, a modo de ejemplo, el histograma de las réplicas bootstrap del logaritmo de los leucocitos. Además, la función $confint$ de la librería $MASS$ de R genera los intervalos de confianza bootstrap BC_a de nivel 95% para los coeficientes de regresión. La única covariable menos influyente en el modelo parece ser *age* que incluye el valor 0 en el intervalo.

```
confint(m1.boot)
Bootstrap quantiles, type =  bca

                   2.5 %      97.5 %
(Intercept)   8.55222175 17.09169052
sexwoman     -1.10328860 -0.08107647
age          -0.07461913  0.01822375
ln.WBC_Admit  0.16125767  1.75739636
ln.CRP_Admit -0.74957956 -0.03029672
ln.crea      -1.16303910 -0.15241299
```

Se tiene además una comparación gráfica de los intervalos de confianza mediante el uso del comando $coefplot$ de la librería *car* de R en la Figura 8.5.

Es común que la visualización de datos y respuestas no sea fácilmente comprensible al utilizar un modelo de regresión lineal usando los datos en su forma cruda. Ciertas transformaciones de los datos pueden ayudar en la interpretación de los resultados. Por ejemplo, las transformaciones lineales no afectan el ajuste de un modelo de regresión lineal (Gelman y Hill (2007)). Aún así la elección de una transformación lineal puede mejorar la interpretación de los coeficientes y del ajuste en si. En el ejemplo que aquí se trata, el reescalamiento y centrado de los datos mejora la interpretabilidad de los intervalos de confianza por ejemplo. Con los datos crudos, las longitudes de los intervalos no son comparables realísticamente entre los coeficientes. Por otro lado, las densidades de las distribuciones bootstrap tampoco lo serían. Es por eso que se ha decidido reescalar y centrar los datos provenientes de distribuciones continuas, es decir, para todas las covariables con excepción del sexo. Esto mismo puede realizarse de la siguiente forma:

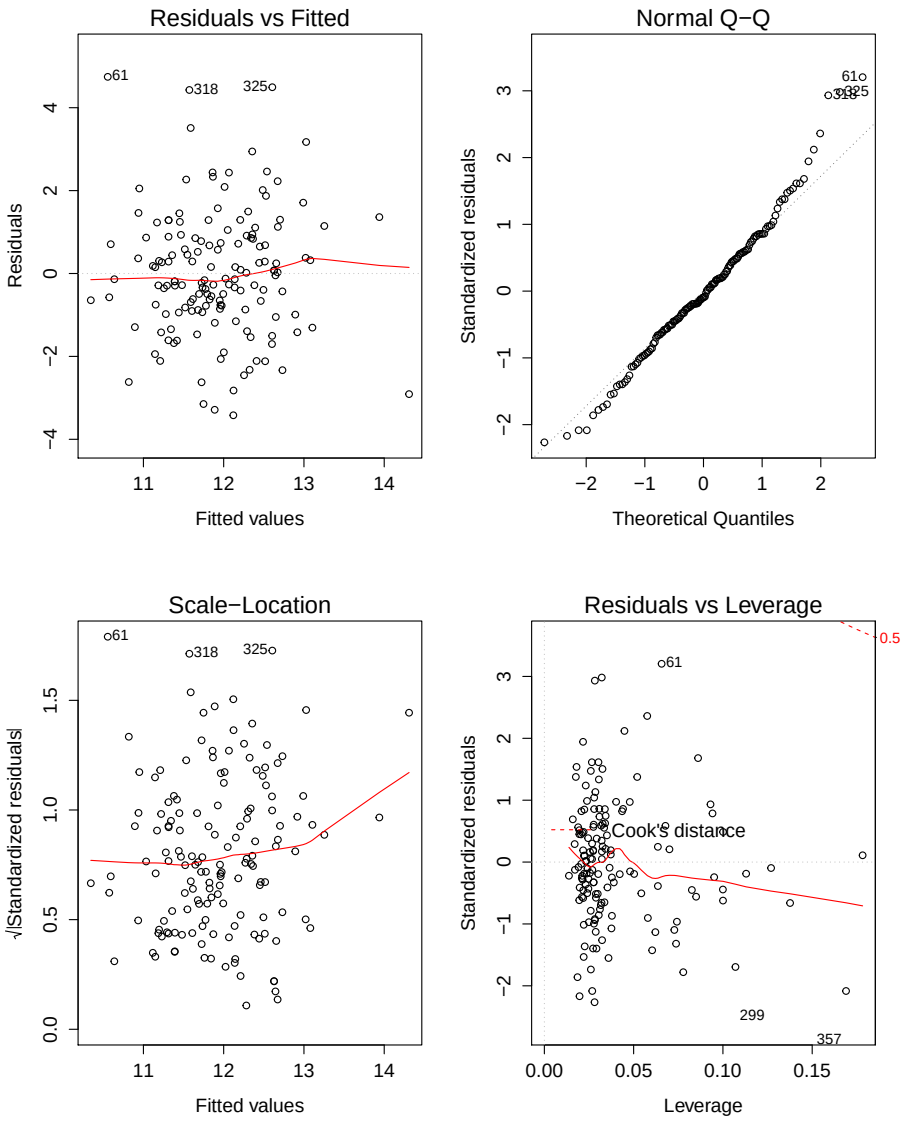


Figura 8.3: *Plot del modelo de regresión múltiple propuesto para el ejemplo de anemia. Diagnóstico de los supuestos del modelo lineal.*

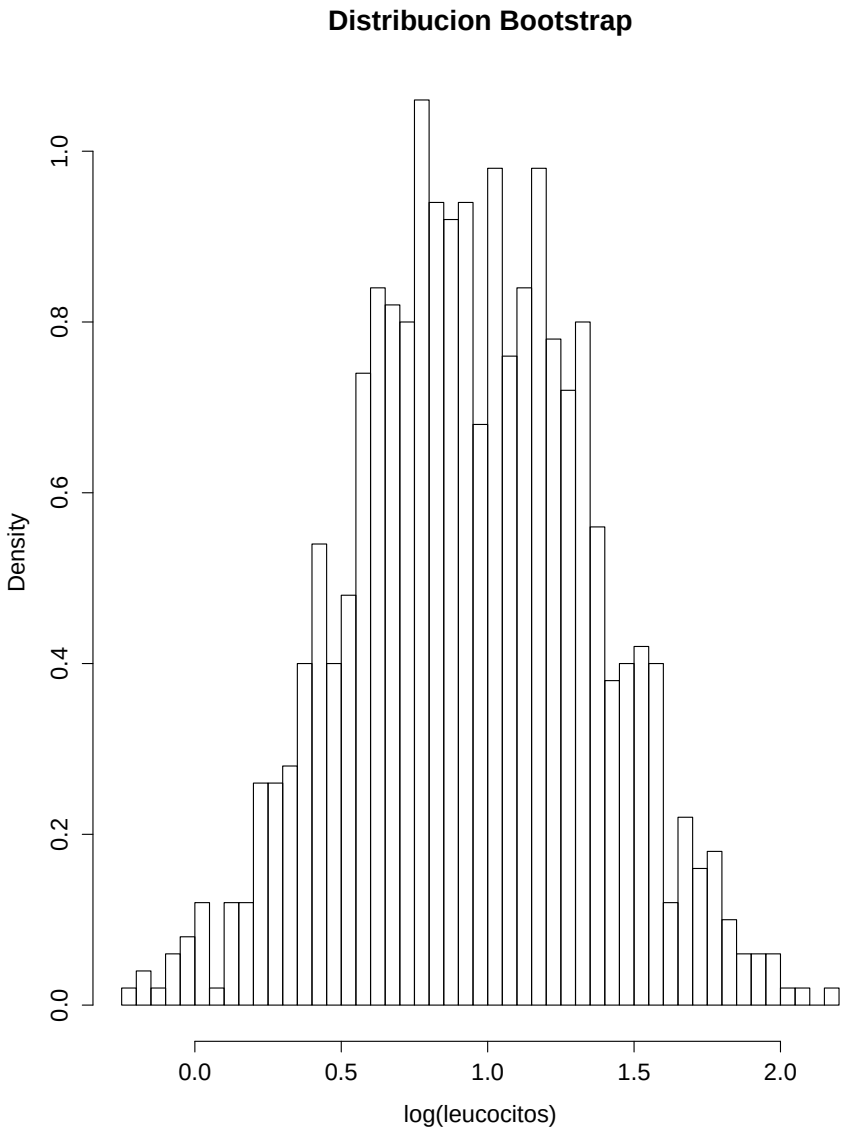


Figura 8.4: *Histograma de las B=1000 replicaciones bootstrap del logaritmo de los leucocitos.*

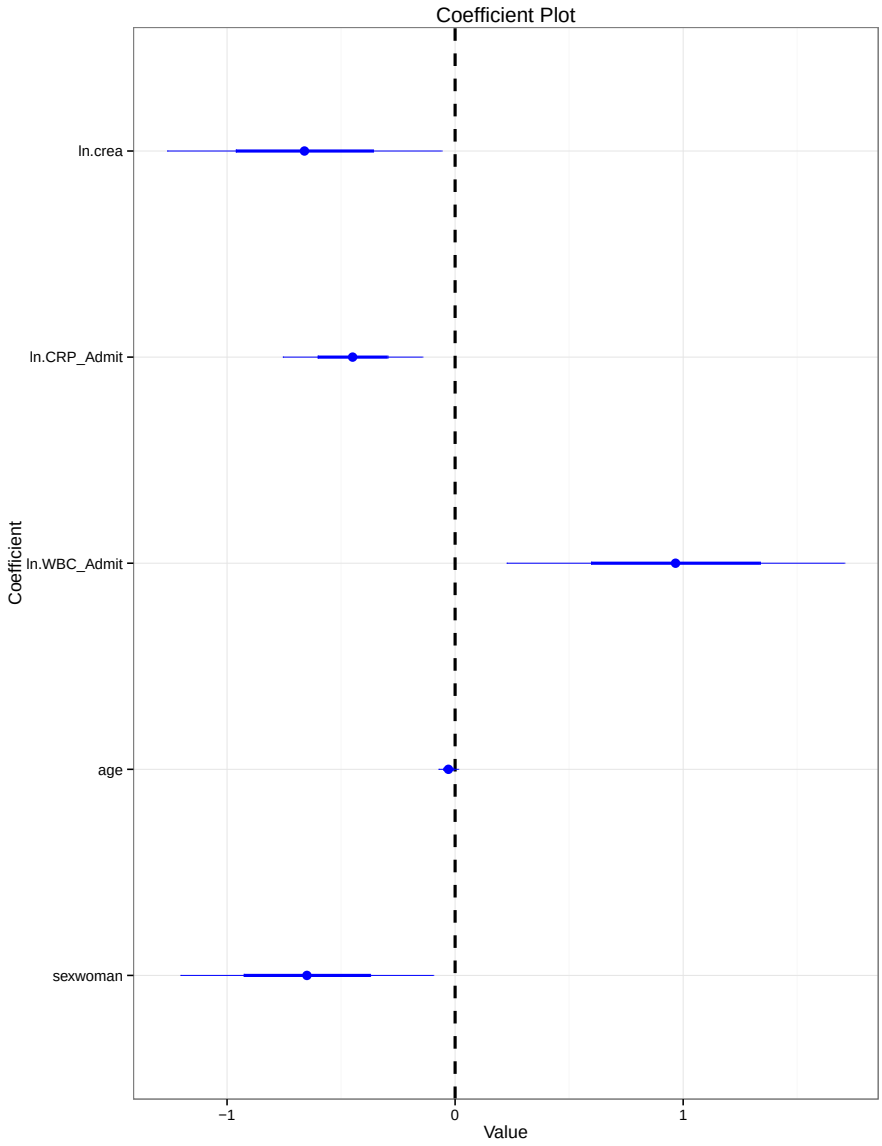


Figura 8.5: *Intervalos de confianza de nivel 95% para los coeficientes de regresión del ajuste múltiple sin escalar previamente las covariables.*

```
attach(dat)
X<-cbind(age,ln.WBC_Admit,ln.CRP_Admit,ln.crea)
X2<-scale(X)
datos<-cbind(Hgb_Admit,sex,X2)
dat2<-data.frame(datos)
model.anem2<-lm(Hgb_Admit ~ sex + age + ln.WBC_Admit +
        ln.CRP_Admit + ln.crea, data = dat2)
```

El resultado del ajuste se puede observar a continuación y en él se comprueba la equivalencia de ambos ajustes a través de los p-valores del test de t.

```
summary(model.anem2)

Call:
lm(formula = Hgb_Admit ~ sex + age + ln.WBC_Admit + ln.CRP_Admit +
    ln.crea, data = dat2)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4202 -0.8779 -0.1377  0.8640  4.7443

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   12.9203     0.4426  29.192  < 2e-16 ***
sex           -0.6498     0.2769  -2.347  0.02030 *
age           -0.1749     0.1280  -1.367  0.17389
ln.WBC_Admit   0.3595     0.1376   2.612  0.00996 **
ln.CRP_Admit  -0.4134     0.1407  -2.938  0.00385 **
ln.crea       -0.3155     0.1436  -2.197  0.02964 *

Residual standard error: 1.532 on 144 degrees of freedom
Multiple R-squared:  0.1575,Adjusted R-squared:  0.1282
F-statistic: 5.382 on 5 and 144 DF,  p-value: 0.0001441
```

La Figura 8.6 muestra los mismos intervalos de confianza para el ajuste con los datos escalados y centrados. La longitud del intervalo para el coeficiente *age* ya no es tan pequeña en comparación con las demás.

Por otro lado, la Figura 8.7 exhibe el histograma de B=1000 replicaciones bootstrap del coeficiente de regresión *sex* en la que se han superpuesto las densidades aproximadas boot-

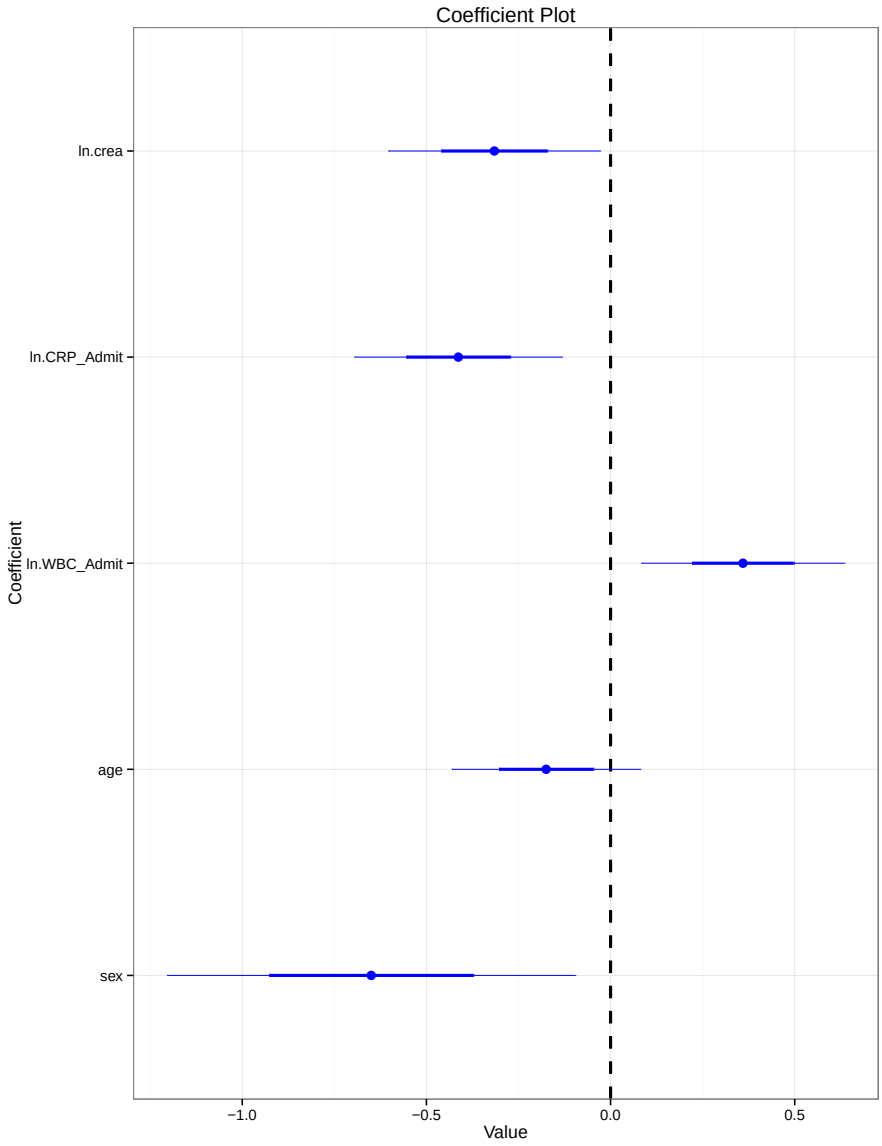


Figura 8.6: *Intervalos de confianza de nivel 95% para los coeficientes de regresión del ajuste múltiple habiendo escalado y centrado previamente las covariables continuas. histograma de las B=1000 replicaciones bootstrap de la ordenada al origen escalada y centrada. Se han superpuesto además las densidades aproximadas de los demás coeficientes de regresión (sexo en rojo, edad en verde, leucocitos en azul, proteína en negro y creatinina en violeta).*

strap del mismo y los demás coeficientes, en donde pueden compararse las distribuciones de los mismos.

Un análisis logístico pudo haberse realizado para este ejemplo usando algún criterio para determinar la anemia o la falta de anemia en los pacientes. Para el caso, el *odds ratio* pudo haber sido una medida capáz de explicar, de forma eficaz, la cantidad de información que una covariable aporta a la variable dependiente. A continuación, se se procederá a seleccionar variables con el método de Akaike en el que se incluirán las 11 covariables originales, es decir, además de las 5 ya utilizadas, el análisis hará uso de las variables discretas CKD (variable dicotómica que discrimina pacientes con enfermedad crónica en los riñones), DM_{Pre} (variable dicotómica que discrimina pacientes diabéticos), KHK_{Pre} (variable dicotómica que discrimina pacientes con enfermedades cardíacas), $LVEF_{Pre}$ (variable dicotómica que discrimina pacientes con disfunción ventricular), $Dialyse_{Pre}$ (variable dicotómica que discrimina pacientes que han recibido diálisis) y $anemie_{admit2}$ (variable dicotómica que discrimina pacientes con anemia utilizando el criterio WHO: 12g/l para las mujeres y 13g/l para los hombres).

El código de R que aquí se presenta propone una metodología para aproximar la distribución bootstrap de la cantidad de variables elegidas en el proceso de selección de variables con el criterio AIC. Para ello se utilizan todas las variables del ejemplo de datos de anemia, es decir 11 covariables más el intercept (7 categóricas y 4 continuas). Por otro lado, se han escalado y centrado las variables continuas tal cual se ha hecho anteriormente. Se propone además el uso de los métodos de pares y de residuos en el proceso de simulación. Las distribuciones bootstrap obtenidas por ambos métodos pueden apreciarse en la Figura 8.8. El programa que permite el cálculo de los estimadores de mínimos cuadrados en el modelo de regresión múltiple se describe a continuación:

```
model.anem1 <- lm(Hgb_Admit ~ .,
 data = dat2)
summary(model.anem1)

Call:
lm(formula = Hgb_Admit ~ ., data = dat2)

Residuals:
    Min      1Q  Median      3Q     Max
-3.2757 -0.9845 -0.0468  0.8948  4.7014

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   11.6755     1.3838   8.437 3.77e-14 ***
```

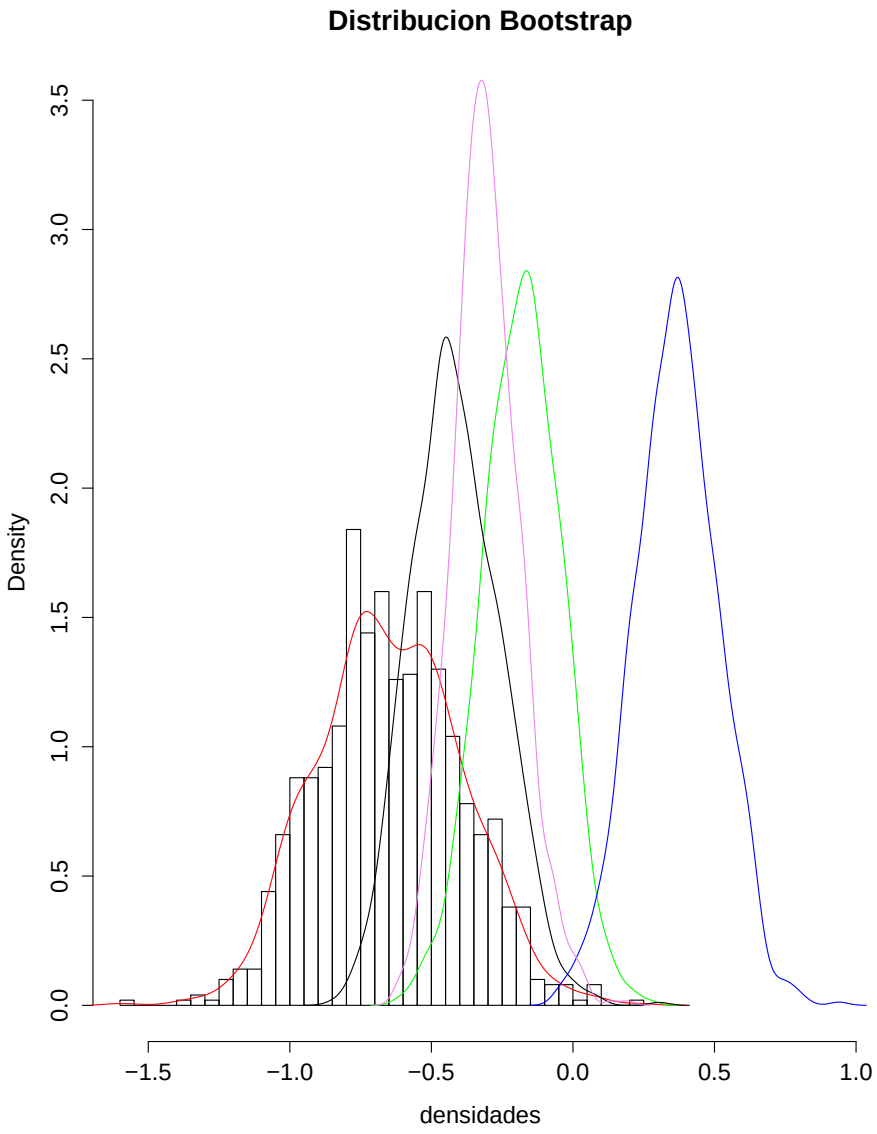


Figura 8.7: *Histograma de las B=1000 replicaciones bootstrap del coeficiente de regresión sex. Se han superpuesto además las densidades aproximadas de los demás coeficientes de regresión (sexo en rojo, edad en verde, leucocitos en azul, proteína en negro y creatinina en violeta) con los datos previamente escalados y centrados.*

```
sex           -0.4540     0.3183  -1.426  0.15600
CKD            0.1610     0.3764   0.428  0.66956
DM_pre         0.4267     0.2789   1.530  0.12824
KHK_pre        0.4654     0.2781   1.674  0.09648 .
LVEF_pre       0.3852     0.2540   1.517  0.13165
Dialyse_pre   -1.1631     0.8869  -1.311  0.19188
age           -0.2195     0.1376  -1.595  0.11309
ln.WBC_Admit   0.3218     0.1371   2.347  0.02035 *
ln.CRP_Admit  -0.4597     0.1399  -3.286  0.00129 **
ln.crea       -0.1380     0.2952  -0.467  0.64106

Residual standard error: 1.501 on 139 degrees of freedom
Multiple R-squared:  0.2191,Adjusted R-squared:  0.1629
F-statistic:   3.9 on 10 and 139 DF,  p-value: 0.0001092
```

Pocas variables parecen significativas en el modelo de regresión múltiple y el método de selección de variables debe ayudar a descartar un subgrupo de poco carácter explicativo. El proceso bootstrap para el método de pares con el criterio AIC se describe a continuación y el histograma de la réplicas se encuentra en el panel derecho de la Figura 8.8.

```
p.star<-NULL   # cantidad de variables del modelo
coef.names<-NULL
for(b in 1:500)
{
  ind <- sample(1:150, replace = T)
  boot.fix.data <- dat2[ind, ]
  tmp.0 <-  lm(Hgb_Admit ~ . , data=boot.fix.data)
  suppressWarnings(tmp<- stepAIC(tmp.0,trace=F))
  p.star<-c(p.star,length(coef(tmp)))
  coef.names <- c(coef.names, names(tmp$coeff))
}
```

En la variable *coef.names* se ha registrado además el porcentaje de veces que las variables han sido elegidas en el proceso de selección de variables que cuenta con 500 repeticiones. El menor número de muestras bootstrap considerado en este caso respecto del análisis previo tiene que ver con el alto costo computacional que este proceso implica. Se considera, de todos modos, que $B = 500$ es un valor razonable. El método bootstrap permite en este caso realizar un análisis de sensibilidad del método de selección de variables

ofreciendo al analista mayor confianza respecto de la calidad explicativa de cada covariable en el modelo de regresión múltiple. En el Capítulo 9 se verá en otro ejemplo la aplicación de este método. Se detalla entonces en la Tabla 8.1 el porcentaje de veces que cada covariable ha sido elegida luego de aplicar el método de selección de variables en 500 muestras bootstrap.

Ordenada al origen 1.000	EDAD 0.660	CKD 0.254	DIALISIS 0.654
DM 0.514	ln.CREA 0.394	ln.CRP 0.944	ln.WBC 0.870
LVEF 0.548	SEXO 0.624	KHK 0.690	

Tabla 8.1: *Proporciones de veces con que las covariables han sido elegidas en el proceso de selección de variables usando el criterio AIC y con 500 muestras bootstrap.*

Se destacan por ejemplo las covariables $ln.CRP_{Admit}$ y $ln.WBC_{Admit}$ que han sido elegidas un 94% y un 87% de las veces respectivamente. Se repite el mismo proceso para el remuestreo por residuos:

```
 m1 <- glm(Hgb_Admit ~ .,
                   data = dat2)

p.diag<-glm.diag.plots(m1,ret=T)
p.res <- p.diag$res
p.res <- p.res - mean(p.res)
p.df <- data.frame(dat2,res=p.res,fit=fitted(m1))

p.fit <- function(data){
  tmp.0<-glm(Hgb_Admit~.,data=data)
  suppressWarnings(tmp<- stepAIC(tmp.0,trace=F))
  length(coef(tmp))
}
p.model <- function(data, i)
{ d <- data
d$Hgb_Admit <- d$fit + d$res[i]
p.fit(d) }
p.boot <- boot(p.df, p.model, R=500)
```

Los resultados de los histogramas no son idénticos como era de esperarse por el uso de métodos distintos. Aún así, en ambos casos, el número de variables del modelo una vez concluido el método de selección parece oscilar entre 6 y 9.

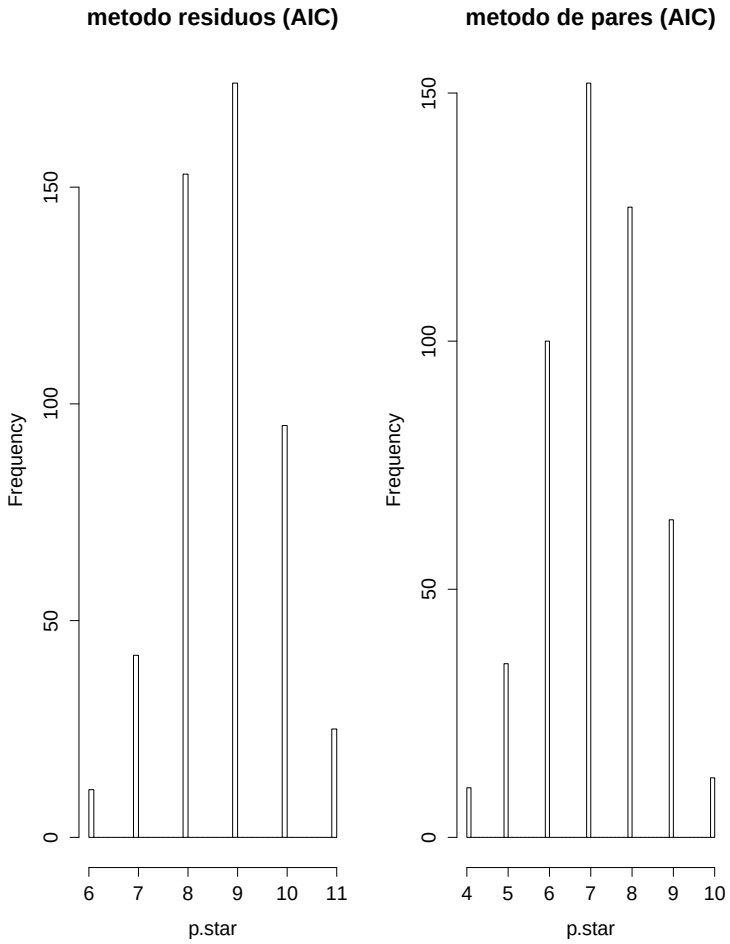


Figura 8.8: *En el panel izquierdo se tiene el histograma de las replicaciones bootstrap del número de variables del modelo en la aplicación del método de selección de variables (AIC). Para ello se han generado 500 muestras bootstrap bajo la metodología de residuos, se ha aplicado el comando stepAIC a cada una de ellas y se ha tenido en cuenta el número de variables del modelo resultante. En el panel derecho se tienen las mismas replicaciones para la metodología de pares.*

Capítulo 9

Un caso de aplicación en problemas de investigación clínica

9.1 Predictores de mortalidad para infecciones de tejidos blandos necrotizantes : un análisis retrospectivo de 64 casos

En Krieg *et al.* (2014), se analiza un conjunto de datos reales para los que el objetivo consiste en la búsqueda de predictores de mortalidad para infecciones de tejidos blandos necrotizantes (NSTI). Se intentará en lo que sigue aplicar los métodos estadísticos descriptos en los Capítulos 3 a 7 a los datos analizados en dicho trabajo a fin de realizar un estudio ms completo de los mismos.

Las infecciones de tejidos blandos necrotizantes aunque raras, son enfermedades con una alta tasa de mortalidad. De hecho se registra un rango histórico de 16% a 34% en la tasa de mortalidad asociada a las enfermedades NSTI. En el conjunto de datos estudiado en Krieg *et al.* (2014), fallecieron el 32.8% de los $n = 64$ pacientes tratados en el mismo hospital de la universidad de Duesseldorf entre 1996 y 2011. Teniendo en cuenta además la rapidez en la progresión de dichas enfermedades, reviste especial interés la búsqueda de predictores que permitan a los pacientes tener una asistencia veloz y pertinente. Con ese fin, se han identificado variables que pueden influir en la evolución de la enfermedad incluyendo variables demográficas, clínicas, de laboratorio y parámetros microbiológicos que se han comparado entre el grupo de sobrevivientes y de no sobrevivientes.

Se distinguen 35 covariables de importancia clínica a priori (19 continuas y 16 discretas) que se describirán a continuación y una variable explicativa y definida por:

$$y = \begin{cases} 1 & \text{si la persona ha fallecido} \\ 0 & \text{si no} \end{cases} \tag{9.1}$$

Krieg *et al.* (2014) realizaron, en primer lugar, un análisis para cada covariable donde se estudió si existen diferencias en esa variable entre vivos y muertos, es decir, se hizo un test de hipótesis para discriminar entre los dos grupos de individuos. En el caso de las variables continuas, se utilizó el test de t para dos muestras independientes con el fin de comparar sus medias. En el caso de variables discretas, se comparó la distribución entre ambos grupos utilizando el test exacto de Fischer. En particular, en el caso de una variable dicotómica, se consideró los Odd Ratio ya que el test basado en la distribución logística es asintótico y no exacto. Este tipo de análisis se denomina screening y tiene por objetivo seleccionar las variables con mayor potencial predictivo respecto de la variable respuesta y. En Krieg *et al.* (2014), los autores proponen seleccionar, para un futuro análisis de regresión logística múltiple entre las 35 variables inicialmente propuestas, aquellas que tuvieron, para los tests de igualdad antes mencionados, un $p-$valor menor a 0.1. Otro enfoque con el mismo fin es propuesto en el capítulo 10 de Efron y Gong (1983).

Aún así, no parece razonable decidir la inclusión de variables en el análisis de regresión logística múltiple solamente considerando los $p-$valores del análisis exploratorio univariado. Deben tomarse en cuenta factores, como la correlación de variables que puedan apreciarse únicamente en el análisis de regresión múltiple. A modo de ejemplo, se mostrará simplemente el análisis univariado correspondiente a la variable 'vasoconstrictor ' que corresponde a si el individuo tomó o no vasodilatadores.

Para los cálculos que se han realizado, los datos faltantes en los factores de riesgo han sido imputados tomando la mediana y la moda para variables cuantitativas y cualitativas, respectivamente. En el análisis de regresión logística múltiple se ha utilizado el criterio de Akaike para la selección de variables, para el cual, su expresión general, descripta a su vez en el Capítulo 6, es $AIC = -2ln(L) + 2p$ donde L es la función de verosimilitud y p la cantidad de covariables del modelo. Como ya se ha mencionado en el Capítulo 6, el valor de AIC definido en dicho capítulo es un caso particular de esta definición para modelos lineales con residuos normales. Por otro lado, se ha enfatizado en el análisis que se presentará, en el *Odds Ratio*, la razón de chances, una medida estadística de especial importancia en el mundo epidemiológico que se ha descripto en el Capítulo 7. Se usará además el método bootstrap para estimar su distribución y analizar sus características.

Variables en el análisis de NSTI

El criterio para confirmar el diagnóstico de NSTI se basó en observaciones médicas como cambios en las características de la piel y en observaciones intra-operativas tales necrosis de la fascia superficial, grasa o músculo, así como en el reporte final histopatológico. El historial de cada paciente fue revisado para extraer información demográfica, comorbilidades preexistentes y factores de riesgo, observaciones clínicas y de laboratorio del momento de ingreso al hospital como también resultados microbiológicos de especímenes de tejidos que fueron obtenidos durante el primer desbridamiento quirúrgico.

Las observaciones clínicas en el momento de ingreso al hospital incluyen cambios en la piel, como desprendimientos, necrosis, ampollas, las causas primarias de infección, así como parámetros fisiológicos como ritmo cardíaco y presión sanguínea. Además, el uso de asistencia como ventilación mecánica o diálisis fue también considerada. Otras variables, relevantes durante el momento de internación, como fallos en la actividad renal han sido tomadas en cuenta.

De forma general se han considerado las $p-1$=35 variables siguientes: Edad, sexo, fiebre, leucocitos elevados, PCR (proteína c-reactiva), ventilación post-operativa, diálisis, ASA, índice de masa corporal elevado, necrosis en piel, desprendimiento de la piel, bullas, hipotonía, taquicardia, falla renal, sepsis, localización infección, tipo de infección, creatinina quinasa elevado, LDH elevado, GOT elevado, GPT elevado, bilirrubina elevado, creatinina elevado, urea elevado, lact, número plaquetas elevado, sodio elevado, tiempo de Quick, fibrinógeno, hemoglobina bajo, infección con E.coli, uso de vasopresores.

El análisis univariado de la covariable vasoconstrictor

Como se mencionó anteriormente, en Krieg *et al.* (2014), se propone un modo de elegir un subgrupo de las 35 variables originalmente consideradas para predecir la mortalidad a través del $p-$valor del test de t o del test exacto de Fischer en una instancia de análisis univariado. En el caso continuo el test compara las medias de los dos grupos considerados (vivos y muertos) en la variable mientras que en el caso discreto se compara la distribución de ambos grupos a través del test de Fischer. Para una variable categórica, que es el caso de la variable 'vasoconstrictor' que se analizará en este apartado, se testea: $H_0 : OR = 1 vs H_1 = OR \neq 1$.

Grupo \ Condición	Presente	Ausente
Grupo 1	a	b
Grupo 2	c	d

Tabla 9.1: *Ejemplo de tabla de contingencia usada para el cálculo del odds ratio.*

mortalidad \ vasoconstrictor	Presente	Ausente
Si	20	1
No	25	18

Tabla 9.2: *Tabla de contingencia para la variable vasoconstrictor.*

Krieg *et al.* (2014) proponen la inclusión de dicha variable en un análisis de regresión logística múltiple únicamente si el $p-$valor del test es menor a 0.1. Esto no es lo que se propone en esta tesis ya que en el análisis de regresión logística múltiple se incluirán las 35 covariables potencialmente predictoras para la mortalidad. A pesar de esto último, se ha elegido la variable 'vasoconstrictor' con el fin de realizar un análisis detallado de su posible contribución como predictor de mortalidad tal como fue hecho en Krieg *et al.* (2014) y comparar posteriormente, este resultado con el obtenido cuando se considera una selección de variables basada en regresión logística múltiple. 'Vasoconstrictor' es categórica y determina si se le suministró al paciente una medicación de vasodilatadores o no. Analogando el formato de una Tabla de contingencia (ver Tabla 9.1) es posible encontrar el valor del Odd Ratio (razón de chances) para esta variable usando R . Para el caso de 'vasoconstrictor' , se presentan los valores obtenidos en la Tabla 9.2. En general, para un tabla como la indicada en la Tabla 9.1, se puede estimar la razón de chances OR como

$$\widehat{OR} = \frac{ad}{bc}.$$

Por otro lado, se vio en el Capítulo 7 que la distribución de $\log(OR)$ es asintóticamente normal donde su desvío asintótico puede ser estimado por

$$SD = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}.$$

Gracias a esto es posible construir un intervalo de confianza de nivel 95% estimado para el logaritmo del estimador de la razón de chances y transformarlo con la función exponencial a la escala de la razón de chances. El intervalo de confianza de nivel 95% viene dado

entonces por la fórmula $\log(\widehat{OR}) \pm 1.96\, SD$. Para el conjunto de datos analizados se obtiene: $IC = [1.77, 117.3]$. El resultado puede interpretarse de la siguiente manera: las personas que han sido suministradas con vasodilatadores tienen entre 1.77 y 117.3 veces más chances de pertenecer al grupo de no sobrevivientes que las personas que no han sido suministradas con esta medicación. La función $fischer.test$ permite además obtener el $p-$valor del test de Fischer que testea la hipótesis nula de independencia entre filas y columnas en un cuadro de contingencia con marginales fijos. En un cuadro de 2×2, esto es equivalente a testear $OR = 1$. El siguiente código permite su cálculo.

```
# Calculo del odds ratio de vasoconstrictor

tmp1<-dat.roughfix$vasopresor ==1
tmp2<-dat.roughfix$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
fisher.ex(a,b,c,d,0.05)
[1] 1.440000e+01 1.767324e+00 1.173299e+02 2.799018e-03
# la funcion fischer.ex devuelve:
# El odds ratio
# los limites del intervalo de confianza
# el p-valor del test de Fischer en ese orden
```

Se tiene por ejemplo que OR=14.4 y que el $p-$valor obtenido a partir del test exacto de Fisher es 0.003. Esto indica, en principio, que la variable puede ser considerada clínicamente importante para el análisis realizado si utilizamos el criterio considerado en Krieg *et al.* (2014), aunque como ya se ha dicho, el $p-$valor obtenido en el análisis univariado no ha de ser un factor determinante para su uso en el análisis de regresión logística múltiple. Aún así, puede ser interesante realizar un análisis de sensibilidad respecto de la covariable para aseverar cierta información. Con este fin se realizan 10000 réplicas bootstrap no paramétricas de la razón de chances para la variable en cuestión. El código de R que sigue debe evitar los casos en que un elemento de la Tabla de contingencia generado sea 0. En ese caso, el cálculo de la razón de chances no debe realizarse pues la teoría previamente mencionada no es adecuada. En el ejemplo, se han decidido eliminar las muestras bootstrap que contengan ceros en las celdas de la Tabla de contingencia aunque es usual sumar un valor pequeño (0.5 por ejemplo) en la celda que contiene este valor problemático. Esta propuesta alternativa se aplica en un próximo caso.

```
dat.roughfix <- na.roughfix(datos)
```

```
# bootstrap de un odds ratio (vassoconstrictor)

boot.fun.vaso<-function(data,ind) { boot.fix.data <- data[ind, ]
tmp1<-boot.fix.data$vasopresor==1
tmp2<-boot.fix.data$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
while( a==0 || b==0 || c==0 || d==0) {
# Se descartan los bootstrap para los cuales
# una celda de la matriz de 2x2
# es 0.
muestra<-sample(1:64,replace=TRUE)
boot.fix.data<-data[muestra,]
tmp1<-boot.fix.data$vasopresor==1
tmp2<-boot.fix.data$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
}
#salida<-c(fisher.ex(a,b,c,d,0.05)[1],fisher.ex(a,b,c,d,0.05)[4])
salida<-fisher.ex(a,b,c,d,0.05)[1]
salida}


boot.res.vaso<-boot(dat.roughfix,boot.fun.vaso,R=10000)
```

La Figura 9.1 exhibe el histograma de dichas réplicas que permite observar su asimetría con colas largas a derecha. Por esta razón, el logaritmo podría ser un buen candidato para normalizar la distibución. Gracias a las replicaciones bootstrap generadas es posible construir intervalos de confianza bootstrap. Una rápida mirada sobre el histograma de la Figura 9.1 descarta el uso de intervalos bootstrap normales y básico aunque en principio no se tiene fundamentos claros para optar por el percentil o el BC_a. Aún así, si el percentil fuese correcto también debería serlo el BC_a pues la corrección automática sólo se realiza si es necesaria. De hecho, se puede ver que en este caso tiene especial interés la corrección por sesgo.

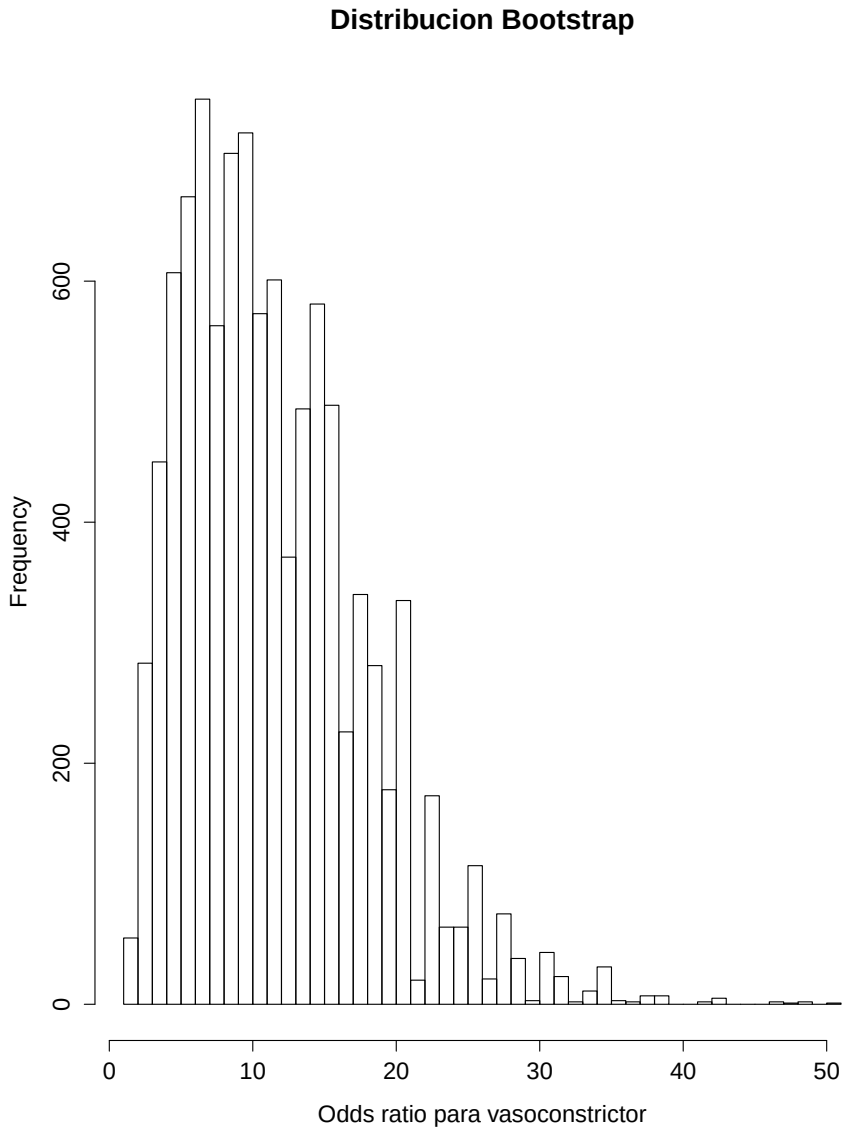


Figura 9.1: *Histograma de las* $B = 10000$ *replicaciones bootstrap de la razón de chances para la variable vasopressors.*

```
ci<-boot.ci(boot.res.vaso)
ci
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 10000 bootstrap replicates

CALL :
boot.ci(boot.out = boot.res.vaso)

Intervals :
Level      Normal              Basic
95%   ( 4.79, 29.80 )   ( 2.36, 25.90 )

Level     Percentile            BCa
95%   ( 2.90, 26.44 )   ( 5.67, 39.18 )
Calculations and Intervals on Original Scale
```

La Figura 9.2 presenta el histograma del logaritmo de las réplicas bootstrap del odd ratio al que se ha superpuesto los límites de los distintos intervalos de confianza. En azul punteado (los límites externos) se han representado los límites del intervalo de la teoría asintótica normal. En verde se han ubicado los límites percentiles, mientras que en azul y con trazo completo se han dibujado los límites del intervalo BC_a. Por otro lado, la línea punteada negra indica la mediana de las réplicas bootstrap mientras que el trazo punteado rojo representa logaritmo del odds ratio calculado con los datos originales. El sesgo, calculado por el parámetro $\widehat{z}_0$ descripto en el Capítulo 3, debe ser positivo lo que explica el corrimiento del intervalo a derecha respecto del intervalo percentil. Todos los intervalos tienen nivel 95% y el de la teoría normal parece ser demasiado amplio obligando al analista a ser escueto respecto de las posibles conclusiones. En definitiva, el método bootstrap parece otorgarle al analista mayor precisión justificando su uso en este caso.

Otro tema pertinente que el bootstrap permite tratar es el análisis de sensibilidad. La técnica de jacknife-after-bootstrap permite detectar datos influyentes que puedan cambiar radicalmente conclusiones. La Figura 9.3 exhibe el gráfico del jacknife para las replicaciones de la covariable 'vasoconstrictor' en donde se destaca el paciente 50. El paciente en cuestión se muestra tan significativo pues su exclusión genera un 0 en la Tabla de contingencia. Se podría estar interesado en conocer los resultados del estudio sin registros del paciente en cuestión. Para ello, se suma 0.5 en la celda que contiene el 0. El paciente 50 tiene registros de fallecimiento y de ausencia de tratamiento por vasodilatadores. Por ejemplo, el $p-$valor

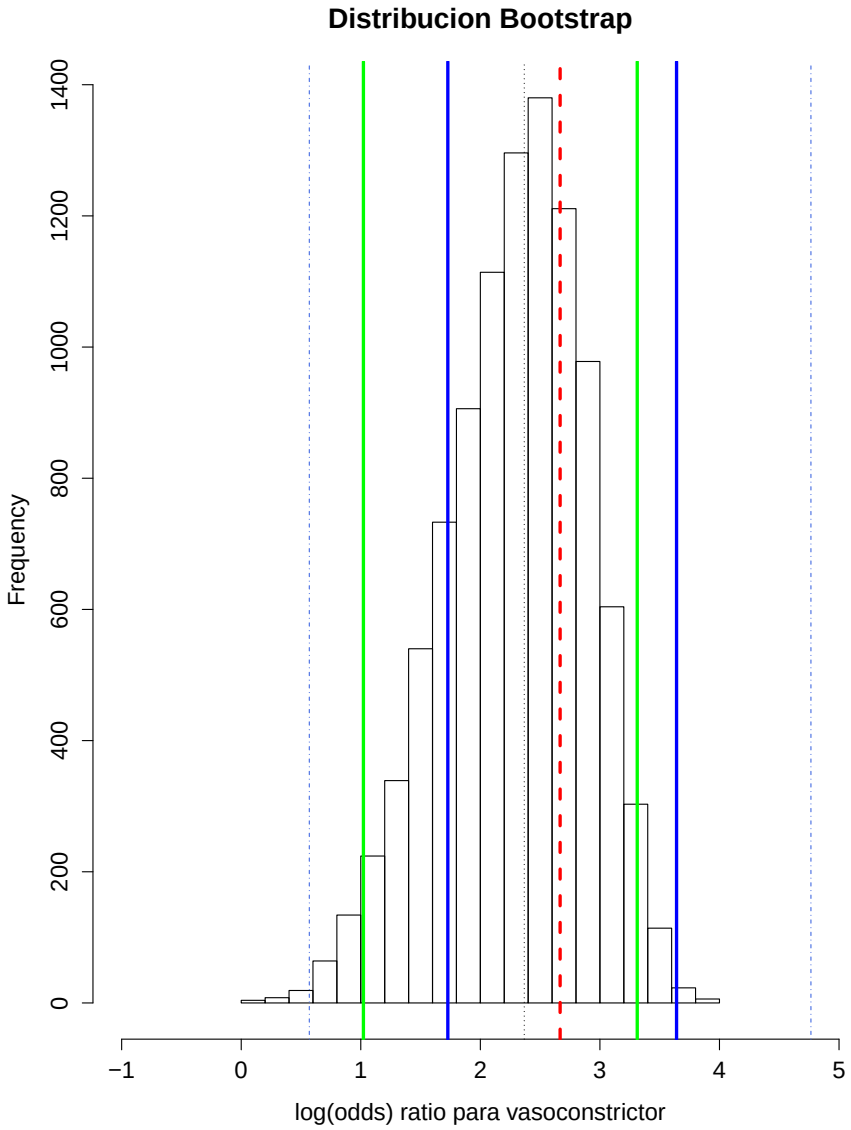


Figura 9.2: *Histograma del logaritmo las* $B = 10000$ *réplicas bootstrap de la razón de chances para la variable vasopressors. En azul punteado (los límites externos) se han representado los límites del intervalo de la teoría asintótica normal. En verde se han ubicado los límites percentiles mientras que en azul y con trazo completo se han dibujado los límites del intervalo* BC_a*. Por otro lado, la línea punteada negra indica la mediana de las observaciones mientras que el trazo punteado rojo representa el valor observado originalmente.*

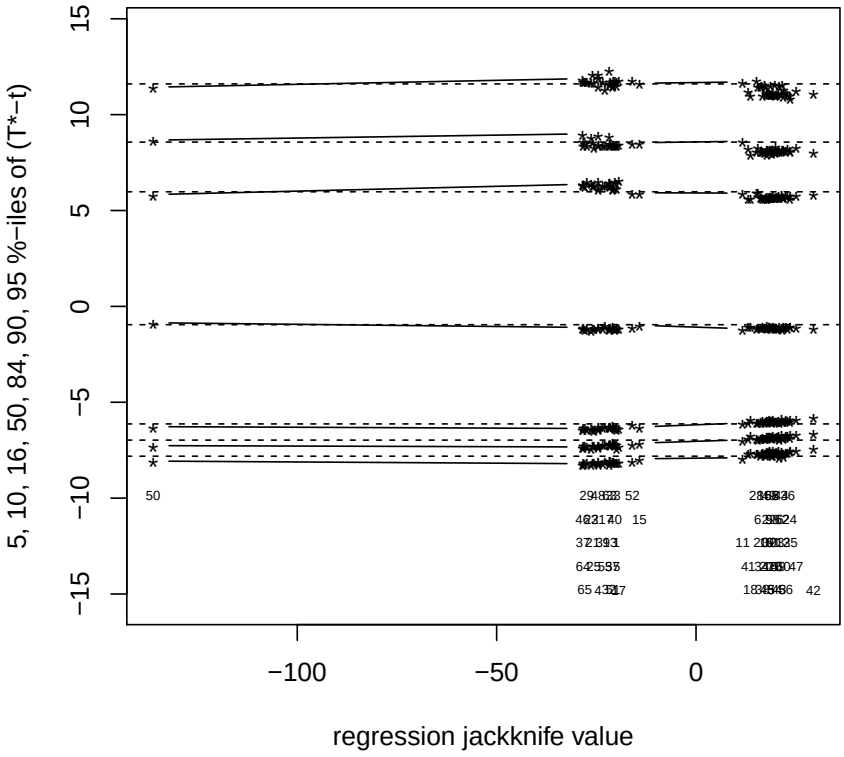


Figura 9.3: *Jacnkife-after-bootstrap de las* $B = 10000$ *replicaciones bootstrap de la razón de chances para la covariable vasopressors.*

del test exacto de Fisher pasa a ser 0.00056 mucho menor que el $p-$valor con todos los datos. Si bien, el resultado obtenido no cambia la importancia clínica de esta variable para el análisis univariado propuesto en Krieg *et al.* (2014). La decisión de eliminar un dato del conjunto de n=64 pacientes sí resultaría pertinente en el caso de la variable continua 'fibrógeno' cuyo estudio se ha realizado y no se presenta en esta tesis. Para dicha variable, el paciente indicado como #2 resulta un dato influyente y su exclusión produce un cambio del $p-$valor del test de t de 0.10 a 0.03 lo que resultaría determinante a nivel $\alpha = 0.05$. Esto último se subraya para mostrar la fragilidad de un análisis basado solo en considerar las variables individualmente y no en forma conjunta.

El análisis de regresión logística múltiple

Una vez realizado el screening, es decir, el análisis exploratorio univariado, proceso que se detalló para la variable 'vascoconstrictor' y que sólo tuvo un interés descriptivo (no se han eliminado ninguna de las 35 variables hasta el momento, a diferencia de lo realizado en Krieg et al (2014)), se prosigue con el método de selección de variables en el esquema de un análisis de regresión logística múltiple. El análisis de sensibilidad realizado previamente ha permitido distinguir distintos posibles modelos con la exclusión de algunos pacientes que se han visto muy influyentes. Aún así, una vez aplicado el método de selección de variables mediante el criterio de Akaike sobre el total de candidatos, se han obtenido siempre las mismas covariables en el modelo final: entre las 35 originalmente consideradas, las variables seleccionadas corresponden a 'edad', 'sexo', 'diálisis', 'necrosis', 'renal', 'ck', 'ldh', 'got', 'bili', 'na', 'ecoli' y 'diálisis' además de la ordenada a la origen. El método de selección de variables por *backward*, descripto en el Capítulo 6 puede realizarse a traves del siguiente código cuyos resultados se presentan a continuación:

```
# Modelo

m1 <- glm(death~.,
                   data = dat.roughfix, family=binomial)
m2 <- stepAIC(m1,trace=F,direction="backward")
summary(m2)

Call:
glm(formula = death ~ Edad+Sexo+Dialisis+Necrosis+Renal+
                      CK+LDH+GOT+Bilirrubina+Sodio+E.coli,
             family = binomial, data = dat.roughfix)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-7.509e-05  -2.100e-08  -2.100e-08   2.100e-08   7.454e-05

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.422e+01  8.058e+05   0.000    1.000
Edad         1.712e+01  1.285e+04   0.001    0.999
Sexo         2.185e+02  6.762e+04   0.003    0.997
Dialisis    -6.630e+02  4.174e+05  -0.002    0.999
Necrosis     3.788e+02  1.020e+05   0.004    0.997
Renal        8.176e+02  5.630e+05   0.001    0.999
CK           5.984e-02  5.008e+01   0.001    0.999
```

```
LDH          1.207e+00  3.326e+02   0.004    0.997
GOT         -3.645e+00  5.131e+03  -0.001    0.999
Bilirrubina  3.305e+01  1.217e+04   0.003    0.998
Sodio       -1.080e+01  1.275e+04  -0.001    0.999
E.coli      -1.086e+02  2.302e+05   0.000    1.000

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8.1004e+01  on 63  degrees of freedom
Residual deviance: 4.0203e-08  on 52  degrees of freedom
AIC: 24

Number of Fisher Scoring iterations: 25
```

En vista a estos resultados, el modelo resultante no parece ser demasiado significativo respecto de su calidad predictiva en la variable respuesta y sería conveniente utilizar algún nuevo método de selección de variables. En el siguiente apartado se describe el uso de una metodología bootstrap con dicho fin.

Análisis de sensibilidad bootstrap

Se ha realizado además un nuevo análisis de sensibilidad bootstrap aunque en este caso se lo ha hecho sobre el método de selección de variables. Se ha analizado entonces la estabilidad de las variables seleccionadas por el criterio AIC usando 500 muestras bootstrap. El menor número de muestras bootstrap considerado respecto de las $B = 1000$ utilizadas anteriormente responde al mismo criterio que el del Capítulo 8, es decir, el costo computacional que esta metodología implica. A continuación se presenta el código utilizado y en la Tabla 9.3 se muestra la proporción de veces que las covariables son elegidas usando el criterio de Akaike.

```
coef.names <- NULL
for(b in 1:500)
{
  ind <- sample(1:64, replace = T)
  boot.fix.data <- na.roughfix(dat[ind, ])
  tmp.mod.0 <-  glm(death ~ . ,
                    data=boot.fix.data, family=binomial)
```

Ordenada al origen 1.000	Edad 0.206	CK 0.232	Crea 0.124
E.coli 0.128	Urea 0.164	Hipotonía 0.166	LDH 0.484
Necrosis 0.682	renal 0.932	Quick 0.088	Sepsis 0.146
Diálisis 0.350	Leucocitos 0.198	Vasopresor 0.090	Adiposidad 0.064
Ventilación 0.130	Bilirrubina 0.172	Bulla 0.082	BMI 0.106
PCR 0.094	Fiebre 0.290	fibrinógeno 0.094	GOT 0.258
GPU 0.358	Desprendimiento 0.166	Hemo 0.194	Lact 0.070
Sexo 0.198	Sodio 0.182	Tromb 0.050	Taquicardia 0.068

Tabla 9.3: *Proporciones de veces con que las covariables han sido elegidas en el proceso de selección de variables usando el criterio AIC y con 500 muestras bootstrap.*

```
  tmp.mod <- stepAIC(tmp.mod.0, trace=F)
  coef.names <- c(coef.names, names(tmp.mod$coeff))
}

table(coef.names)/500
```

El proceso completo, desde la imputación de datos faltantes a la selección de variables se ha replicado en cada muestra bootstrap. Por último, se han conservado en el modelo sólo las variables elegidas al menos un 60% de las veces. Se destaca en los resultados el caso de la variable 'vasoconstrictor' que, esta variable que resultaba muy significativa en el análisis exploratorio univariado, sólo ha sido elegida un 6.8% de las veces. La Figura 9.4 presenta un estimador de la densidad de las résplicas bootstrap de los odd ratio de la variable 'vasoconstrictor' . Se destaca una moda en 0, que representa el 93% aproximado de las veces en que la variable no fue elegida.

En un segundo análisis, se ha aplicado el mismo procedimiento considerando únicamente las variables elegidas al menos un 15% de las veces en la instancia anterior. Se han eliminado además, del análisis de selección de variables bootstrap, los pacientes con datos faltantes en las covariables elegidas. La variable 'fibrógeno' no se ha tenido en cuenta para el análisis pues presenta más de un 30% de datos faltantes. Se redujo el número total de pacientes a

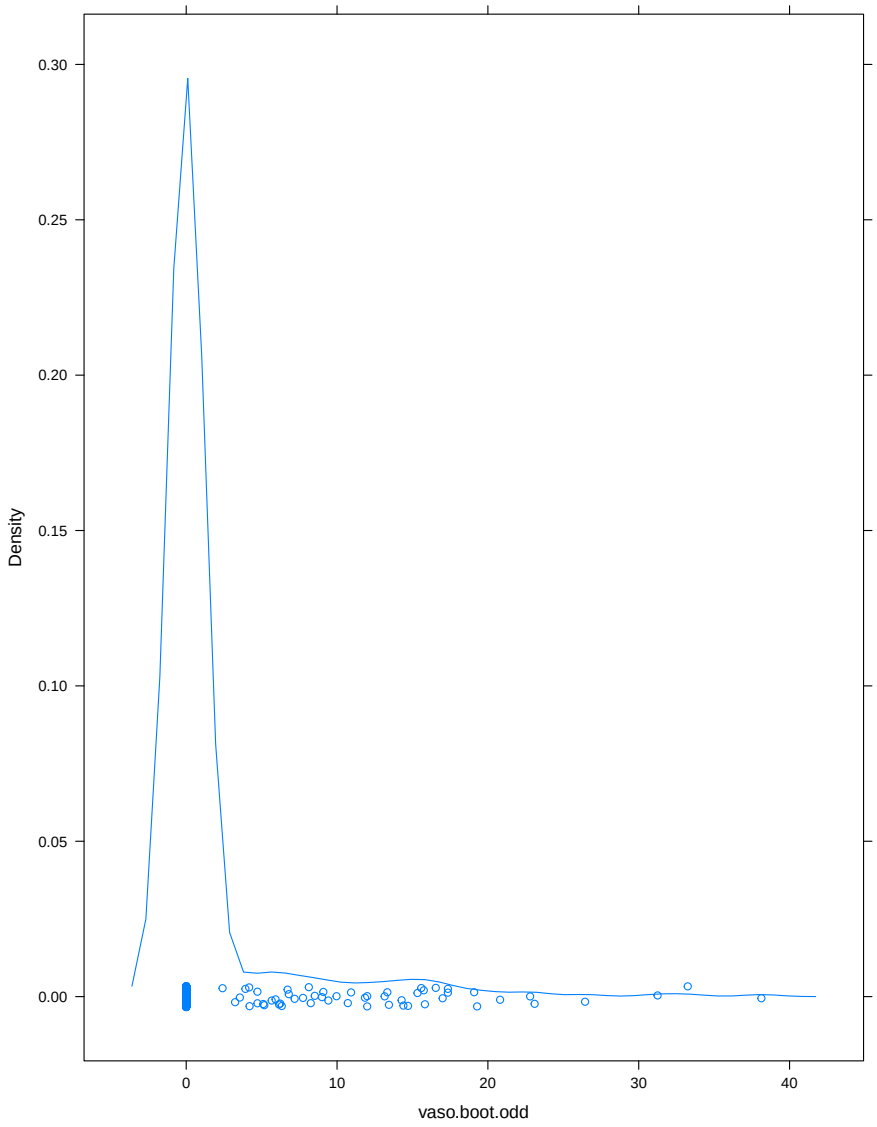


Figura 9.4: *Densidad bootstrap del odds ratio para la variable 'vasoconstrictor' en el modelo de selección.*

Ordenada al origen 1.000	Edad 0.478	CK 0.272	Diálisis 0.586
GPU 0.538	Urea 0.270	Desprendimiento 0.200	Hemo 0.212
Necrosis 0.868	Reanl 0.948	hipotonía 0.226	LDH 0.570
Leucocitos 0.160	Sodio 0.340	Sexo 0.370	

Tabla 9.4: *Proporciones de veces con que las covariables han sido elegidas en el proceso de selección de variables usando el criterio AIC y con 500 muestras bootstrap considerando únicamente pacientes sin datos faltantes y covariables elegidas al menos 15% de las veces en la primera instancia.*

$n = 48$. Las conclusiones obtenidas son análogas ya que las variables elegidas al menos un 60% de las veces son una vez más 'renal' y 'necrosis' (ver Tabla 9.4). Para dichas variables se han realizado los histogramas de las réplicas bootstrap de los coeficientes de regresión del modelo de regresión logística que considera únicamente estas dos covariables. Dichos histogramas se presentan en la Figura 9.5. Los histogramas concentran su peso lejos de 0 lo que es clara evidencia, aún con un tamaño muestral chico y con gran variabilidad, que los factores de riesgo son significativos.

El programa para calcular los intervalos de confianza BC_a de nivel 95% para los coeficientes de regresión y su resultado se presenta a continuación.

```
mf<-glm(death~necrosis+renal,
        data=dat.roughfix,family=binomial)
library(car)
m1.boot <- Boot(mf, R = 2000)
confint(m1.boot)
Bootstrap quantiles, type =  bca

                 2.5 %    97.5 %
(Intercept) -20.3054043 -2.036004
Necrosis     -0.1588904  3.816156
Renal         1.6231304 20.469684
```

El intervalo de confianza para el coeficiente de la variable 'necrosis' en el modelo de regresión logística incluye el valor 0. Esto debilita la calidad predictora para la mortalidad

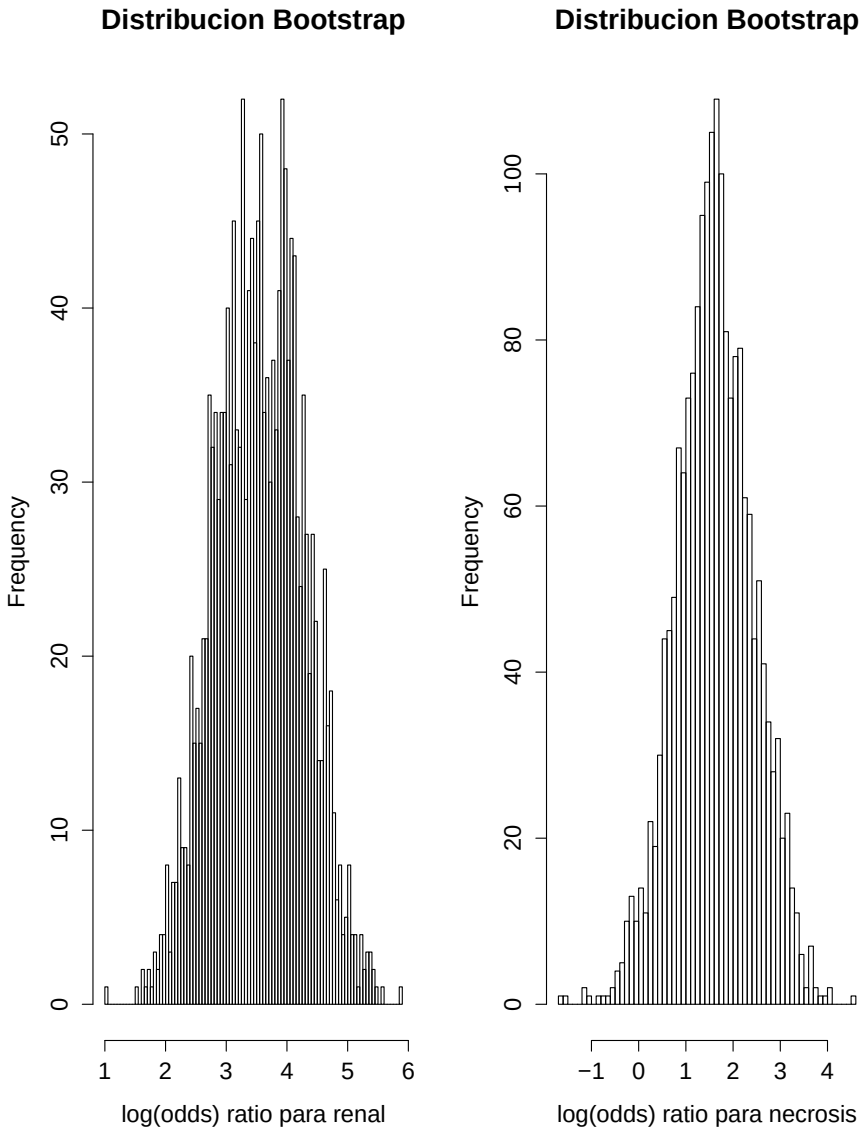


Figura 9.5: *Distribución bootstrap de los coeficientes de regresión para 'renal' y 'necrosis' respectivamente.*

de dicha variable en el ejemplo estudiado. En base a los intervalos obtenidos, se concluiría que la variable 'renal ' es quien aporta mayor información a la variable respuesta y. En Krieg *et al.* (2014) se explica que los resultados respecto a la calidad de un predictor son muy variables de un hospital a otro.

Para el problema en estudio, podría ser interesante analizar el error de predicción (en este caso, de clasificación) cometido al usar únicamente estas dos variables como predictoras. Para ello, se analizó el error bootstrap mejorado propuesto por Efron (1993) (descripto en el Capítulo 7) y el error bootstrap 0.632. Al igual que en el ejemplo de datos de orina de la Sección 7.6 se utilizó la función de costos $c(y, \hat{y}) = I_{\{|y-\hat{y}|>1/2\}}$.

El programa para el cálculo del error aparente, el boostrap mejorado y el 0.632 se encuentran a continuación. Los códigos para la generación de los mismos son equivalentes a los del ejemplo de datos de orina del Capítulo 7.

```
datos<-na.roughfix(datos)
cost <- function(r, pi=0) mean(abs(r-pi)>0.5)
app.err <- cost(datos$death, fitted(mf))
app.err
[1] 0.125
err.boot<-opt+app.err
err.boot
[1] 0.14
err.632 <- 0.368*app.err + 0.632*err.632
err.632
[1] 0.1345387
```

Por otro lado, el histograma de los estimadores bootstrap del optimismo definido en el Capítulo 5 se presenta en la Figura 9.6. Se recuerda que el optimismo es una medida del sesgo del error de predicción aparente propuesto por Efron (1993) y se define como:

$$\omega(F) = E_F[err(z, F) - err(z, \widehat{F})].$$

Esta medida no puede ser calculada explícitamente, por lo que se utiliza el método bootstrap para dar una aproximación de la misma. Las réplicas bootstrap, basadas en $B = 500$ replicaciones, se presentan en el histograma de la Figura 9.6, y están dadas por:

$$\omega(\widehat{F}) = E_{\widehat{F}}[err(z^*, \widehat{F}) - err(z^*, \widehat{F}^*)].$$

Se recuerda que $\widehat{F}^*$ es la función de distribución empírica de la muestra z^*. La dispersión de los resultados en el histograma hace sospechar sobre la precisión del cálculo del error. Aún así, este gráfico permite tener una idea del sesgo del error aparente que parece tender a ser positivo.

Por último, la Figura 9.7 exhibe las componentes de la estimación 0.632 boostrap del error de predicción para $B = 500$ replicaciones al igual que se hizo en el ejemplo de datos de orina del Capítulo 7. Se recuerda la descripción del gráfico que también puede hallarse en el ejemplo 7.6 del capítulo 7. Los boxplot son las cantidades $y_i - \mu(x_i^*, \widehat{F})$, es decir, las cantidades correspondientes al error de clasificación del valor predicho $i-$ésimo obtenido a partir de una muestra bootstrap x^* generada por la distribución $\widehat{F}$. Con líneas punteadas se han dibujado las rectas $y = 0.5$ e $y = -0.5$ que permiten saber que valores han sido mal predichos o que valores contribuyen al error 0.632 bootstrap ya que la función de costos considerada es $c(y, \hat{y}) = I_{\{|y-\hat{y}|>1/2\}}$. El orden de los boxplot en el gráfico es el orden creciente de los residuos del modelo ajustado. Se tiene como simple observación que los casos 50 y 52 así como los casos 21 y 33 están siempre mal clasificados mientras que la mayoría de los boxplot no llegan a cruzar las líneas horizontales punteadas indicando que estos casos han sido siempre bien clasificados. Para saber a que observaciones corresponde cada boxplot se da a continuación los índices de los individuos ordenados por el valor de sus residuos.

```
21 33 52 4 17 25 29 31 40 63 64
1 3 5 13 37 46 48 57 7 9 10 14
15 22 23 24 30 32 34 39 42 43 45
47 51 53 54 55 56 58 59 62 2 6 8
11 20 26 28 44 49 61 12 16 18 19
27 35 36 41 60 38 50
```

Los pacientes 50 y 52 se comportan atípicamente respecto de la categoría a la que pertenecen. Por ejemplo, el paciente 50 ha fallecido pero sus registros son similares al promedio de los registros de personas que no lo han hecho. Exactamente lo opuesto ocurre con el paciente 52. Por otro lado, el comportamiento de los boxplot, mucho mejor en términos de clasificación respecto del ejemplo del Capítulo 7, resalta la eficacia del ajuste propuesto.

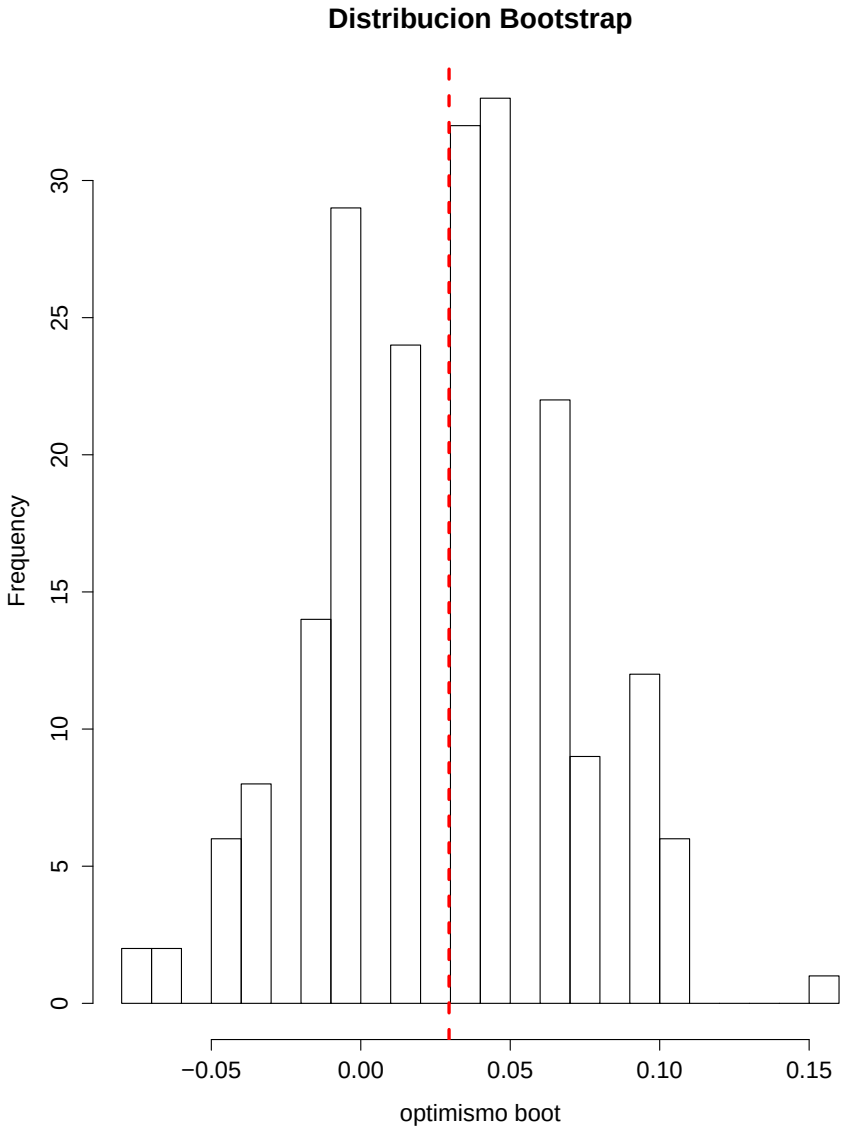


Figura 9.6: *Histograma de* $B = 500$ *replicaciones boostrap del optimismo.*

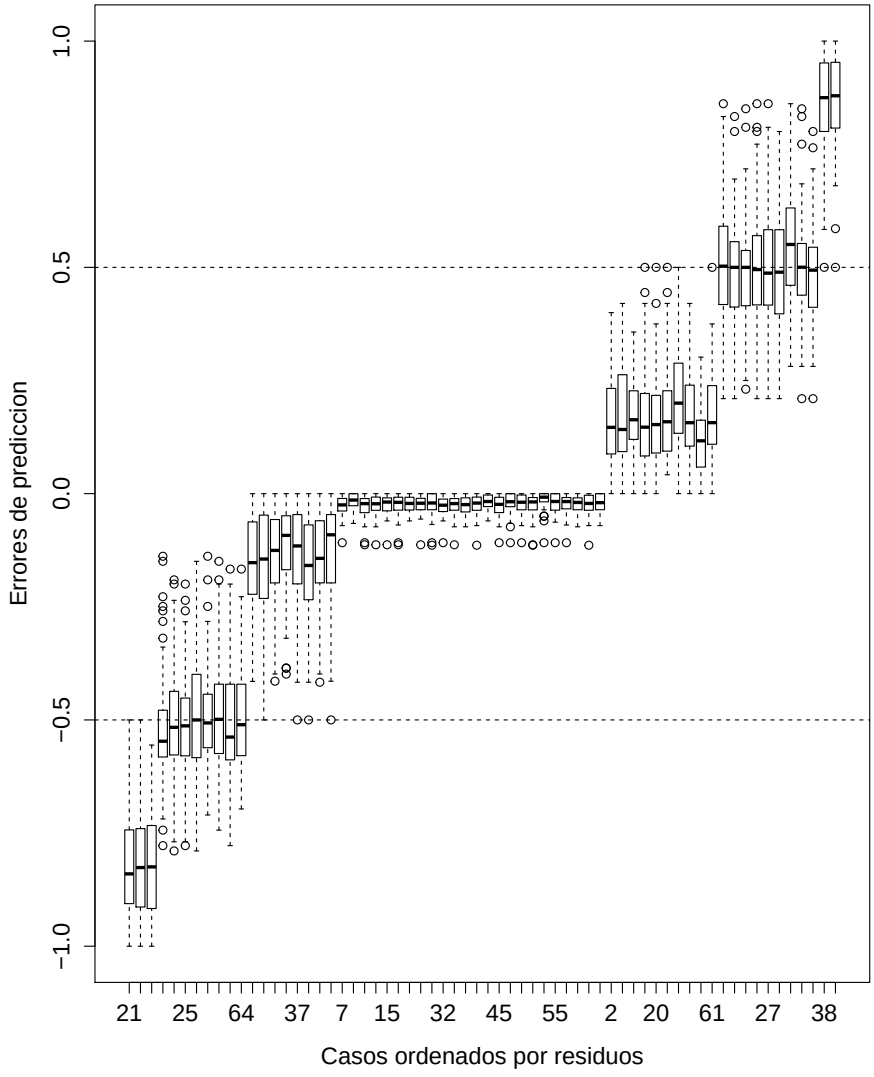


Figura 9.7: *Componentes de la estimación 0.632 bootstrap del error de predicción agregado para B=500 simulaciones. Los boxplot están ordenados en función del orden creciente del valor de los residuos para los pacientes.*

Resumen y discusión

En este capítulo se realiza un breve resumen de este trabajo y se discuten algunos puntos importantes del método bootstrap en el mundo estadístico.

La publicación del primer artículo sobre métodos bootstrap en 1979 por Efron B. fue un evento de suma importancia en la Estadística. Sintetizó las ideas de remuestreo preexistentes y estableció un nuevo marco para el análisis estadístico basado en simulaciones. *Esto no es sólo una cuestión de trabajar más rápidamente o con conjuntos de datos más grandes. El poder computacional ha liberado al estadístico de la lucha con la tratabilidad matemática. Podemos ahora reponder preguntas de real interés para el científico y ya no sólo elegir dentro de un pequeño catálogo de casos que la matemática puede resolver*, comenta Efron (1993).

Una vez superado el escepticismo incial, producto de libros que han detallado los supuestos y características del método, hoy en día, prácticamente todas las disciplinas científicas hacen uso del bootstrap. Como se ha destacado en el capítulo introductorio, las ventajas de la metodología bootstrap son fácilmente apreciables. Quizás, el éxito más considerable repose en la cita previa de Efron: es posible enfrentarse a problemas de índole científica sin necesidad de hacer uso de suposiciones o cálculos teóricos que la matemática sólo es capáz de resolver en pocos casos. En el mismo capítulo se ha destacado a su vez el esquema general de la metodología en donde el cambio de F a $\widehat{F}$ es el artífice principal del accionar bootstrap.

En el Capítulo 3, dedicado a los intervalos de confianza bootstrap, se ha visto la adaptabilidad del método BC_a en 2 ejemplos distintos, su nivel de cobertura exacto en un ejemplo teórico y el nivel de cobertura en un ejemplo de simulación. Se ha analizado la importancia del parámetro a en casos donde la varianza del estimador depende del parámetro de interés. Se ha descripto además el método de jacknife-after-bootstrap que realiza un análisis de sensibilidad respecto de la influencia de datos individuales. El uso de los intervalos de confianza bootstrap descriptos se ha visto facilitado por las librerías del software R que

acumula secciones para el bootstrap. El gran aporte de Davison y Hinkley (1997) con la librería *boot* ha sido excelso en la aplicabilidad del bootstrap y ha mejorado notablemente la librería *bootstrap* creada por Efron (1993).

En el Capítulo 4, se han desarrollado métodos bootstrap para el análisis de los coeficientes de regresión de un modelo de regresión lineal. De esta froma, se ha podido ampliar el espectro del simple esquema bootstrap para el caso univariado. Se han descripto dos modelos de simulación distintos en la generación de replicaciones bootstrap de los coeficientes de regresión, el método de remuestreo por pares y de remuestreo por residuos, y se los ha aplicado en diversos ejemplos a través de dos librerías distintas de R, *boot* y *car*. Se ha hablado de las diferencias entre los métodos y se ha distinguido la fragilidad del método de remuestreo por residuos en situaciones de heteroscedasticidad. Para dichas situaciones se ha descripto a su vez el método wild bootstrap. Aún así, un ejemplo de datos reales, el ejemplo del 6MWT del Capítulo 4, ha mostrado un comportamiento similar en los dos primeros métodos analizados, aún con pocos datos.

El Capítulo 5 ha hecho una breve revisión del problema de predicción y ha mostrado un método bootstrap para el cálculo del error de predicción. Se han comparado el método de Validación Cruzada y el método Bootstrap, Efron (1993), en un ejemplo extraído de la librería *boot* de R. Los resultados se han visto similares en el ejemplo de datos de la central nuclear de la Sección 6.5 para el problema de selección de variables. Se han podido observar además las dificultades del uso del método bootstrap en un caso particular: efectivamente, para el caso de la central nuclear, la naturaleza desbalanceada de las covariables, en conjunción con la naturaleza binaria de alguna de ellas, han propiciado frecuentemente diseños singulares. Esto puede ser un problema infranqueable como se ha visto en el mismo ejemplo para el cual modelos con más de 5 covariables no eran identificables.

En el Capítulo 7, se ha hecho una breve revisión del modelo de regresión logística, las posibles definiciones de residuos y de técnicas de remuestreo. Se ha analizado además un ejemplo de categorización con la metodología bootstrap mediante el uso del error bootstrap 0.632 y se ha estudiado la distribución bootstrap de la desviación con 3 técnicas distintas de remuestreo en el ejemplo de brotes enfermos en cañas de azúcar de la Sección 7.4. Los resultados en el caso de categorización permiten tener una noción del error cometido en el ajuste del modelo mientras que se ha visto que la desviación juega en rol importante en la precisión del ajuste del modelo.

Por último, en los Capítulos 8 y 9, se han llevado el global de las técnicas analizadas en los capítulos previos a la práctica. En el Capítulo 8 se ha analizado un conjunto

de datos de anemia mediante el uso de modelo de regresión lineal mientras que en el Capítulo 9 se ha analizado otro conjunto de datos correspondientes a infecciones de tejidos blandos necrotizantes a través del uso de un modelo de regresión logística. Se han aplicado intervalos de confianza, se han realizado selecciones de variables, y se han calculado errores de predicción. El ejemplo ha querido subrayar que la aplicación de los métodos en un ejemplo real puede ser más complicado, menos trivial que en pequeños ejemplos teóricos propuestos por sus bondades.

En definitiva, surge una pregunta evidente a esta altura: cuáles son las características atrayentes del bootstrap? El bootstrap permite al analista estudiar la precisión estadística de procedimientos complicados haciendo uso del poder computacional. El uso del bootstrap exime al analista de complicados cálculos teóricos y ofrece a veces respuestas cuando ningún cálculo analítico las ofrece. El bootstrap puede ser usado paramétricamente o no paramétricamente. En el modo no paramétrico, se evitan restrictivos y a veces peligrosos supuestos paramétricos sobre la forma de la distribución poblacional. El modo paramétrico puede proveer estimaciones más precisas como en el caso del error en comparación con el *método de información de base de Fischer* (Efron 1993).

Capítulo 10

Apéndice

10.1 Códigos

10.1.1 todo.R

```
#cap2
###### media
###### (uso de la funcion sample())

mvo2 <- c(62.9, 57, 56.5, 51, 43.3, 61, 61.5, 52.9,
          45.9, 60, 58.6, 63, 56.6, 50.5, 57, 50.5,
          63.8, 53.8,63.2, 62.8, 63.7,
          58.8, 40, 58.1)

set.seed(123); m <- 5000; b.res.1 <- numeric(m)
for(i in 1:m)
{
  b.res.1[i] <- mean(sample(mvo2, replace=T))
}

mean(b.res.1 - mean(mvo2))
sd(b.res.1)

##### mediana

m <- 5000; b.res.1 <- numeric(m)
for(i in 1:m)
{
  b.res.1[i] <- median(sample(mvo2, replace=T))
}

mean(b.res.1 - median(mvo2))
```

```
sd(b.res.1)

median(mvo2)
a<-dnorm(0,0,sd(mvo2))

# desvio asintotico de la mediana
# suponiendo normalidad en los datos
n<-24
(sd(mvo2)*sqrt(2*pi))/(2*sqrt(n))

##### cociente de variacion

m <- 5000; b.res.1 <- numeric(m);
cv<-sd(mvo2)/mean(mvo2)
for(i in 1:m)
{
  b.res.1[i] <- sd(sample(mvo2, replace=T))/
    mean(sample(mvo2, replace=T))
}

sd(b.res.1)

#### graficos de densidad

library("lattice")


densityplot(b.res.1,xlab="cv.boot")



#### discreteness (ejemplo: distribucion discreta
#### del bootstrap de la mediana)

m <- 5000; b.res.1 <- numeric(m);
for(i in 1:m)
{
  b.res.1[i] <- median(sample(mvo2, replace=T))
}
hist(b.res.1,breaks=80,main="bootstrap no parametrico",
     xlab="mediana.boot")


b.res.2 <- numeric(m);
for(i in 1:m)
{
  b.res.2[i] <- median(sample(mvo2, replace=T)
                       +rnorm(n=24)*.5)
}
hist(b.res.2,breaks=80,main="bootstrap suavizado",
     xlab="mediana.boot")


##### secuencia aleatoria falsa?


Nboot<-2000;long<-numeric(Nboot);sw<-numeric(Nboot)
for (i in 1:Nboot){
```

```
  tmp<-rbinom(100,1,0.5)
  cont<-1
  cont2<-0
  maximo<-numeric(100)
  for (j in 1:99){
    if (tmp[j+1]==tmp[j]){
      cont<-cont+1; maximo[j]<-cont;
      cont2<-cont2
    }
    if (tmp[j+1]!=tmp[j]){
      cont<-1 ;maximo[j]<-cont ;
      cont2<-cont2+1
    }
  }
  long[i]<-max(maximo)
  sw[i]<-cont2
}
plot(jitter(long),jitter(sw),xlab="corrida mas larga",
     ylab="numero de cambios",pch=".")
text(8,43,1,col="red",cex=1.5)
text(4,52,2,col="red",cex=1.5)

#### chequeo de la veracidad
#### de las secuencias

mean(sw<43)
mean(sw<52)
mean(long<8)
mean(long<4)

##########################################################################
##########################################################################
#cap3


#### ejemplo normal para introducir int conf normales


library(boot)

mvo2 <- c(62.9, 57, 56.5, 51, 43.3, 61, 61.5, 52.9,
          45.9, 60, 58.6, 63, 56.6, 50.5, 57, 50.5,
          63.8, 53.8,63.2, 62.8, 63.7, 58.8,
          40, 58.1)

boot.fun<-function(data,ind) mean(data[ind])
boot.media<-boot(mvo2,boot.fun,R=1000)
hist(boot.media$t,breaks=20,main="",
     xlab="media bootstrap")
ci<-boot.ci(boot.media,conf=0.95)

abline(v=ci$normal[2],col='red')
abline(v=ci$normal[3],col='red')
abline(v=ci$perc[4],col='blue')
abline(v=ci$perc[5],col='blue')
abline(v=boot.media$t0,lty=3)
```

```
#####################################################################
##### Datos espaciales

library(boot)
library(bootstrap)
library(MASS)
attach(spatial)

# estadistico

n<-length(A)
var_hat<-(1/n)*sum((A-mean(A))^2)


# caso no parametrico

boot_var<-function(data, ind) (1/n)*sum((data[ind]-mean(data[ind]))^2)
boot_1<-boot(A,boot_var,R=1000)
A.boot<-(1/1000)*sum((boot_1$t-mean(boot_1$t))^2)
hist(boot_1$t,main="",xlab="bootstrap",
     breaks=40)
abline(v=var_hat,col="red",lwd=3)
ci<-boot.ci(boot_1,conf=0.9)
abline(v=ci$bca[c(4,5)],col="blue",lty=3,lwd=3)
abline(v=ci$perc[c(4,5)],col="green",lty=3,lwd=3)

#caso parametrico


# considero el siguiente modelo ajustado
mu<-c(mean(A),mean(B))
sigma<-(1/26)*matrix(c(sum((A-mean(A))^2),sum((A-mean(A))*
                                                  (B-mean(B))),sum((A-mean(A))*(B-↩
                                                      mean(B))),
                       sum((B-mean(B))^2)),2,2,byrow=T)
r.gen<-function(data,mle){
  out<-data
  out$A<-mvrnorm(n,mle,sigma)[,1]
  out$A
}
Aboot<-boot(A,boot_var,1000,sim="parametric",
            ran.gen=r.gen,mle=mu)
boot.A<-(1/1000)*sum((Aboot$t-mean(Aboot$t))^2)
hist(Aboot$t,main="",xlab="parametric bootstrap",
     breaks=20)
abline(v=var_hat,col="red",lwd=3)
boot.ci(Aboot,conf=0.90,type=c("norm","perc"))

# Para el intervalo BCa
# boot.ci exige valores de influencia

U <- empinf(data = A,statistic = boot_var,type = "jack",
            stype="reg")

ci.90 <- boot.ci(Aboot, conf=0.90,
                 type=c("norm", "perc", "bca"), L = U)
ci.90
hist(Aboot$t, breaks = 90, xlab="boot parametrico",main="")
```

```
abline(v = ci.90$bca[c(4,5)], lwd =2, col = "blue")

# intervalo exacto usando teoria normal
#comparacion exacto vs BCa

v<-(1/n)*sum((A-mean(A))^2)
upper<-n*v/(qchisq(0.05,n-1))
lower<-n*v/(qchisq(0.95,n-1))
IC_exacto<-c(lower,upper)

abline(v = IC_exacto, lwd =2, col = "green")


##########################################################################

###### datos de leyes (ejemplo de correlacion)

library(boot)
library(bootstrap)
LSAT<-c(622,542,579,653,606,576,620,615,
        553,607,558,596,635,581,661,547,
        599,646,622,611,546,614,628,575,662,
        627,608,632,587,581,605,704,477,591,578,
        572,615,606,603,535,595,575,573,644,545,645,
        651,562,609,555,586,580,594,594,560,641,512,
        631,597,621,617,637,572,610,562,635,614,546,
        698,606,570,570,605,565,686,608,595,590,558,
        611,564,575)
GPA<-c(323,283,324,312,309,339,310,340,
       297,291,311,324,330,322,343,291,
       323,347,315,333,299,319,303,301,
       339,341,304,329,316,317,313,336,
       257,302,303,288,337,320,323,298,
       311,292,285,338,276,327,336,319,
       317,300,311,307,296,305,293,328,
       301,321,332,324,303,333,308,313,
       301,330,315,282,320,344,301,292,
       345,315,350,316,319,315,281,316,
       302,274)

boot.cor<-function(data,ind) cor(data[ind,])[1,2]
cor.boot_15<-boot(law,boot.cor,R=10000)
hist(cor.boot_15$t,breaks=20,main="",
     xlab="non-param bootstrap",
     main="muestra aleatoria")
ci<-boot.ci(cor.boot_15,conf=0.95)
ci$normal
abline(v=ci$normal[2],col='red')
abline(v=ci$normal[3],col='red')
abline(v=ci$bca[4],col='blue')
abline(v=ci$bca[5],col='blue')

# ahora con toda la poblacion

cor.boot<-boot(law82,boot.cor,R=10000)
hist(cor.boot$t,breaks=20,main="",
     xlab="non-param bootstrap",
     main="all data")
```

```
ci2<-boot.ci(cor.boot,conf=0.95)
abline(v=ci2$normal[2],col='red')
abline(v=ci2$normal[3],col='red')
abline(v=ci2$bca[4],col='blue')
abline(v=ci2$bca[5],col='blue')


#### por que mejora el BCa?
#### Existencia de un sesgo

boot.cor<-function(data,ind) cor(data[ind,])[1,2]
boot.res<-boot(law,boot.cor,R=999)
hist(boot.res$t-cor(law)[1,2],breaks=20,
     xlab="hat.rho*-hat.rho",main="")
med<-median(boot.res$t-cor(law)[1,2])
abline(v=med,lty=2,col="red")
abline(v=0,col="blue")
pp<-mean(boot.res$t-cor(law)[1,2]<0) # porcentaje de datos
#inferiores a hat.rho
# existe entonces un sesgo que
#el intervalo BCa puede corregir
boot.ci(boot.res,conf=0.9)


#########################################################################

## jacknife after bootstrap

library(boot)
library(bootstrap)
law.boot <- boot(law, corr, R=999, stype="w")
law.L <- empinf(data=law, statistic=corr)
split.screen(c(1,2))
screen(1)
split.screen(c(2,1))
screen(4)
attach(law)
plot(LSAT,GPA,type="n")
text(LSAT,GPA,round(law.L,2))
screen(3)
plot(LSAT,GPA,type="n")
text(LSAT,GPA,1:nrow(law))
screen(2)
jack.after.boot(boot.out=law.boot,useJ=F,stinf=F, L=law.L)
#10*(corr(law)-corr(ducks[-7,]))

par(mfrow=c(1,1))




#########################################################################
#########################################################################
#cap4

###### simulacion de datos (regresion lineal)
```

```
library(boot)
library(MASS)
library(arm)
alfa<-0.05  ## nivel
x<-runif(10,-1,1) ## covariable aleatoria uniforme
n<-length(x)
X<-cbind(rep(1,10),x) ## matriz de diseno
y<-2*x+rnorm(10,0,0.5^2)
plot(x,y)
ajuste<-lm(y~x)
abline(ajuste,col="blue",lwd=2)

# elementos del ajuste
b<-ajuste$coefficients
rs<-ajuste$residuals
s<-sqrt(sum(rs^2)/(n-2))

#matriz de proyeccion
P<-X%*%solve(t(X)%*%X)%*%t(X)
I<-diag(rep(1,10))

#varianza real
var.b<-solve(t(X)%*%X)

# distribuci n de hat(beta)<-N(beta,sigma^2(X'X)^-1)

# IC(beta1)<-hat(beta1)+-t_(n-2,alfa/2)*s*sqrt(d11)
IC_b0<-c(b[1]-qt(1-alfa/2,n-2)*s*sqrt(var.b[1,1]),
         b[1]+qt(1-alfa/2,n-2)*s*sqrt(var.b[1,1]))

IC_b1<-c(b[2]-qt(1-alfa/2,n-2)*s*sqrt(var.b[2,2]),
         b[2]+qt(1-alfa/2,n-2)*s*sqrt(var.b[2,2]))

# bootstrap no parametrico
Nboot<-1000
boot.fun<-function(rs,i){
  y.boot<-b[1]+b[2]*x+rs[i]
  betaboot.0<-lm(y.boot~x)$coefficients[1]
  betaboot.1<-lm(y.boot~x)$coefficients[2]
  #cbind(betaboot.0,betaboot.1)
  betaboot.0
}
boot.res<-boot(rs,boot.fun,R=1000)

#para betaboot.0

hist(boot.res$t[,1],main="",xlab="boot.beta0",
     breaks=90,freq=FALSE)
abline(v=b[1],col="red",lwd=3)
fitted<-fitdistr(boot.res$t[,1],"normal")$estimate
curve(dnorm(x,fitted[1],fitted[2]),col="blue",
      lwd=2,add=TRUE)
curve(dnorm(x,b[1],0.5^2*(solve(t(X)%*%X)[1,1])),
      col="green",add=TRUE)

# parametrico

Nboot<-1000
```

```
betaboot.0<-numeric(Nboot)
betaboot.1<-numeric(Nboot)
for (i in 1:Nboot){
  rs2<-mvrnorm(1,rep(0,10),0.5^2*I)
  y.boot<-b[1]+b[2]*x+rs2
  betaboot.0[i]<-lm(y.boot~x)$coefficients[1]
  betaboot.1[i]<-lm(y.boot~x)$coefficients[2]
}

#betaboot.0

hist(betaboot.0,breaks=90,freq=FALSE)
abline(v=b[1],col="red",lwd=3)
fitted<-fitdistr(betaboot.0,"normal")$estimate
curve(dnorm(x,fitted[1],fitted[2]),col="blue",lwd=2,add=TRUE)
curve(dnorm(x,b[1],0.5^2*(solve(t(X)%*%X)[1,1])),col="green",add=TRUE)




################################################################################



##### 6MWT

library(boot)
library(car)
library(ggplot2)
library(MASS)

dat<-read.table("dat6MWT",sep=",")
names(dat) <- c("x.6MWD", "peakVO2")

head(dat)

# Boostrapping and visualization model variability
#index <- sample(1:102, size = 30) # N = 30

#dat2 <- dat[index, ]

m1 <- lm(peakVO2~x.6MWD, data = dat)
#m1 <- lm(peakVO2~x.6MWD, data = dat2)

summary(m1)

plot(dat, xlim = c(80, 800), ylim = c(0, 40))
abline(m1, lwd = 2, col = "blue")

m1.boot <- Boot(m1, R = 1000)
m1.boot.residuals <- Boot(m1, R = 1000, method = "residual")

m1.boot$t[1:4, ]

# pares
a1 <- m1.boot$t[, "(Intercept)"]
b1 <- m1.boot$t[, "x.6MWD"]
```

```
# residauls
a1.r <- m1.boot.residuals$t[, "(Intercept)"]
b1.r <- m1.boot.residuals$t[, "x.6MWD"]

# pares
for(i in 1:20)
{
  curve(a1[i] + b1[i]*x, from = 100, to = 750, add = TRUE,
        col = "red", lty =2)
}

# residuals
for(i in 1:20)
{
  curve(a1.r[i] + b1.r[i]*x, from = 100, to = 750, add = TRUE,
        col = "green", lty =2)
}


summary(m1.boot)
confint(m1.boot)



# bootstrap distributions for slope

hist(b1, breaks = 50, prob = TRUE, main = "Distribucion Bootstrap")
lines(density(b1), col = "red")
lines(density(b1.r), col = "green")


##############################################################################
##############################################################################
#cap5


#### anemia en pacientes mayores
library(xtable)
library(MASS)
library(PASWR)
library(arm)
library(car)
library(randomForest)
library(coefplot)


# Read data
dat <- read.table("dat_anemia.txt")
summary(dat)


#Exploratory analysis for continues variables

attach(dat)

# se escalan y centran
# las variables continuas
```

```
X<-cbind(age,ln.WBC_Admit,ln.CRP_Admit,ln.crea)
X2<-scale(X)
datos<-cbind(Hgb_Admit,sex,X2)
dat2<-data.frame(datos)

# graficos de pares

scatterplotMatrix( ~ Hgb_Admit + age + WBC_Admit +
                     CRP_Admit + Crea_pre, data = dat)


scatterplotMatrix( ~ Hgb_Admit + age + ln.WBC_Admit
                   + ln.CRP_Admit + ln.crea, data = dat)


# Analisis solo con variables continuas

model.anem1 <- lm(Hgb_Admit ~ sex + age +
                    ln.WBC_Admit + ln.CRP_Admit + ln.crea,
                  data = dat)

summary(model.anem1)

model.anem2<-lm(Hgb_Admit ~ sex + age +
                  ln.WBC_Admit + ln.CRP_Admit + ln.crea,
                data = dat2)

summary(model.anem2)

# int de confianza

coefplot(model.anem1, color= "blue",zeroColor = "black",
         intercept = FALSE)+ theme_bw()

coefplot(model.anem2, color= "blue",zeroColor = "black",
         intercept = FALSE)+ theme_bw()

# diagnostics
par(mfrow = c(2,2))
plot(model.anem1)
par(mfrow =c (1,1))

m1.boot <- Boot(model.anem1, R = 1000)
m2.boot <- Boot(model.anem2, R = 1000)
m1.boot.residuals <- Boot(model.anem1, R = 1000,
                          method = "residual")
hist(m1.boot$t[,4], breaks = 50, prob = TRUE,
     xlab="log(leucocitos)",
     main = "Distribucion Bootstrap")
hist(m2.boot$t[,4], breaks = 50, prob = TRUE,
     xlab="log(leucocitos)",
     main = "Distribucion Bootstrap")
summary(m1.boot)
confint(m1.boot)
confint(m2.boot)
b0<-m2.boot$t[,1]
b1<-m2.boot$t[,2]
b2<-m2.boot$t[,3]
```

```
b3<-m2.boot$t[,4]
b4<-m2.boot$t[,5]
b5<-m2.boot$t[,6]

# bootstrap distributions para coef de regresion

hist(b1, breaks = 50, prob = TRUE, xlab="densidades",
     main = "Distribucion Bootstrap",
     xlim=c(min(c(b1,b2,b3,b4,b5)),
            max(c(b1,b2,b3,b4,b5))),ylim=c(0,3.5))
lines(density(b1), col = "red")
lines(density(b2), col = "green")
lines(density(b3), col = "blue")
lines(density(b4), col = "black")
lines(density(b5), col = "violet")


##############################################################################

#### datos cemento

library(MASS)
library(boot)
x1<-c(7,1,11,11,7,11,3,1,2,21,1,11,10)
x2<-c(26,29,56,31,52,55,71,31,54,47,40,66,68)
x3<-c(6,15,8,8,6,9,17,22,18,4,23,9,8)
x4<-c(60,52,20,47,33,22,6,44,22,26,34,12,12)
y<-c(78.5,74.3,104.3,87.6,95.9,109.2,
     102.7,72.5,93.1,115.9,83.8,113.3,109.4)
T<-cbind(x1,x2,x3,x4,y)
X<-cbind(rep(1,length(x1)),x1,x2,x3,x4)
cemento<-data.frame(cbind(y,x1,x2,x3,x4))
colnames(cemento)<-c("y","x1","x2","x3","x4")
eigen(t(X)%*%X)$values
ajuste<-lm(y~x1+x2+x3+x4)
beta<-ajuste$coef
rs<-ajuste$residuals

# boostrap por residuos
boot.fun<-function(rs,i){
  y.boot<-beta[1]+beta[2]*x1+beta[3]*x2+beta[4]*x3+beta[5]*x4+rs[i]
  betaboot<-lm(y.boot~x1+x2+x3+x4)$coefficients
  betaboot
}
Nboot<-999
boot.res<-boot(data=rs,statistic=boot.fun,R=Nboot)

par(mfrow=c(2,2))
hist(boot.res$t[,1],main="",xlab="boot.beta0",breaks=20,freq=FALSE)
fitted0<-fitdistr(boot.res$t[,1],"normal")$estimate
curve(dnorm(x,fitted0[1],fitted0[2]),col="blue",lwd=2,add=TRUE)

hist(boot.res$t[,2],main="",xlab="boot.beta1",breaks=20,freq=FALSE)
fitted1<-fitdistr(boot.res$t[,2],"normal")$estimate
curve(dnorm(x,fitted1[1],fitted1[2]),col="blue",lwd=2,add=TRUE)

hist(boot.res$t[,3],main="",xlab="boot.beta2",breaks=20,freq=FALSE)
fitted2<-fitdistr(boot.res$t[,3],"normal")$estimate
```

```
curve(dnorm(x,fitted2[1],fitted2[2]),col="blue",lwd=2,add=TRUE)

hist(boot.res$t[,4],main="",xlab="boot.beta3",breaks=20,freq=FALSE)
fitted3<-fitdistr(boot.res$t[,4],"normal")$estimate
curve(dnorm(x,fitted3[1],fitted3[2]),col="blue",lwd=2,add=TRUE)


# desvio de los bootstrap por residuos
# se obtienen desde boot.res


# resampling pairs by library

cemento.fit<-function(data){
  n<-nrow(data)
  X<-cbind(rep(1,n),data$x1,data$x2,data$x3,data$x4)
  c(coef(lm(data$y~data$x1+data$x2+data$x3+data$x4)),
    min(eigen(t(X)%*%X)$values))
}
cemento.case<-function(data,i) cemento.fit(data[i,])
cemento.boot<-boot(cemento,cemento.case,R=999)
sd(cemento.boot$t[,2])


# quedarse solo con las matrices con minimo
#autovalor en los 500 del medio
par(mfrow=c(1,1))
plot(cemento.boot$t[,6],cemento.boot$t[,1],
     ylab="beta_hat_0*",xlab="autov")
plot(cemento.boot$t[,6],cemento.boot$t[,2],
     ylab="beta_hat_1*",xlab="autov")

middle.eigen.0<-((cemento.boot$t[,1])
                 [order(cemento.boot$t[,6])])[250:750]
sd(middle.eigen.0)

middle.eigen.1<-((cemento.boot$t[,2])
                 [order(cemento.boot$t[,6])])[250:750]
sd(middle.eigen.1)

middle.eigen.2<-((cemento.boot$t[,3])
                 [order(cemento.boot$t[,6])])[250:750]
sd(middle.eigen.2)

middle.eigen.3<-((cemento.boot$t[,4])
                 [order(cemento.boot$t[,6])])[250:750]
sd(middle.eigen.3)

middle.eigen.4<-((cemento.boot$t[,5])
                 [order(cemento.boot$t[,6])])[250:750]
sd(middle.eigen.4)

##############################################################################
##############################################################################
#cap6

# poverty related to abortion rates
```

```
library(boot)
library(MASS)
x<-c(8,9.4,9.5,9.8,10.15,10.2,11.85,11.96,13,
     13.2,13.5,14.8,16.5,16.5,18.6)
y<-c(30,35,30,28,33,22,25,32,18,37.5,
     30.2,20.6,22.1,17.4,146)
n<-length(y)
X<-cbind(rep(1,n),x)
P<-X%*%solve(t(X)%*%X)%*%t(X)
plot(x,y,main="diagrama de dispersin",
     xlab="pobreza %",ylab="tasa de abortos")
ajuste<-lm(y~x)   # all data
ajuste2<-lm(y[-n]~x[-n])  # saco a DC
Nboot<-1000
abline(ajuste,col="red",lwd=3)
# bootstrap residuos con all data

beta<-c(ajuste$coefficients[1],
        ajuste$coefficients[2])
beta2<-c(ajuste2$coefficients[1],
         ajuste2$coefficients[2])
rs<-ajuste$residuals

# bootstrap por residuos
boot.fun<-function(rs,i){
  y.boot<-beta[1]+beta[2]*x+rs[i]
  betaboot.0<-lm(y.boot~x)$coefficients[1]
  betaboot.1<-lm(y.boot~x)$coefficients[2]

  cbind(betaboot.0,betaboot.1)
}
boot.res<-boot(rs,boot.fun,R=1000)

par(mfrow=c(2,1))
hist(boot.res$t[,1],main="residuos",
     xlab="boot.beta0",breaks=20,freq=FALSE)
fitted0<-fitdistr(boot.res$t[,1],"normal")$estimate
curve(dnorm(x,fitted0[1],fitted0[2]),
      col="blue",lwd=2,add=TRUE)
abline(v=ajuste$coefficients[1],col="red",lwd=3)

hist(boot.res$t[,2],main="residuos",
     xlab="boot.beta1",breaks=20,freq=FALSE)
fitted1<-fitdistr(boot.res$t[,2],"normal")$estimate
curve(dnorm(x,fitted1[1],fitted1[2]),
      col="blue",lwd=2,add=TRUE)
abline(v=ajuste$coefficients[2],col="red",lwd=3)



# metodo de remuestreo por pares all data
datos<-data.frame(cbind(y,x))
names(datos)<-c("y","x")
boot.fit <- function(data) coef(lm(data$y~data$x))
boot.case <- function(data, i) boot.fit(data[i,])
boot1 <- boot(datos, boot.case, R=1000)
```

```
par(mfrow=c(2,1))
hist(boot1$t[,1],main="pares",
     xlab="boot.beta0",breaks=20,freq=FALSE)


hist(boot1$t[,2],main="pares",
     xlab="boot.beta1",breaks=20,freq=FALSE)

par(mfrow=c(1,1))

##############################################################################

##### Wild bootstrap

library(MASS)
library(boot)
# install.packages("fANCOVA")
library(fANCOVA)
n <- 1000
x <- runif(n, min=0, max=1)
## generate heteroscedastic error variances
sig.x <- sqrt(exp(x)/2.5-0.4)
err <- sapply(sig.x, function(x) rnorm(1, sd=x))
x2 <- x^2
y <- 10+3*x+2*x2 +err
plot(x,y)
fit <- lm(y ~ x + x2)
plot(fit$fit,fit$res,xlab="predichos",
     ylab="residuos",main="predichos vs residuos (OLS)")
abline(h=0,col="red",lwd=3)
## obtain 499 samples of the wild bootstrap residuals
res.boot <- wild.boot(fit$res, nboot=499)
## obtain 499 samples of the wild bootstrap responses
y.boot <- matrix(rep(fit$fit,time=499),
                 ncol=499) + res.boot

coef.wild0<-numeric(499)
coef.wild1<-numeric(499)
coef.wild2<-numeric(499)
for (i in 1:499){
  ajuste.wild<-lm(y.boot[,i]~x+x2)
  coef.wild0[i]<-coef(ajuste.wild)[1]
  coef.wild1[i]<-coef(ajuste.wild)[2]
  coef.wild2[i]<-coef(ajuste.wild)[3]
}
#sesgos y desvios del metodo wild
sd.0<-sd(coef.wild0)
bias.0<-10-mean(coef.wild0)

sd.1<-sd(coef.wild1)
bias.1<-3-mean(coef.wild1)

sd.2<-sd(coef.wild2)
bias.2<-2-mean(coef.wild2)


ajuste.wild<-lm(y.boot[,1]~x+x2)
```

```
plot(ajuste.wild$fit,ajuste.wild$res,xlab="predichos",
     ylab="residuos",main="predichos vs residuos (Wild)")
abline(h=0,col="red",lwd=3)

#usando metodologia clasica

r<-sample(fit$res,1000,replace=TRUE)
y.boot.clasico<-fit$fit+r
ajuste.clasico<-lm(y.boot.clasico~x+x2)
plot(ajuste.clasico$fit,ajuste.clasico$res,
     xlab="predichos",ylab="residuos",
     main="predichos vs residuos (Clasico)")
abline(h=0,col="red",lwd=3)

# sesgos y varianzas con diversas metodologias
#remuestreo por residuos
datos<-data.frame(cbind(x,y))
datos.res <- fit$res
datos.df <- data.frame(datos,res=datos.res,
                       fit=fitted(fit))
datos.fit <- function(data){
  data1<-data$x
  data2<-data1^2
  coef(lm(data$y~data$x+data2))
}
datos.model <- function(data, i)
{ d <- data
d$y <- d$fit + d$res[i]
datos.fit(d) }
datos.boot <- boot(datos.df, datos.model, R=499)
datos.boot

#remuestreo por pares

datos.case <- function(data, i) datos.fit(data[i,])
datos.bootp <- boot(datos, datos.case, R=499)
datos.bootp

#####################################################################

## ejemplo de prediccion

library(MASS)
library(boot)
dat<-read.table("dat6MWT",sep=",")
names(dat) <- c("x.6MWT", "peakVo2")
dat<-data.frame(dat)
head(dat)
ajuste<-lm(dat$peakVo2~dat$x.6MWT)
beta<-ajuste$coef
rs<-ajuste$residuals
n<-nrow(dat)

# con libreria asociada
pred.fit<-function(data){
  n<-nrow(data)
  X<-cbind(rep(1,n),dat$x.6MWT)
  c(mean((lm(data$peakVo2~data$x.6MWT)$res)^2),
```

```
    mean((dat$peakVo2-X%*%lm(data$peakVo2~data$x.6MWT)$coef)^2))
}
pred.case<-function(data,i) pred.fit(data[i,])
pred.boot<-boot(dat,pred.case,R=999)

# error de prediccion mejorado (RSS/n+optimismo)
RSS.p<-(1/n)*sum(rs^2)
err.boot.mejorado<-RSS.p+mean(pred.boot$t[,2]-pred.boot$t[,1])



#####################################################################
#####################################################################
#cap7

### Ejemplo: uso del AIC

library(MASS)
library(arm)
data(quine)
attach(quine)
summary(fm1 <- lm( log(Days + 0.5) ~ Eth+Sex+Lrn+Age , data= quine))
coefplot(fm1)
par(mfrow = c(2, 2))
plot(fm1, ask = FALSE, main = "Diagnostic Plots")
par(mfrow = c(1,1))
fm2 <- stepAIC(fm1, trace = F)
summary(fm2)
plot(Eth,log(Days+0.5))

#############################################################################

### Ejemplo: uso de Lasso

library(ncvreg)
data(prostate)
attach(prostate)
X<-cbind(lweight,age,lbph,svi,lcp,gleason,pgg45,lcavol)
library(lars)
ajuste<-lars(X,lpsa)
plot(ajuste)
ajuste2<-lm(lpsa~lweight+age+lbph+svi+lcp+gleason+pgg45+lcavol)
ajuste2$coefficients



library(parcor)
adalasso(X=X,y=lpsa,k=10)

# comparacion con AIC
aic<-stepAIC(ajuste2)
summary(aic)

#############################################################################

## seleccion datos de anemia
library(boot)
library(MASS)
```

```
# Read data
dat <- read.table("dat_anemia.txt")
summary(dat)


#Exploratory analysis for continues variables

attach(dat)
length(dat[1,])
datos<-dat[,-c(10,11,12,13,17)]
head(datos)
datos$anemie_admit2 <- factor(datos$anemie_admit2)
datos$sex <- factor(datos$sex)
datos$CKD <- factor(datos$CKD)
datos$DM_pre <- factor(datos$DM_pre)
datos$KHK_pre <- factor(datos$KHK_pre)
datos$LVEF_pre <- factor(datos$LVEF_pre)
datos$Dialyse_pre <- factor(datos$Dialyse_pre)

# se escalan y centran
# las variables continuas

X<-cbind(age,ln.WBC_Admit,ln.CRP_Admit,ln.crea)
X2<-scale(X)
datos<-cbind(Hgb_Admit,sex,CKD,DM_pre,
             KHK_pre,LVEF_pre,Dialyse_pre,X2)
dat2<-data.frame(datos)
head(dat2)

# Analisis con todas las variables

model.anem1 <- lm(Hgb_Admit ~ .,
                  data = dat2)

summary(model.anem1)

##### AIC

### pares

p.star<-NULL
coef.names<-NULL
for(b in 1:500)
{
  ind <- sample(1:150, replace = T)
  boot.fix.data <- dat2[ind, ]
  tmp.0 <-  lm(Hgb_Admit ~ . , data=boot.fix.data)
  suppressWarnings(tmp<- stepAIC(tmp.0,trace=F))
  p.star<-c(p.star,length(coef(tmp)))
  coef.names <- c(coef.names, names(tmp$coeff))
  cat("bootstrap sample number", b, "....................................","\n")
}

hist(p.star,breaks=50,main="metodo de pares")
table(coef.names)/500


###residuos
```

```
m1 <- glm(Hgb_Admit ~ .,
                  data = dat2)

p.diag<-glm.diag.plots(m1,ret=T)
p.res <- p.diag$res
p.res <- p.res - mean(p.res)
p.df <- data.frame(dat2,res=p.res,fit=fitted(m1))

p.fit <- function(data){
  tmp.0<-glm(Hgb_Admit~.,data=data)
  suppressWarnings(tmp<- stepAIC(tmp.0,trace=F))
  length(coef(tmp))
}
p.model <- function(data, i)
{ d <- data
d$Hgb_Admit <- d$fit + d$res[i]
p.fit(d) }
p.boot <- boot(p.df, p.model, R=500)
par(mfrow=c(1,2))
hist(p.boot$t,breaks=50,
     main="metodo residuos (AIC)",xlab="p.star")
hist(p.star,breaks=50,main="metodo de pares (AIC)")
par(mfrow=c(1,1))


#### LASSO
library(ncvreg)
library(lars)
library(parcor)
### pares

p.star<-NULL
for(b in 1:500)
{
  ind <- sample(1:150, replace = T)
  b.d <- dat2[ind, ]
  X<-cbind(b.d$age,b.d$CKD,b.d$Dialyse_pre,b.d$DM_pre,b.d$KHK_pre,
           b.d$ln.crea,b.d$ln.CRP_Admit,
           b.d$ln.WBC_Admit,b.d$LVEF_pre,b.d$sex)
  p.star<-c(p.star,
            sum(adalasso(X=X,y=b.d$Hgb_Admit,k=10)$coefficients.adalasso!=0))
  cat("bootstrap sample number", b, ".....................................","\n")
}

hist(p.star,breaks=50,main="metodo de pares")



###residuos
m1 <- glm(Hgb_Admit ~ .,
          data = dat2)

p.diag<-glm.diag.plots(m1,ret=T)
p.res <- p.diag$res
p.res <- p.res - mean(p.res)
p.df <- data.frame(dat2,res=p.res,fit=fitted(m1))

p.fit <- function(data){
```

```
  X<-cbind(data$age,data$CKD,data$Dialyse_pre,data$DM_pre,data$KHK_pre,
           data$ln.crea,data$ln.CRP_Admit,
           data$ln.WBC_Admit,data$LVEF_pre,data$sex)
  sum(adalasso(X=X,y=data$Hgb_Admit,k=10)$coefficients.adalasso!=0)
}
p.model <- function(data, i)
{ d <- data
d$Hgb_Admit <- d$fit + d$res[i]
p.fit(d) }
p.boot <- boot(p.df, p.model, R=500)
par(mfrow=c(1,2))
hist(p.boot$t,breaks=50,
     main="metodo residuos (Lasso)",xlab="p.star")
hist(p.star,breaks=50,main="metodo pares (Lasso)")
par(mfrow=c(1,1))

##############################################################################


### Nuclear station

library(boot)
attach(nuclear)
ajuste<-lm(log(cost)~log(cap)+log(cum.n)+pt+
             pr+ne+ct+bw+t1+t2+date,
           data=nuclear)
summary(ajuste)
par(mfrow=c(2,2))
plot(ajuste)
par(mfrow=c(1,1))

# leave one out cross validation
loocv<-function(fit){
  h=lm.influence(fit)$h
  mean((residuals(fit)/(1-h))^2)
}
loocv(ajuste)


# plot loocv segun la cantidad de variables consideradas
# forma bruta
# mas abajo: forma elegante

ajuste1<-lm(log(cost)~date,data=nuclear)
ajuste2<-lm(log(cost)~date+log(cap),data=nuclear)
ajuste3<-lm(log(cost)~date+log(cap)+ne,data=nuclear)
ajuste4<-lm(log(cost)~date+log(cap)+ne+ct,data=nuclear)
ajuste5<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n),data=nuclear)
ajuste6<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n)+
              pt,data=nuclear)
ajuste7<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n)+
              pt+t1,data=nuclear)
ajuste8<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n)+
              pt+t1+t2,data=nuclear)
ajuste9<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n)+
              pt+t1+t2+pr,data=nuclear)
ajuste10<-lm(log(cost)~date+log(cap)+ne+ct+log(cum.n)+
               pt+t1+t2+pr+bw,data=nuclear)
```

```
cv<-numeric(10)
cv[1]<-loocv(ajuste1)
cv[2]<-loocv(ajuste2)
cv[3]<-loocv(ajuste3)
cv[4]<-loocv(ajuste4)
cv[5]<-loocv(ajuste5)
cv[6]<-loocv(ajuste6)
cv[7]<-loocv(ajuste7)
cv[8]<-loocv(ajuste8)
cv[9]<-loocv(ajuste9)
cv[10]<-loocv(ajuste10)
plot(seq(1:10),cv,type="b",col="blue",
     xlab="n mero de covariables",
     ylab="error de predicci n")


# idem pero mediante bootstrap
Nboot<-100
n<-length(date)
m<-0
muestreo<-sample(1:(n-m),n-m,replace=TRUE)
y<-log(cost)

r.star1<-numeric(Nboot)
r.star2<-numeric(Nboot)
r.star3<-numeric(Nboot)
r.star4<-numeric(Nboot)
r.star5<-numeric(Nboot)
r.star6<-numeric(Nboot)
r.star7<-numeric(Nboot)
r.star8<-numeric(Nboot)
r.star9<-numeric(Nboot)
r.star10<-numeric(Nboot)

X1<-cbind(rep(1,n-m),date)
X2<-cbind(rep(1,n-m),date,log(cap))
X3<-cbind(rep(1,n-m),date,log(cap),ne)
X4<-cbind(rep(1,n-m),date,log(cap),ne,ct)
X5<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n))
X6<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt)
X7<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1)
X8<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1,t2)
X9<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1,t2,pr)
X10<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
           pt,t1,t2,pr,bw)

for (i in 1:Nboot){
  muestra<-sample(1:n,n-m,replace=TRUE)
  X1.star<-X1[muestra,]
  X2.star<-X2[muestra,]
  X3.star<-X3[muestra,]
  X4.star<-X4[muestra,]
  X5.star<-X5[muestra,]
  X6.star<-X6[muestra,]
```

```
  X7.star<-X7[muestra,]
  X8.star<-X8[muestra,]
  X9.star<-X9[muestra,]
  X10.star<-X10[muestra,]
  y.star<-y[muestra]
  tmp1<-lm(y.star~X1.star[,2])
  tmp2<-lm(y.star~X2.star[,2]+X2.star[,3])
  tmp3<-lm(y.star~X3.star[,2]+X3.star[,3]+
             X3.star[,4])
  tmp4<-lm(y.star~X4.star[,2]+X4.star[,3]+
             X4.star[,4]+X4.star[,5])
  tmp5<-lm(y.star~X5.star[,2]+X5.star[,3]+
             X5.star[,4]+X5.star[,5]+X5.star[,6])
  tmp6<-lm(y.star~X6.star[,2]+X6.star[,3]+
             X6.star[,4]+X6.star[,5]+X6.star[,6]+
             X6.star[,7])
  tmp7<-lm(y.star~X7.star[,2]+X7.star[,3]+
             X7.star[,4]+X7.star[,5]+X7.star[,6]+
             X7.star[,7]+X7.star[,8])
  tmp8<-lm(y.star~X8.star[,2]+X8.star[,3]+
             X8.star[,4]+X8.star[,5]+X8.star[,6]+
             X8.star[,7]+X8.star[,8]+X8.star[,9])
  tmp9<-lm(y.star~X9.star[,2]+X9.star[,3]+X9.star[,4]+
             X9.star[,5]+X9.star[,6]+X9.star[,7]+
             X9.star[,8]+X9.star[,9]+X9.star[,10])
  tmp10<-lm(y.star~X10.star[,2]+X10.star[,3]+
              X10.star[,4]+X10.star[,5]+X10.star[,6]+
              X10.star[,7]+X10.star[,8]+X10.star[,9]+
              X10.star[,10]+X10.star[,11])

  r.star1[i]<-mean((y-X1%*%tmp1$coefficients)^2)
  r.star2[i]<-mean((y-X2%*%tmp2$coefficients)^2)
  r.star3[i]<-mean((y-X3%*%tmp3$coefficients)^2)
  r.star4[i]<-mean((y-X4%*%tmp4$coefficients)^2)
  r.star5[i]<-mean((y-X5%*%tmp5$coefficients)^2)
  r.star6[i]<-mean((y-X6%*%tmp6$coefficients)^2)
  r.star7[i]<-mean((y-X7%*%tmp7$coefficients)^2)
  r.star8[i]<-mean((y-X8%*%tmp8$coefficients)^2)
  r.star9[i]<-mean((y-X9%*%tmp9$coefficients)^2)
  r.star10[i]<-mean((y-X10%*%tmp10$coefficients)^2)

}
cv.star<-c(mean(r.star1),mean(r.star2),mean(r.star3),
           mean(r.star4),mean(r.star5),mean(r.star6),
           mean(r.star7),mean(r.star8),mean(r.star9),
           mean(r.star10))

Nboot<-100
n<-length(date)
m<-16
muestreo<-sample(1:(n-m),n-m,replace=TRUE)
y<-log(cost)

r.star1<-numeric(Nboot)
r.star2<-numeric(Nboot)
r.star3<-numeric(Nboot)
r.star4<-numeric(Nboot)
r.star5<-numeric(Nboot)
```

```
r.star6<-numeric(Nboot)
r.star7<-numeric(Nboot)
r.star8<-numeric(Nboot)
r.star9<-numeric(Nboot)
r.star10<-numeric(Nboot)

X1<-cbind(rep(1,n-m),date)
X2<-cbind(rep(1,n-m),date,log(cap))
X3<-cbind(rep(1,n-m),date,log(cap),ne)
X4<-cbind(rep(1,n-m),date,log(cap),ne,ct)
X5<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n))
X6<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt)
X7<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1)
X8<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1,t2)
X9<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
          pt,t1,t2,pr)
X10<-cbind(rep(1,n-m),date,log(cap),ne,ct,log(cum.n),
           pt,t1,t2,pr,bw)

for (i in 1:Nboot){
  muestra<-sample(1:n,n-m,replace=TRUE)
  X1.star<-X1[muestra,]
  X2.star<-X2[muestra,]
  X3.star<-X3[muestra,]
  X4.star<-X4[muestra,]
  X5.star<-X5[muestra,]
  X6.star<-X6[muestra,]
  X7.star<-X7[muestra,]
  X8.star<-X8[muestra,]
  X9.star<-X9[muestra,]
  X10.star<-X10[muestra,]
  y.star<-y[muestra]
  tmp1<-lm(y.star~X1.star[,2])
  tmp2<-lm(y.star~X2.star[,2]+X2.star[,3])
  tmp3<-lm(y.star~X3.star[,2]+X3.star[,3]+
             X3.star[,4])
  tmp4<-lm(y.star~X4.star[,2]+X4.star[,3]+
             X4.star[,4]+X4.star[,5])
  tmp5<-lm(y.star~X5.star[,2]+X5.star[,3]+
             X5.star[,4]+X5.star[,5]+X5.star[,6])
  tmp6<-lm(y.star~X6.star[,2]+X6.star[,3]+
             X6.star[,4]+X6.star[,5]+X6.star[,6]+
             X6.star[,7])
  tmp7<-lm(y.star~X7.star[,2]+X7.star[,3]+
             X7.star[,4]+X7.star[,5]+X7.star[,6]+
             X7.star[,7]+X7.star[,8])
  tmp8<-lm(y.star~X8.star[,2]+X8.star[,3]+
             X8.star[,4]+X8.star[,5]+X8.star[,6]+
             X8.star[,7]+X8.star[,8]+X8.star[,9])
  tmp9<-lm(y.star~X9.star[,2]+X9.star[,3]+X9.star[,4]+
             X9.star[,5]+X9.star[,6]+X9.star[,7]+
             X9.star[,8]+X9.star[,9]+X9.star[,10])
  tmp10<-lm(y.star~X10.star[,2]+X10.star[,3]+
              X10.star[,4]+X10.star[,5]+X10.star[,6]+
              X10.star[,7]+X10.star[,8]+X10.star[,9]+
```

```
              X10.star[,10]+X10.star[,11])

  r.star1[i]<-mean((y-X1%*%tmp1$coefficients)^2)
  r.star2[i]<-mean((y-X2%*%tmp2$coefficients)^2)
  r.star3[i]<-mean((y-X3%*%tmp3$coefficients)^2)
  r.star4[i]<-mean((y-X4%*%tmp4$coefficients)^2)
  r.star5[i]<-mean((y-X5%*%tmp5$coefficients)^2)
  r.star6[i]<-mean((y-X6%*%tmp6$coefficients)^2)
  r.star7[i]<-mean((y-X7%*%tmp7$coefficients)^2)
  r.star8[i]<-mean((y-X8%*%tmp8$coefficients)^2)
  r.star9[i]<-mean((y-X9%*%tmp9$coefficients)^2)
  r.star10[i]<-mean((y-X10%*%tmp10$coefficients)^2)

}
cv.star2<-c(mean(r.star1),mean(r.star2),mean(r.star3),
            mean(r.star4),mean(r.star5),mean(r.star6),
            mean(r.star7),mean(r.star8),mean(r.star9),
            mean(r.star10))
plot(seq(1:10),cv.star,type="b",col="green",
     xlab="n mero de covariables",
     ylab="error de predicci n agregado")
points(seq(1:10),cv,type="b",col="blue")
points(seq(1:10),cv.star2,type="b",col="red")



# metodo optimo para hacer el analisis
# funcion de binarizacion
integer.base.b <-
  function(x, b=2){
    xi <- as.integer(x)
    if(any(is.na(xi) | ((x-xi)!=0)))
      print(list(ERROR="x not integer", x=x))
    N <- length(x)
    xMax <- max(x)
    ndigits <- (floor(logb(xMax, base=2))+1)
    Base.b <- array(NA, dim=c(N, ndigits))
    for(i in 1:ndigits){#i <- 1
      Base.b[, ndigits-i+1] <- (x %% b)
      x <- (x %/% b)
    }
    if(N ==1) Base.b[1, ] else Base.b
  }



nuclear.glm<-glm(log(cost)~date+log(cap)+ne+ct+
                   log(cum.n)+pt+t1+t2+pr+bw,
                 data=nuclear)
n<-nrow(nuclear)
#error de pred K-fold cross-validation ajustado
cv.err <- cv.glm(data, nuclear.glm, K=n)$delta # leave one out
cv.16.err <- cv.glm(data, nuclear.glm,  K=16)$delta
cv.10.err <- cv.glm(data, nuclear.glm, K=10)$delta
cv.7.err <- cv.glm(data, nuclear.glm, K=7)$delta
cv.2.err <- cv.glm(data, nuclear.glm, K=2)$delta
```

```
todo<-matrix(NA,1024,14)
n<-nrow(nuclear)
m<-8
Nboot<-10
for (x in 0:1023){
  binario<-as.integer(intToBits(x))[1:10]
  s1<-ifelse(binario[1]==1,"+date","")
  s2<-ifelse(binario[2]==1,"+log(cap)","")
  s3<-ifelse(binario[3]==1,"+ne","")
  s4<-ifelse(binario[4]==1,"+ct","")
  s5<-ifelse(binario[5]==1,"+log(cum.n)","")
  s6<-ifelse(binario[6]==1,"+pt","")
  s7<-ifelse(binario[7]==1,"+t1","")
  s8<-ifelse(binario[8]==1,"+t2","")
  s9<-ifelse(binario[9]==1,"+pr","")
  s10<-ifelse(binario[10]==1,"+bw","")

  X<-rep(1,n)
  X<-cbind(X)
  s11<-if(binario[1]==1) X<-cbind(X,date)
  s22<-if(binario[2]==1) X<-cbind(X,log(cap))
  s33<-if(binario[3]==1) X<-cbind(X,ne)
  s44<-if(binario[4]==1) X<-cbind(X,ct)
  s55<-if(binario[5]==1) X<-cbind(X,log(cum.n))
  s66<-if(binario[6]==1) X<-cbind(X,pt)
  s77<-if(binario[7]==1) X<-cbind(X,t1)
  s88<-if(binario[8]==1) X<-cbind(X,t2)
  s99<-if(binario[9]==1) X<-cbind(X,pr)
  s1010<-if(binario[10]==1) X<-cbind(X,bw)


  laformula<-paste("log(cost)~1",s1,s2,s3,s4,s5,s6,
                   s7,s8,s9,s10,sep="")
  ajuste<-lm(laformula,data=nuclear)
  r<-numeric(Nboot)
  rr<-numeric(Nboot)
  for (j in 1:Nboot){
    muestra<-sample(1:n,n-m,replace=TRUE)
    X2<-X[muestra,]
    datos.m<-nuclear[muestra,]
    laformula.boot<-paste("log(cost)~1",s1,s2,s3,s4,s5,s6,
                          s7,s8,s9,s10,sep="")
    ajuste.boot<-lm(laformula.boot,data=datos.m)
    r[j]<-mean(ajuste.boot$res^2,na.rm=TRUE)
    ifelse (length(X2)==n-m,
            rr[j]<-mean((log(cost)[muestra]-X2*ajuste.boot$coef)^2,
                        na.rm=TRUE),
            rr[j]<-mean((log(cost)[muestra]-X2%*%ajuste.boot$coef)^2,
                        na.rm=TRUE))

  }
  todo[x+1,]<-c(sum(binario)+1,binario,loocv(ajuste),
                mean(ajuste$res^2,na.rm=TRUE)-mean(rr-r,na.rm=TRUE),
                mean(rr,na.rm=TRUE))
}
todo
indice.cv<-which(todo[,12]==min(todo[,12])) # indice del menor cv
bin.cv<-todo[indice.cv,]   # valores para el minimo cv
```

```
indice.boot<-which(todo[,13]==min(todo[,13]))
# metodo bootstrap no consistente
bin.boot<-todo[indice.boot,]

indice.boot.c<-which(todo[,14]==min(todo[,14]))
# metodo bootstrap consistente
bin.boot.c<-todo[indice.boot.c,]


par(mfrow=c(1,2))
plot(todo[,1],todo[,12],pch="*",xlab="numero de variables",
     ylab="loocv errors")

plot(todo[,1],todo[,14],pch="*",xlab="numero de variables",
     ylab="bootstrap errors")

par(mfrow=c(1,1))

#########################################################################
#########################################################################
#cap8
#### Ejemplo basico de regresion logistica

library(arm)
# Parametros
b0 <- 1
b1 <- 2.5
b2 <- 2
# variables de regresion
x1 <- rnorm(200)
x2 <- rbinom(200, 1, 0.5)
# respuesta binaria como funcion de "b0 + b1 * x1 + b2 * x2"
y <- rbinom(200, 1,
            invlogit(b0 + b1 * x1 + b2 * x2))

# Grafico y visualizacion de los datos
jitter.binary <- function(a, jitt = 0.05)
{
ifelse(a==0, runif(length(a), 0, jitt),
       runif(length(a), 1-jitt, 1))
}
plot(x1, jitter.binary(y), xlab = "x1",
     ylab = "Probabilidad de exito")
curve(invlogit(b0 + b1*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2)
curve(invlogit(b0 + b1*x + b2),
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2)
legend("bottomright", c("b2 = 0", "b2 = 2"),
       col = c("blue", "red"), lwd = 2, lty = 1)

## Ajuste del modelo logistico

fn <- glm(y ~ x1 + x2, family = binomial())
summary(fn)


# Coeficientes y visualizacion
```

```
plot(x1, jitter.binary(y), xlab = "x1",
     ylab = "Probabilidad de exito")
beta <- coef(fn)
b0.hat <- beta[1]
b1.hat <- beta[2]
b2.hat <- beta[3]
curve(invlogit(b0 + b1*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2)
curve(invlogit(b0.hat + b1.hat*x),
      from = -2.5, to = 2.5, add = TRUE, col ="blue", lwd = 2,
      lty =2)
curve(invlogit(b0 + b1*x + b2),
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2)
curve(invlogit(b0.hat + b1.hat*x + b2.hat),
      from = -2.5, to = 2.5, add = TRUE, col ="red", lwd =2,
      lty =2)
legend("bottomright", c("b2 = 0", "b2 = 2"),
       col = c("blue", "red"), lwd = 2, lty = 1)




################################################################################


## cane figure

data(cane)
head(cane)

model.cane.1 <- glm(cbind(r, n-r) ~  var + block,
                    data = cane, family = binomial(link = logit))

summary(model.cane.1)

#model.cane.2 <- glm(cbind(r, n-r) ~  var + block, data = cane, family = ↩
    quasibinomial(link = "logit"))

#summary(model.cane.2)

# Grafico predictor vs variedades
par(mfrow = c(1,2))
df <- data.frame(est.var = predict(model.cane.1), cane)
with(df[df$block=="A",], plot(est.var[-c(31,1,3)],
                              ylim=c(-7,2),xlab="Variedad",ylab="eta"))
p <- c(0.5, 0.2, 0.05, 0.01)
lines.level <- log(p/(1-p))
abline(h = lines.level)
text(31,df[df$block=="A",]$est.var[31],31)
text(1,df[df$block=="A",]$est.var[1],1)
text(2,df[df$block=="A",]$est.var[3],3)

plot(model.cane.1$linear.predictors, residuals(model.cane.1),xlab="eta",ylab="rP")
abline(h = c(-2, 2), lty = 2, col = "red")
par(mfrow = c(1,1))
```

```
## Bootstrap Deviance
library(boot)
attach(cane)

cane.glm<-glm(cbind(r, n-r) ~  var + block,
              family = binomial(link = "logit"),data=cane)
cane.diag<-glm.diag.plots(cane.glm,ret=T)
num<-r/n-cane.glm$fit
denom<-sqrt((1/n)*cane.glm$fit*(1-cane.glm$fit)*8.3*(1-cane.diag$h))
cane.res <- num/denom # residuos pearson estandarizados
cane.res <- cane.res - mean(cane.res)
cane.df <- data.frame(cane,res=cane.res,fit=fitted(cane.glm))


df <- data.frame(est.var = predict(cane.glm),cane)
with(df[df$block=="A",], rank(est.var)[1])




# parametric binomial

cane.fun <- function(data)
{ tmp <- glm(cbind(r, n-r)~  var + block,
             family = binomial(link = "logit"),data=data)
deviance(tmp)}
cane.sim <- function(data, mle)
{
  for (i in 1:nrow(data)){
    data$r[i] <- rbinom(1,data$n[i], mle[i])
  }
  data
}
cane.mle <- fitted(cane.glm)
cane.boot.sim <- boot(cane, cane.fun, R=199,
                      sim="parametric", ran.gen=cane.sim,
                      mle=cane.mle)


# non parametric bootstrap usando residuos pearson y no estratificados

cane.model <- function(data, i)
{ d <- data
y<-d$fit + sqrt((1/(d$n))*8.3*d$fit*(1-d$fit))*d$res[i]
y2 <- y*d$n
for (i in 1:length(d$n)){

  if(y2[i]<0) y2[i]<-0
  if(y2[i]>cane.df$n[i]) y2[i]<-cane.df$n[i]
  if (y2[i]-floor(y2)[i]>0.5) y2[i]<-floor(y2[i]+1)
  else y2[i]<- floor(y2[i])
}
d$r<-y2
tmp<-glm(cbind(d$r, d$n-d$r)~  var + block,
         family = binomial(link = "logit"),data=d)
deviance(tmp)
```

```
}
cane.boot<-boot(cane.df,cane.model,R=199,strata=cane.df$block)
cane.boot$t
boxplot(cane.boot$t/132,ylim=c(0,12))
abline(h=deviance(cane.glm)/132)

uno<- c(cane.boot$t/132)
dos<-c(cane.boot.sim$t/132)
boxplot(dos,uno,ylim=c(0,12),ylab="deviance/df")
abline(h=deviance(cane.glm)/132,lty=2)


# non parametric with stratified residuals
cane2.df<-cane.df
attach(cane2.df)
est.var<-predict(cane.glm)
cane2.df$est<-est.var

bloqueD<-cane2.df[cane2.df$block=="D",]
ordenD<-order(bloqueD$est)
cane2.df$strata<-rep(1,180)
bloqueD$strata<-rep(1,45)
bloqueD$strata[ordenD]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="D",]<-bloqueD

bloqueA<-cane2.df[cane2.df$block=="A",]
ordenA<-order(bloqueA$est)
bloqueA$strata<-rep(1,45)
bloqueA$strata[ordenA]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="A",]<-bloqueA

bloqueB<-cane2.df[cane2.df$block=="B",]
ordenB<-order(bloqueB$est)
bloqueB$strata<-rep(1,45)
bloqueB$strata[ordenB]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="B",]<-bloqueB

bloqueC<-cane2.df[cane2.df$block=="C",]
ordenC<-order(bloqueC$est)
bloqueC$strata<-rep(1,45)
bloqueC$strata[ordenC]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="C",]<-bloqueC



cane.boot.2<-boot(cane2.df,cane.model,R=199,strata=cane2.df$strata)
cane.boot.2$t/132



boxplot(c(cane.boot.sim$t/132),c(cane.boot$t/132),c(cane.boot.2$t/132),ylim=c↩
    (0,12),ylab="deviance/df")
abline(h=deviance(cane.glm)/132,lty=2)
```

```
### distribucion rangos bootstrap para variedad 1 y 3


cane.model <- function(data, i)
{ d <- data
y<-d$fit + sqrt((1/(d$n))*8.3*d$fit*(1-d$fit))*d$res[i]
y2 <- y*d$n
for (i in 1:length(d$n)){

  if(y2[i]<0) y2[i]<-0
  if(y2[i]>cane.df$n[i]) y2[i]<-cane.df$n[i]
  if (y2[i]-floor(y2)[i]>0.5) y2[i]<-floor(y2[i]+1)
  else y2[i]<- floor(y2[i])
}
d$r<-y2
tmp<-glm(cbind(d$r, d$n-d$r)~  var + block,
         family = binomial(link = "logit"),data=d)
df <- data.frame(est.var = predict(tmp),cane)
a<-with(df[df$block=="A",], rank(est.var)[1])
b<-with(df[df$block=="A",], rank(est.var)[3])
c<-with(df[df$block=="A",], rank(est.var)[31])
c(a,b,c)
}



cane2.df<-cane.df
attach(cane2.df)
est.var<-predict(cane.glm)
cane2.df$est<-est.var

bloqueD<-cane2.df[cane2.df$block=="D",]
ordenD<-order(bloqueD$est)
cane2.df$strata<-rep(1,180)
bloqueD$strata<-rep(1,45)
bloqueD$strata[ordenD]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="D",]<-bloqueD

bloqueA<-cane2.df[cane2.df$block=="A",]
ordenA<-order(bloqueA$est)
bloqueA$strata<-rep(1,45)
bloqueA$strata[ordenA]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="A",]<-bloqueA

bloqueB<-cane2.df[cane2.df$block=="B",]
ordenB<-order(bloqueB$est)
bloqueB$strata<-rep(1,45)
bloqueB$strata[ordenB]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="B",]<-bloqueB

bloqueC<-cane2.df[cane2.df$block=="C",]
ordenC<-order(bloqueC$est)
bloqueC$strata<-rep(1,45)
bloqueC$strata[ordenC]<-c(rep(1,15),rep(2,15),rep(3,15))
cane2.df[cane2.df$block=="C",]<-bloqueC


cane.boot.2<-boot(cane2.df,cane.model,
```

```
                  R=1000,strata=cane2.df$strata)
par(mfrow=c(1,2))
hist(cane.boot.2$t[,1],breaks=30,
     xlab="rank variedad 1",main="")
hist(cane.boot.2$t[,3],breaks=30,
     xlab="rank variedad 31",main="")
par(mfrow=c(1,1))

mean(cane.boot.2$t[,1]==45)
mean(cane.boot.2$t[,3]==1)


##############################################################################

# Datos de orina: prediccion en reg log

library(boot)
data<-urine[c(-1,-55),]
attach(data)
#visualizacion de la data

library(lattice)
parallelplot(~data[, 2:7] | factor(r), data = data)




#error de prediccion
cost <- function(r, pi=0) mean(abs(r-pi)>0.5)
urine.glm <- glm(r~gravity+cond+log(calc)+log(urea),
                 binomial,data=data)
urine.diag <- glm.diag(urine.glm)
#error de prediccion aparente
app.err <- cost(r, fitted(urine.glm))
#error de pred K-fold cross-validation ajustado
cv.err <- cv.glm(data, urine.glm, cost, K=77)$delta
# leave one out
cv.38.err <- cv.glm(data, urine.glm, cost, K=38)$delta
cv.10.err <- cv.glm(data, urine.glm, cost, K=10)$delta
cv.7.err <- cv.glm(data, urine.glm, cost, K=7)$delta
cv.2.err <- cv.glm(data, urine.glm, cost, K=2)$delta
#error de pred 0.632 bootstrap
#For resampling-based estimates and plot for 0.632 errors:
urine.pred.fun <- function(data, i, model)
{
  d <- data[i,]
  d.glm <- update(model,data=d)
  # vuelve a fitear el modelo actualizado
  pred <- predict(d.glm,data,type="response")
  # type response devuelve predichos en la
  #escala de la variable dependiente
  #(por default lo daria en la escala del predictor lineal)
  D.F.Fhat <- cost(data$r, pred)
  # error de pred del modelo ajustado por
  #la muestra boot sobre la muestra original
  D.Fhat.Fhat <- cost(d$r, fitted(d.glm))
  # error aparente en cada muestra boot
  c(data$r-pred, D.F.Fhat - D.Fhat.Fhat)
```

```
  # la resta del primero con el segundo
  # devuelve el optimismo
  #(eso aparece en la ultima columna del urine.boot$t)
}
urine.boot <- boot(data, urine.pred.fun, R=200,
                   model=urine.glm)
urine.boot$f <- boot.array(urine.boot)
# dice cuantas veces aparece cada observacion
#en cada muestra bootstrap
n <- nrow(data)
err.boot <- mean(urine.boot$t[,n+1]) + app.err
# error bootstrap mejorado
ord <- order(urine.diag$res)
# se ordenan los residuos
urine.pred <- urine.boot$t[,ord]
# y se ordenan las predicciones segun el orden
# de los residuos
err.632 <- 0
n.632 <- NULL
pred.632 <- NULL
for (i in 1:n) {
  # uno se  fija las muestras bootstrap en
  # las que no aparece cada observacion
  inds <- urine.boot$f[,i]==0
  err.632 <- err.632 + cost(urine.pred[inds,i])/n
  # se calcula el error de la obs i con las
  # muestras en las que no aparece i
  n.632 <- c(n.632, sum(inds))
  # cantidad de muestras boot en
  #las que no aparecio la obs i.
  #Se Guarda en un vector para
  #despues generar los factores
  pred.632 <- c(pred.632, urine.pred[inds,i])
  # aca se guarda en el orden de los residuos
  # los varios errores de prediccion 0.632 de cada observacion
}

err.632 <- 0.368*app.err + 0.632*err.632
# calculo finalmente el error boot 0.632
urine.fac <- factor(rep(1:n,n.632),labels=ord)
# distingo a qu  observacion pertenecen las predicciones
plot(urine.fac, pred.632,ylab="Prediction errors",
     xlab="Case ordered by residual")
abline(h=-0.5,lty=2)
abline(h=0.5,lty=2)

## los mal clasificados siempre

cont<-numeric(77)
for (i in 1:77){
  cont[i]<-sum(urine.pred[,i]<0.5)

}
obs<-cont==0

## el 74 y el 77 siempre mal clasificados. Por que?

parallelplot(~data[, 2:7] | factor(obs), data = data)
```

```
## se comparan medias entre el grupo
## con presencia de cristales y ausencia

cero<-data[r %in% c(0),]
uno<-data[r %in% c(1),]
nombres<-c("gravity","ph","osmo","cond","urea","calc")
cero.1<-c(mean(cero[,2]),mean(cero[,3]),mean(cero[,4]),
          mean(cero[,5]),mean(cero[,6]),mean(cero[,7]))

uno.1<-c(mean(uno[,2]),mean(uno[,3]),mean(uno[,4]),
          mean(uno[,5]),mean(uno[,6]),mean(uno[,7]))

#para 77, es del tipo 1
plot(cero.1,type="o",ylim=c(-10,max(uno.1)),main="observacion 77",lwd=3)
points(1:6,c(data[77,2:7]),col="red",pch="0",lwd=4)
points(1:6,uno.1,col="green",pch="0",lwd=4)

# comparacion: diferencia del 77
#. con el tipo 1 y el tipo 0
as.numeric(data[77,2:7])
dif77.0<-abs(as.numeric(data[77,2:7])-cero.1)
dif77.1<-abs(as.numeric(data[77,2:7])-uno.1)

#para 74, es del tipo 1
plot(cero.1,type="o")
points(1:6,c(data[74,2:7]),col="red",pch="*")
points(1:6,uno.1,col="green",pch="+")

#para 27, es del tipo 0
plot(uno.1,type="o",main="observacion 27",lwd=3)
points(1:6,c(data[27,2:7]),col="red",pch="0",lwd=4)
points(1:6,cero.1,col="green",pch="0",lwd=4)

# comparacion: diferencia del 27
#. con el tipo 1 y el tipo 0

dif27.0<-mean(abs(as.numeric(data[27,2:7])-cero.1))
dif27.1<-mean(abs(as.numeric(data[27,2:7])-uno.1))


## cantidad de falsos
# es decir, tipos mas (en media) parecido a su opuesto
cont<-numeric(77)
for (i in 1:77){
  tipo<-as.numeric(data[i,1])
  dif.0<-abs(as.numeric(data[i,2:7])-cero.1)
  dif.1<-abs(as.numeric(data[i,2:7])-uno.1)
  dif<-dif.0-dif.1
  if (tipo==0){
    cont[i]<-sum(dif>0)
  }
  if (tipo==1){
    cont[i]<-sum(dif<0)
  }
}
mean(cont>=5)
```

```
#############################################################################
#############################################################################
#cap9

library(MASS)
library(boot)
dat<-read.csv2("C:/Users/Gaspard/Desktop/datwork.csv")
names(dat)


# llevo algunos a factores

dat$death <- factor(dat$death)
dat$sex <- factor(dat$sex)
dat$dialyse <- factor(dat$dialyse)
dat$nekrosen <- factor(dat$nekrosen)
dat$hautabl <- factor(dat$hautabl)
dat$blasenbild <- factor(dat$blasenbild)
dat$hypotonie <- factor(dat$hypotonie)
dat$tachykardie <- factor(dat$tachykardie)
dat$nierenvers <- factor(dat$nierenvers)
dat$sepsis <- factor(dat$sepsis)
dat$lokalisation <- factor(dat$lokalisation)
dat$typ <- factor(dat$typ)
dat$adipositas <- factor(dat$adipositas)
dat$asa <- factor(dat$asa)
dat$ecoli <- factor(dat$ecoli)
dat$vasopressors <- factor(dat$vasopressors)

library(randomForest)
dat.roughfix <- na.roughfix(dat)

# An lisis individuales

# fisher's exact test
# introducir en orden a,b,c,d (a es y=1 & x=1, b es y=1 & x=0, c es y=0 & x=1)
fisher.ex<-function(a,b,c,d,level){
  or<-(a/c)/(b/d)
  z<-qnorm(1-level/2)
  l<-c(log(or)-z*sqrt(1/a+1/b+1/c+1/d),log(or)+z*sqrt(1/a+1/b+1/c+1/d))
  lim<-exp(l)
  salida<-c(or,lim[1],lim[2],fisher.test(rbind(c(a,b),c(c,d)))$p.value)
  salida}



# test de fisher para vasopressors

tmp1<-dat.roughfix$vasopressors ==1
tmp2<-dat.roughfix$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
fisher.ex(a,b,c,d,0.05)
```

```
# modelo con las variables significativas
# en la instancia univariada

m3<-glm(death~nierenvers+sepsis+nekrosen+
          fiber+nekrosen+nierenvers+ldh+harnst
        +dialyse+ecoli+leukoz+hypotonie+vasopressors,
        data=dat.roughfix, family=binomial)
summary(m3)
exp(coefficients(m3))


# bootstrapeando un odds ratio (fiber)
# variable continua
boot.fun<-function(data,ind) { boot.fix.data <- na.roughfix(dat[ind, ])
m.tmp <- glm(formula = death ~ fiber,
             family = binomial,
             data = boot.fix.data)
salida<-coefficients(m.tmp)[2]
salida}

boot.res<-boot(dat.roughfix$fiber,boot.fun,R=10000)
hist(boot.res$t,
     xlab = "log(odds) ratio of fiber",
     breaks=80, main = "Bootstrap Distribution")
abline(v=boot.res$t0,lty=2,lwd=3,col="red")
boot.ci(boot.res)
# la distribucin de OR es skewed a derecha
# entonces tiene sentido creerle a
# BCa aunque log(OR) normaliza bastante bien la distribucin.
# log(OR)= coef asociado

# jacknife after bootstrap del predictor lineal de una cov continua




boot.res.2<-boot(dat.roughfix,boot.fun.2,R=2000)
fiber.L <- empinf(data=dat.roughfix,
                  statistic=boot.fun.2)
#jack.after.boot(boot.out=boot.res.2,useJ=F,stinf=F)

fiber.an.2<-glm(formula = death ~ fiber,
                family = binomial, data = dat.roughfix[-2,])
summary(fiber.an)
summary(fiber.an.2)

# al no considerar el dato 2, entendiendolo como un dato muy influyente
# se ve que la variable fiber se vuelve mucho mas significativa en el
# analisis individual

# que pasa ahora si elimino ese dato y realizo la regresion multiple
m1 <- glm(death~., data = dat.roughfix, family=binomial)
m2 <- stepAIC(m1,trace=F,direction = "both")
summary(m2)

# aun asi no se elije la variable segun AIC
```

```
# analisis similar ahora con una variable dicotomica: vasopressors


boot.fun.vaso<-function(data,ind) { boot.fix.data <- data[ind, ]
tmp1<-boot.fix.data$vasopressors==1
tmp2<-boot.fix.data$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
while( a==0 || b==0 || c==0 || d==0) {
  muestra<-sample(1:64,replace=TRUE)
  boot.fix.data<-data[muestra,]
  tmp1<-boot.fix.data$vasopressors==1
  tmp2<-boot.fix.data$death==1
  tmp3<-tmp1+tmp2
  a<-sum(tmp3==2)
  b<-sum(tmp2==1)-a
  c<-sum(tmp1==1)-a
  d<-sum(tmp2==0)-c
}
#salida<-c(fisher.ex(a,b,c,d,0.05)[1],fisher.ex(a,b,c,d,0.05)[4])
salida<-fisher.ex(a,b,c,d,0.05)[1]
salida}


boot.res.vaso<-boot(dat.roughfix,boot.fun.vaso,R=10000)
hist(boot.res.vaso$t,breaks=40,main="Distribucion Bootstrap",xlab="Odds ratio para↩
     vasoconstrictor")
abline(v=boot.res.vaso$t0,lty=2,lwd=3,col="red")
ci<-boot.ci(boot.res.vaso)
abline(v=ci$bca[c(4,5)],col="blue",lwd=3)
abline(v=ci$perc[c(4,5)],col="green",lwd=3)
hist(log(boot.res.vaso$t),breaks=20,xlim=c(-1,5),main="Distribucion Bootstrap",↩
    xlab="log(odds) ratio para vasoconstrictor")
abline(v=log(boot.res.vaso$t0),lty=2,lwd=3,col="red")
abline(v=log(ci$bca[c(4,5)]),col="blue",lwd=3)
abline(v=log(ci$perc[c(4,5)]),col="green",lwd=3)
#mediana
orden<-order(boot.res.vaso$t)
med<-orden[5000]
abline(v=log(boot.res.vaso$t[med]),lty=3)
jack.after.boot(boot.out=boot.res.vaso,useJ=F,stinf=F)


# p valor con todos los datos
tmp1<-dat.roughfix$vasopressors==1
tmp2<-dat.roughfix$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
fisher.ex(a,b,c,d,0.05)
abline(v=log(fisher.ex(a,b,c,d,0.05)[c(2,3)]),col="royalblue",lty=4)
```

```
#p-valor sin el dato 50

tmp1<-dat.roughfix[-50,]$vasopressors==1
tmp2<-dat.roughfix[-50,]$death==1
tmp3<-tmp1+tmp2
a<-sum(tmp3==2)
b<-sum(tmp2==1)-a+0.5
c<-sum(tmp1==1)-a
d<-sum(tmp2==0)-c
fisher.test(rbind(c(a,b),c(c,d)))$p.value


# supresion individual de pacientes para ver la influencia en la eleccion
# de la covariable en el analisis univariable
#dialyse
cont<-numeric(64)
odd<-numeric(64)
for (i in 1:64){

  tmp1<-dat.roughfix[-i,]$dialyse==1
  tmp2<-dat.roughfix[-i,]$death==1
  tmp3<-tmp1+tmp2
  a<-sum(tmp3==2)
  b<-sum(tmp2==1)-a
  c<-sum(tmp1==1)-a
  d<-sum(tmp2==0)-c
  cont[i]<-fisher.test(rbind(c(a,b),c(c,d)))$p.value
  odd[i]<-fisher.ex(a,b,c,d,0.05)[1]
}
mean(cont>0.1)

#nekrosen
cont<-numeric(64)
odd<-numeric(64)
for (i in 1:64){

  tmp1<-dat.roughfix[-i,]$nekrosen==1
  tmp2<-dat.roughfix[-i,]$death==1
  tmp3<-tmp1+tmp2
  a<-sum(tmp3==2)
  b<-sum(tmp2==1)-a
  c<-sum(tmp1==1)-a
  d<-sum(tmp2==0)-c
  cont[i]<-fisher.test(rbind(c(a,b),c(c,d)))$p.value
  odd[i]<-fisher.ex(a,b,c,d,0.05)[1]
}
mean(cont>0.1)

#nierenvers
cont<-numeric(64)
odd<-numeric(64)
for (i in 1:64){

  tmp1<-dat.roughfix[-i,]$nierenvers==1
  tmp2<-dat.roughfix[-i,]$death==1
  tmp3<-tmp1+tmp2
  a<-sum(tmp3==2)
  b<-sum(tmp2==1)-a
```

```
  c<-sum(tmp1==1)-a
  d<-sum(tmp2==0)-c
  cont[i]<-fisher.test(rbind(c(a,b),c(c,d)))$p.value
  odd[i]<-fisher.ex(a,b,c,d,0.05)[1]
}
mean(cont>0.1)



# diminuye notablemente el p-valor sin el dato 50
# para dialyse
# incluyo entonces esta digresion en el analisis multiple

m3<-glm(death~nierenvers+nekrosen+vasopressors,
        data=dat.roughfix[-50,], family=binomial)
summary(m3)
summary(stepAIC(m3))

# pareciera nuevamente que en la regresi n m ltiple
# deja de ser influyente esta variable
# Veamos qu  ocurre con OR sin el dato 50 (demasiados 0's)

# An lisis de selecci n de variables
# A continuaci n:
# el error de predicci n con el optimismo calculado
# como en urine
# quizs el 0.632 tambin.
# es importante primero seleccion de variables
# por el tema de missing data
# para el optimismo est  bueno s lo tener en cuenta
# los pacientes con todos
# los datos para las covariables elegidas.



# seleccion de variables

# todo el modelo
# de p-valor m s chico a m s grande
summary(m1 <- glm(death~nierenvers+sepsis+vasopressors+
                    harnst+crea+nekrosen+hypotonie+quirck+
                    age+ldh+leukoz+ck+dialyse+ecoli,
                  data = dat.roughfix, family=binomial))
m2 <- stepAIC(m1,trace=F,direction="both")
summary(m2)
AIC(m2)

## si no incluyo a los pacientes que no tienen
# todos los datos para las variables
## en cuestion
## obtengo identicos resultados
coef.names <- NULL
cont<-numeric(500)
for(i in 1:500)
{
  ind <- sample(1:64, replace = T)
  boot.fix.data <- na.roughfix(dat[ind, ])
  tmp.mod.0 <-  glm(death ~ .,
```

```
                    data=boot.fix.data, family=binomial)
  tmp.mod <- stepAIC(tmp.mod.0, trace=F)
  coef.names <- c(coef.names, names(tmp.mod$coeff))

  tmp1<-boot.fix.data$vasopressors==1
  tmp2<-boot.fix.data$death==1
  tmp3<-tmp1+tmp2
  a<-sum(tmp3==2)
  b<-sum(tmp2==1)-a
  c<-sum(tmp1==1)-a
  d<-sum(tmp2==0)-c
  if (b==0) b<-b+0.5
  if (c==0) c<-c+0.5
  odd<-fisher.ex(a,b,c,d,0.05)[1]
  ifelse ("vasopressors1" %in% names(tmp.mod$coeff),
          cont[i]<-odd,
          cont[i]<-0)
  cat("bootstrap sample number", i, "....................................","\n")
}

table(coef.names)/500

# density del bootstrap de vasopressors en el modelo de seleccion

library(lattice)
vaso.boot.odd<-cont
densityplot(vaso.boot.odd)

#reemplazando por missing data



# eliminando los pacientes con missing data en las variables correspondientes


## no incluyo a los pacientes que no tienen todos los datos para las variables
## en cuestion
datos<-dat[-c(3,4,9,21,23,24,26,30,37,44,54,43,47,21,27,34,51),]
dim(datos)
coef.names <- NULL
for(b in 1:500)
{
  ##ind <- sample(1:64, replace = T)
  ind <- sample(1:51, replace = T)
  ##boot.fix.data <- na.roughfix(dat[ind, ])
  boot.fix.data <- datos[ind, ]
  tmp.mod.0 <-  glm(death ~ age+ck+harnst+hypotonie+ldh+nekrosen+nierenvers+
                      dialyse+leukoz+gpu+hautabl+hb+sex+na,
                    data=boot.fix.data, family=binomial)
  tmp.mod <- stepAIC(tmp.mod.0, trace=F)
  coef.names <- c(coef.names, names(tmp.mod$coeff))
  cat("bootstrap sample number", b, "....................................","\n")
}

table(coef.names)/500
```

```
### con todos los datos




mf<-glm(death~nekrosen+nierenvers,
        data=dat.roughfix,family=binomial)
summary(glm(death~nekrosen+nierenvers,
            data=dat.roughfix,family=binomial))



  # Bootstrap distribution for beta1 beta2 beta3

  library(randomForest)
beta1 <- NULL
beta2 <- NULL

dist.fun<-function(data,ind){
  boot.fix.data <- na.roughfix(data[ind, ])

  m.tmp <- glm(formula = death ~ nekrosen+nierenvers,
               family = binomial,
               data = boot.fix.data)

  beta1 <- coef(m.tmp)[2]
  beta2 <- coef(m.tmp)[3]
  #c(beta1,beta2,beta3)
  beta1  }
library(car)
m1.boot <- Boot(mf, R = 2000)
confint(m1.boot)
dist.boot<-boot(dat,dist.fun,R=2000)
par(mfrow = c(1,2))
beta1<-dist.boot$t
beta1<-beta1[which(beta1<15)]
hist((beta1[which(beta1>-10)]),
     xlab = "log(odds) ratio para renal",
     breaks = 80, main = "Distribucion Bootstrap")
abline(v =boot.ci(dist.boot)$bca[c(4,5)],col="red",lty=2,lwd=3 )
beta2<-dist.boot$t
beta2<-beta2[which(beta2>-15)]

hist((beta2[which(beta2<10)]),
     xlab = "log(odds) ratio para necrosis",
     main = "Distribucion Bootstrap", breaks = 80)
#abline(v = quantile(beta2), prob = c(0.025, 0.5, 0.975),col="red")
beta3<-dist.boot$t
beta3<-beta3[which(beta3<15)]

hist((beta3[which(beta3>-10)]),
     xlab = "log(odds) ratio of renal failure",
     main = "Bootstrap Distribution", breaks = 80)
abline(v =boot.ci(dist.boot)$bca[c(4,5)],col="red",lty=2,lwd=3)

par(mfrow = c(1,1))
```

```
##  Prediccion
## Primero error bootstrap mejorado

mf<-glm(death~nekrosen+nierenvers,
        data=dat.roughfix,family=binomial)


datos<-read.csv2("C:/Users/Gaspard/Desktop/datwork.csv")
datos<-na.roughfix(datos)
cost <- function(r, pi=0) mean(abs(r-pi)>0.5)
app.err <- cost(datos$death, fitted(mf))


dat.pred.fun <- function(data, i, model)
{
  d <- data[i,]
  d<-na.roughfix(d)
  d$dialyse <- factor(d$dialyse)
  d$nekrosen <- factor(d$nekrosen)
  d$nierenvers <- factor(d$nierenvers)
  data$dialyse <- factor(data$dialyse)
  data$nekrosen <- factor(data$nekrosen)
  data$nierenvers <- factor(data$nierenvers)
  d.glm <- update(model,data=d)
  pred <- predict(d.glm,na.roughfix(data),type="response")
  D.F.Fhat <- cost(na.roughfix(data$death), pred)
  D.Fhat.Fhat <- cost(d$death, fitted(d.glm))
  c(na.roughfix(data$death)-pred,D.F.Fhat - D.Fhat.Fhat)
}
dat.boot <- boot(data=datos, dat.pred.fun, R=200, model=mf)
n <- nrow(datos)
opt<-mean(dat.boot$t[,n+1])
err.boot<-opt+app.err
par(mfrow=c(1,1))
hist(dat.boot$t[,n+1],breaks=20,xlab="optimismo boot",
     main="Distribucion Bootstrap")
abline(v=opt,col="red",lty=2,lwd=3)

# la estimacin del optimismo est  lejos de ser buena por
# la dispersion de los datos
## que se observa en el histograma
## al menos da una idea del posible sesgo en el error aparente.
# (ligeramente
## likely to be positive)

#### error 0.632

dat.diag <- glm.diag(mf)
dat.boot$f <- boot.array(dat.boot)
ord <- order(dat.diag$res)
dat.pred <- dat.boot$t[,ord]
err.632 <- 0
n.632 <- NULL
pred.632 <- NULL
for (i in 1:n) {
  inds <- dat.boot$f[,i]==0
```

```
  err.632 <- err.632 + cost(dat.pred[inds,i])/n
  n.632 <- c(n.632, sum(inds))
  pred.632 <- c(pred.632, dat.pred[inds,i])
}

err.632 <- 0.368*app.err + 0.632*err.632
dat.fac <- factor(rep(1:n,n.632),labels=ord)
plot(dat.fac, pred.632,ylab="Errores de prediccion",
     xlab="Casos ordenados por residuos")
abline(h=-0.5,lty=2)
abline(h=0.5,lty=2)



#Repitamos todo el proceso en completo sin el dato 50


dat<-read.csv2("C:/Users/Gaspard/Desktop/datwork.csv")
dat50<-dat[-50,]




# llevo algunos a factores

dat50$death <- factor(dat50$death)
dat50$sex <- factor(dat50$sex)
dat50$dialyse <- factor(dat50$dialyse)
dat50$nekrosen <- factor(dat50$nekrosen)
dat50$hautabl <- factor(dat50$hautabl)
dat50$blasenbild <- factor(dat50$blasenbild)
dat50$hypotonie <- factor(dat50$hypotonie)
dat50$tachykardie <- factor(dat50$tachykardie)
dat50$nierenvers <- factor(dat50$nierenvers)
dat50$sepsis <- factor(dat50$sepsis)
dat50$lokalisation <- factor(dat50$lokalisation)
dat50$typ <- factor(dat50$typ)
dat50$adipositas <- factor(dat50$adipositas)
dat50$asa <- factor(dat50$asa)
dat50$ecoli <- factor(dat50$ecoli)
dat50$vasopressors <- factor(dat50$vasopressors)

library(randomForest)
dat.roughfix50 <- na.roughfix(dat50)




coef.names <- NULL
for(b in 1:500)
{
  ind <- sample(1:64, replace = T)
  boot.fix.data <- na.roughfix(dat50[ind, ])
  tmp.mod.0 <-  glm(death ~ nierenvers+sepsis+vasopressors+
                      harnst+crea+nekrosen+hypotonie+quirck+
                      age+ldh+leukoz+ck+dialyse+ecoli ,
                    data=boot.fix.data, family=binomial)
  tmp.mod <- stepAIC(tmp.mod.0, trace=F)
```

```
  coef.names <- c(coef.names, names(tmp.mod$coeff))
  cat("bootstrap sample number", b, "....................................","\n")
}

table(coef.names)/500



mf2<-glm(death~nekrosen+dialyse+nierenvers+age,
         data=dat.roughfix50,family=binomial)
summary(mf2)
```

10.2 Tablas de datos

10.2.1 Datos del ejemplo 6MWT del Capítulo 4

10.2.2 Datos del Capítulo 8

10.2.3 Datos del Capítulo 9

x.6MWD	peakVO2
572.00	402.00
485.00	646.00
665.00	480.00
378.00	517.00
463.00	559.00
529.00	116.00
654.00	634.00
395.00	193.00
505.00	765.00
434.00	480.00
470.00	408.00
574.00	647.00
500.00	432.00
732.00	403.00
673.00	305.00
560.00	445.00
535.00	634.00
575.00	450.00
575.00	423.00
322.00	482.00
527.00	622.00
620.00	453.00
625.00	505.00
580.00	381.00
510.00	530.00

x.6MWD	peakVO2
288.00	554.00
254.00	554.00
460.00	448.00
315.00	353.00
591.00	611.00
450.00	536.00
526.00	383.00
400.00	595.00
540.00	415.00
492.00	623.00
561.00	515.00
398.00	680.00
679.00	483.00
660.00	548.00
505.00	599.00
420.00	550.00
428.00	406.00
625.00	633.00
640.00	470.00
534.00	408.00
420.00	542.00
593.00	593.00
607.00	696.00
268.00	600.00
312.00	370.00
485.00	454.00

x.6MWD	peakVO2
14.50	18.40
18.60	28.50
29.90	11.30
20.60	19.90
16.40	19.80
25.50	8.50
27.10	36.00
15.30	6.40
18.80	27.20
16.20	12.10
11.10	13.40
32.70	25.80
24.00	18.20
36.20	10.00
20.80	10.60
27.00	20.70
26.50	29.00
17.30	20.30
23.00	19.70
13.70	11.80
18.90	20.20
22.00	18.70
30.50	16.40
18.00	20.20
31.90	15.60
15.70	21.60
9.20	20.20

x.6MWD	peakVO2
16.00	12.90
12.20	14.40
29.00	31.10
13.50	22.80
13.60	16.80
9.50	16.90
16.20	17.80
22.00	27.90
20.20	32.70
19.50	28.60
34.70	28.80
22.00	22.30
26.40	30.50
22.20	17.20
15.80	19.30
24.40	33.90
33.90	16.90
24.80	13.60
20.20	19.40
28.00	26.50
19.30	26.70
16.80	20.30
8.20	14.00
17.80	17.10

Hgb_Admit	anemie2	sex	age	CKD	DM	KHK	LVEF	Dialyse	ln.crea	ln.WBC	ln.CRP
12.40	0	w	85	1	0	1	0	0	0.10	1.19	-0.22
11.60	1	w	60	1	1	1	0	1	0.26	1.77	0.34
12.70	0	w	84	1	0	0	1	0	0.18	1.96	-0.92
9.10	1	w	79	1	0	1	1	0	0.10	1.74	0.74
9.30	1	w	80	0	1	0	1	0	-0.22	2.10	-0.11
10.00	1	w	79	0	1	1	0	0	-0.36	2.00	-0.51
14.30	0	w	76	1	0	1	1	0	0.26	1.67	-0.92
12.30	0	w	79	1	1	0	1	0	0.10	2.08	-0.92
8.60	1	w	84	0	1	0	0	0	-0.22	2.12	0.18
11.20	1	w	84	0	1	1	1	0	0.00	1.82	0.34
12.30	1	m	79	0	0	1	0	0	0.18	1.93	-0.92
12.30	0	w	86	0	0	0	0	0	-0.22	1.93	-0.92
11.30	1	w	66	1	0	1	0	1	1.84	1.70	-0.92
11.80	1	m	85	1	0	1	0	0	0.26	1.63	-0.92
9.60	1	w	89	1	0	1	0	0	0.18	1.70	0.59
12.20	0	w	90	0	0	1	1	0	-0.22	1.77	0.34
13.30	0	w	75	0	0	0	1	0	-0.36	2.20	0.10
10.40	1	m	75	0	0	0	0	0	0.00	1.69	-0.92
10.30	1	w	84	1	0	0	1	0	0.18	1.25	-0.92
13.80	0	m	80	1	0	1	1	0	0.74	2.16	1.36
11.80	1	m	76	0	0	1	1	0	-0.22	2.05	-0.69
15.30	0	w	93	0	1	0	1	0	0.26	2.07	1.74
13.80	0	m	86	1	0	1	0	0	0.34	1.81	-0.92
12.50	0	w	87	0	0	0	0	0	-0.11	2.03	0.00
12.10	1	m	83	0	1	1	0	0	0.18	1.81	-0.92
14.70	0	m	77	0	0	1	1	0	0.00	2.01	-0.92
14.00	0	m	74	0	1	1	1	0	0.10	2.12	0.00
11.50	1	w	85	1	0	0	1	0	0.10	1.36	-0.51
10.00	1	w	84	1	0	0	0	0	0.26	0.96	-0.11
11.20	1	w	86	0	0	0	0	0	-0.22	1.70	-0.69
9.80	1	m	76	1	0	1	0	1	1.72	2.16	0.88
9.20	1	w	74	0	1	1	1	0	0.00	2.07	2.05
11.40	1	w	83	1	0	0	0	0	0.10	1.82	-0.51
11.20	1	w	75	1	0	0	0	0	0.26	2.12	-0.11
13.00	0	m	91	1	0	1	1	0	0.92	1.84	0.99
11.30	1	m	83	0	0	1	1	0	0.26	1.77	0.34
12.70	1	m	75	1	0	1	1	0	0.34	1.81	-0.92
10.70	1	w	95	1	0	1	0	0	0.69	2.27	-0.92
11.50	1	m	92	1	1	1	1	0	0.47	1.48	-0.92
11.80	1	w	78	0	0	0	1	0	-0.36	1.70	-0.51
11.40	1	w	87	0	0	1	0	0	-0.22	2.09	-0.92
10.90	1	w	82	1	0	0	0	0	0.26	1.57	-0.92

Hgb_Admit	anemie2	sex	age	CKD	DM	KHK	LVEF	Dialyse	ln.crea	ln.WBC	ln.CRP
11.00	1	m	84	0	0	0	1	0	0.00	1.55	-0.51
13.20	0	w	82	0	1	0	0	0	-0.51	2.14	-0.22
12.40	0	w	89	1	0	1	0	0	0.34	1.70	-0.92
11.30	1	w	89	1	0	1	0	0	0.10	2.35	2.01
10.80	1	w	78	0	0	1	0	0	-0.22	1.89	-0.92
12.10	0	w	79	0	1	1	1	0	0.00	1.48	-0.36
11.40	1	m	68	0	0	1	1	0	0.10	3.49	-0.22
11.30	1	m	82	1	1	1	1	1	1.48	2.42	2.78
11.10	1	m	86	0	0	1	0	0	0.00	1.89	-0.92
12.90	0	w	82	0	0	1	0	0	-0.36	1.76	-0.92
11.20	1	w	85	1	0	0	0	0	0.34	1.50	-0.92
11.90	1	m	77	0	1	1	0	0	-0.11	1.84	-0.92
12.90	1	m	80	0	0	1	1	0	0.00	1.76	-0.92
11.00	1	m	86	0	0	1	0	0	0.18	1.74	0.34
10.50	1	m	73	1	0	0	0	1	2.14	1.86	-0.69
13.10	0	w	82	1	0	1	1	0	0.00	2.05	-0.51
16.00	0	m	78	1	1	1	1	0	0.64	1.74	0.64
11.50	1	m	73	1	1	1	1	0	0.53	1.55	-0.22
13.20	0	m	79	1	0	0	1	0	0.26	1.53	-0.92
14.90	0	m	75	1	1	1	1	0	0.41	1.90	-0.92
14.40	0	m	85	0	0	1	0	0	0.18	1.90	-0.92
8.70	1	m	89	1	0	1	0	0	0.41	2.01	-0.36
12.60	1	m	82	0	0	1	0	0	0.26	1.99	-0.92
12.70	1	m	83	1	1	0	1	0	0.59	2.50	0.10
12.10	1	m	84	1	0	1	1	0	0.53	1.96	-0.92
9.90	1	w	87	0	0	1	1	0	-0.22	1.77	-0.92
10.90	1	m	84	1	0	1	1	1	1.81	1.86	-0.51
12.00	1	m	84	1	0	1	1	0	0.59	1.63	-0.51
10.10	1	m	89	1	0	1	0	0	0.64	2.05	-0.36
10.40	1	m	67	0	0	0	0	0	0.26	2.01	0.41
11.60	1	w	81	1	0	0	0	0	0.18	2.32	1.50
15.10	0	w	81	1	0	1	1	0	0.10	2.67	1.77
11.30	1	w	88	0	0	0	0	0	-0.22	1.79	-0.92
11.00	1	w	90	0	0	0	1	0	-0.11	1.25	-0.92
12.00	1	m	86	1	1	1	0	0	0.83	1.97	-0.92
11.80	1	w	82	1	0	1	1	0	0.26	2.00	0.53
10.80	1	w	86	1	0	0	0	0	0.26	1.89	-0.69
11.80	1	m	73	0	1	1	0	0	-0.22	1.19	-0.92
13.20	0	m	81	1	1	1	0	0	0.26	1.82	-0.92
11.00	1	m	89	1	1	1	0	0	0.79	2.14	0.47

Hgb_Admit	anemie2	sex	age	CKD	DM	KHK	LVEF	Dialyse	ln.crea	ln.WBC	ln.CRP
14.50	0	m	78	0	0	1	0	0	0.18	1.59	-0.11
11.60	1	w	85	0	0	1	1	0	-0.22	1.50	-0.92
10.90	1	m	89	1	0	1	1	0	0.79	2.19	1.53
13.10	0	w	72	1	0	1	0	0	0.18	2.05	0.41
10.20	1	w	75	1	0	0	1	0	0.18	2.03	0.47
17.10	0	m	88	0	0	1	1	0	0.10	2.01	-0.92
14.10	0	w	81	0	1	1	0	0	-0.11	1.72	-0.92
12.70	1	m	76	1	0	1	0	0	0.47	2.08	-0.69
12.50	0	w	80	0	0	1	0	0	0.00	1.57	-0.92
10.80	1	w	85	1	0	0	0	0	0.00	1.63	-0.92
12.70	0	w	76	0	1	1	0	0	-0.11	2.21	-0.92
10.40	1	w	80	0	0	1	0	0	-0.22	1.19	0.10
10.90	1	m	74	0	0	1	1	0	0.18	2.12	0.10
9.10	1	w	83	1	0	1	1	0	0.47	2.14	-0.36
12.30	0	w	76	1	0	1	1	0	0.18	1.97	-0.92
14.20	0	m	80	1	0	1	1	0	0.53	1.77	0.10
12.80	1	m	84	0	1	1	1	0	0.18	1.86	-0.92
12.00	0	w	88	0	1	1	1	0	-0.11	2.13	0.53
8.20	1	m	83	1	0	1	1	1	1.36	2.37	2.29
12.50	0	w	81	0	1	0	0	0	0.00	2.12	0.59
13.50	0	w	77	0	1	0	1	0	-0.11	2.35	0.26
9.70	1	m	88	1	0	0	0	0	0.64	1.39	1.96
10.50	1	w	65	1	0	1	1	1	1.41	1.70	0.92
11.30	1	w	85	1	0	1	1	0	0.74	2.20	0.59
11.50	1	m	83	0	0	0	1	0	0.10	2.19	-0.92
14.50	0	m	74	1	0	1	1	0	0.64	1.84	-0.92
13.80	0	m	82	1	0	1	1	0	0.34	2.58	0.18
10.30	1	m	86	0	0	0	0	0	0.00	1.69	-0.92
13.50	0	w	84	1	0	1	1	0	0.34	2.03	-0.92
11.60	1	w	78	0	0	1	1	0	-0.51	2.01	-0.92
15.30	0	m	83	1	0	1	1	0	0.53	1.90	-0.92
11.90	1	m	78	1	0	1	1	1	1.55	1.50	0.00
12.20	0	w	83	0	1	1	1	0	-0.36	2.08	0.83
10.90	1	m	86	0	0	1	1	0	0.10	1.63	-0.92
13.10	0	w	81	0	0	1	1	0	-0.69	1.77	-0.92
13.40	0	m	81	0	0	1	1	0	-0.22	2.01	-0.92
12.60	1	m	91	1	0	1	1	0	0.41	1.70	0.64
12.90	0	w	75	0	0	0	1	0	-0.11	1.34	-0.11
11.10	1	w	82	1	1	0	0	0	0.53	1.55	-0.92

Hgb_Admit	anemie2	sex	age	CKD	DM	KHK	LVEF	Dialyse	ln.crea	ln.WBC	ln.CRP
10.70	1	m	83	1	0	1	1	0	0.74	2.31	0.69
11.80	1	w	84	1	1	1	1	0	0.00	2.01	-0.92
9.70	1	w	80	1	0	0	1	0	0.64	2.71	1.59
11.40	1	m	87	0	0	1	1	0	0.18	2.10	1.06
11.90	1	w	89	1	0	1	1	0	0.10	1.69	-0.92
16.20	0	m	72	1	1	1	0	0	0.59	2.30	-0.92
12.40	0	w	76	1	0	0	1	0	0.10	1.87	1.31
14.40	0	w	76	0	1	0	0	0	0.00	2.93	-0.92
10.70	1	m	73	1	1	0	1	1	1.63	1.59	-0.69
11.90	1	w	85	0	0	0	1	0	-0.11	1.86	-0.92
9.80	1	m	84	1	0	1	0	0	0.26	3.14	2.29
11.50	1	w	87	1	1	0	0	0	0.59	2.71	1.61
8.60	1	m	81	1	0	0	1	0	1.03	1.93	-0.11
15.00	0	m	73	0	1	1	0	0	0.18	2.87	1.92
11.10	1	w	86	0	0	1	1	0	0.00	1.89	-0.92
12.70	0	w	78	1	0	1	1	0	0.10	1.74	0.26
11.30	1	m	85	1	1	1	0	0	0.59	1.53	-0.69
13.50	1	m	89	0	0	0	0	0	0.18	1.99	-0.69
11.20	1	w	74	0	1	0	1	0	0.00	1.65	-0.69
12.60	0	w	83	1	1	1	1	0	0.47	2.13	0.53
9.80	1	m	77	0	0	1	1	0	0.26	1.92	1.99
13.20	0	w	86	0	0	1	1	0	-0.11	2.24	-0.92
11.20	1	w	88	0	0	0	1	0	-0.22	1.53	-0.92
9.70	1	w	91	0	0	1	1	0	-0.22	1.95	0.64
13.40	0	m	81	0	1	0	0	0	0.00	2.42	-0.51
10.00	1	m	79	1	1	1	1	0	0.34	1.77	1.61
12.50	0	w	90	0	0	1	1	0	-0.22	1.93	-0.69
11.20	1	w	79	1	1	1	1	0	0.53	1.81	-0.36
15.30	0	m	71	0	0	1	1	0	0.00	2.92	-0.69

age	death	sex	fiber	leukoz	cpr	beat.stage	dialyse	asa	bmi	adipositas	nekrosen
49	0	m	39,6	9500	58,1	19	0	4	32,3	1	1
68	1	w	41	20100	49,7	61	1	4	51,6	1	1
57	0	m	38,8	9300		0	0	2	29,3	0	1
51	0	m	36,5	13300	47,8	7	1	3	24,7	0	0
33	0	w		15000	13,4	11	0	4	27,8	0	1
50	1	m	38	17100	38,5	11	1	4	34,3	1	1
36	0	w	38	12900	24,69	31	0	3	34	1	0
40	1	m	37,8	19600	26,5	16	1	4	30,4	1	1
34	0	w	40,7	12700	7,1	0	0	2	23,5	0	0
64	0	w		26700	14,8	0	0	2	26,4	0	0
65	1	m	39	37260	24,4	16	1	4	30	1	1
48	1	m	35,1	5500	24,3	1	0	5	26,3	0	0
39	0	m		2200	35,2	17	0	4	20,23	0	1
70	0	m	39,1	16600	39,7	0	0	3	31,1	1	0
59	0	w		21500	29,3	6	0	4	35,2	1	0
67	1	w	38,6	8400	54,5	3	0	4	31,2	1	0
72	0	m		23600	33,7	17	1	4	24,1	0	0
38	1	m		24890	25,2	15	0	4	20,1	0	0
82	1	w	36,5	14500	42	5	0	4	18,5	0	0
67	1	w		23600	42,6	58	1	3	44,1	1	1
38	0	m		28200	27,2	3	1	4	18,9	0	1
45	0	w		15320	37,5	40	0	4	22	0	0
47	0	w		19700	21,8	0	0	3	31,1	1	0
38	0	m		18200	26,2	0	0	3	22,1	0	0
60	0	m	39,5	19100	16	63	1	4	32,1	1	0
77	1	m		12400	13,8	6	0	4	19,04	0	1
69	1	m	39,4	9200	23,2	18	1	4	29,1	0	0
71	1	m		9200	38,3	3	1	4	23,1	0	1
72	0	m	36,5	6300	37,8	13	1	4	23,1	0	0
61	0	m		34700	27,2	0	0	3	26,87	0	0
70	0	m	38,5	24200	23,2	11	1	5	34	1	0
42	0	w		27000	50,9	53	1	4	31,2	1	0
58	0	m		10800	20,6	24	1	4	29,1	0	1
21	0	w	39	18500	42,2	0	0	2	23,9	0	0
43	1	m		1900	27,9	3	0	4			0
47	1	m	38	1400	34,5	28	1	5	25,2	0	0
48	0	m		59570	47,97	5	0	4	24,9	0	1
67	1	w	37,9	1100	9,7	1	0	4	27	0	1
62	0	w		13300	33	3	0	2	25,3	0	0

age	death	sex	fiber	leukoz	cpr	beat.stage	dialyse	asa	bmi	adipositas	nekrosen
70	0	m		8660	42,41	26	1	4	29,2	0	0
62	1	w		9300	22,5	6	0	3	40,4	1	0
57	0	m	38	3800	27,3	0	0	3	33,2	1	0
53	0	w		13600	12,5	0	0	3	18	0	0
45	1	m	38,8	8800	30,6	4	0	4	24,7	0	1
32	0	m	38,8	17500	25,6	0	0	3	26,8	0	0
64	0	m	40	10200	12,7	26	0	4	32,4	1	1
70	0	m	38,1	19000		0	0	3	24,2	0	0
54	0	m	38	28800	13,2	1	0	2	29,88	0	1
60	1	w		1470	46,13	1	0	5	31,2	1	1
65	1	m	37,5	17000	34,8	0	0	3	22,6	0	1
76	0	m		27720	25,13	3	0	3	31,1	1	
42	0	m		17700	39,2	5	0	4	21,2	0	1
31	0	m	38	14300	1,6	0	0	2	22,2	0	0
23	0	w	40	26800		0	0	2	29,1	1	0
50	0	m	39,7	31900	31,8	0	0	4	28,6	0	0
69	0	m	38	2600	11,8	0	0	3	20	0	0
45	0	m	37,4	18900	39	21	0	3	21,4	0	1
58	0	m	39,6	19300	20,6	0	0	3	45,3	1	0
44	0	w	39,5	19400	30,1	0	0	3	36,3	1	0
44	1	m	39	17100	2,5	20	1	4	29,3	0	0
64	1	m	36,5	16500	9,6	64	1	5	23,2	0	1
47	0	m		11500	10,6	0	0	2	29,21	0	0
60	0	m		20400	30,7	30	1	3	27,7	0	0
36	0	m	40	13400	51,1	29	1	4	26,3	0	0

hautabl	blasenbild	hypotonie	tachykardie	nierenvers	sepsis	lokalisation	typ	ck	ldh
0	0	1	1	0	1	3	3	196	269
0	0	1	1	1	1	1	2	128	271
1	0	0	0	0	0	1	2		268
0	0	1	1	1	1	1	2	118	212
1	0	1	1	0	1	3	2	8	193
0	0	1	1	1	1	3	1	144	221
0	0	1	1	0	1	1	1	3165	165
1	1	1	1	1	1	3	2	429	128
0	0	0	0	0	0	3	1		202
0	1	0	0	0	0	1	1	24	171
0	0	0	1	1	1	2	1	479	526
1	1	1	1	1	1	1	1	1935	347
0	1	1	1	0	1	1	1	313	211
0	0	0	1	0	0	2	1	84	177
0	0	1	1	0	1	3	2	59	268
0	0	1	1	1	1	3	1	3313	432
0	0	1	1	1	1	3	1	225	199
0	0	1	1	1	1	3	1	785	697
0	0	1	0	1	1	2	1	464	482
0	0	0	0	1	1	2	1	442	302
0	0	0	0	1	0	3			
0	0	1	1	0	1	3	1	548	182
0	0	0	0	0	0	1	1		204
0	0	0	0	0	0	1	1	2049	444
0	1	0	1	1	1	1	1	63	336
0	1	0	0	1	1	1	1		
0	0	0	0	1	1	1	1	83	262
0	0	0	1	1	1	3	1	599	259
0	0	1	1	1	1	3	1	2815	1283
0	0	0	0	0	0	2	1		435
0	0	1	1	1	1	2	1	99	233
0	0	0	1	0	1	3	1	641	334
0	0	1	1	1	1	1	1	837	174
0	1	0	0	0	0	1	1	462	
0	0	1	1	1	1	1	1	1226	303
0	0	1	1	1	1	1	2	6687	500
1	0	0	1	0	1	1	1		
0	0	1	1	0	1	3	1	486	106
0	0	0	0	0	0	3	1	50	189

hautabl	blasenbild	hypotonie	tachykardie	nierenvers	sepsis	lokalisation	typ	ck	ldh
0	0	1	1	1	1	2	1	4758	411
0	0	1	1	1	1	3	1	75	203
0	0	0	0	0	0	3	1	85	132
0	0	0	0	0	0	3	1	22	
0	0	1	1	1	1	3	1		281
0	1	0	1	0	0	1	2	146	198
0	0	1	1	0	1	3	1	40	222
0	0	0	1	0	0	1	1	8	254
0	0	0	0	0	0	1	1	55	176
1	0	1	1	1	1	3	2	1922	308
0	0	0	1	0	0	1	1	60	204
		0	1	0	1	2	0	86	172
1	1	0	1	1	1	1	2	2863	241
0	0	1	1	0	0	1	0	481	191
0	0	0	1	0	1	1	2		173
0	0	0	1	0	0	2	0	8	254
0	0	0	0	0	0	2	1	9	185
0	0	0	1	0	1	2	1	343	314
0	0	0	0	0	0	2	1	122	146
0	0	0	0	0	0	1	1	174	443
0	0	1	1	1	1	1	1	42869	1221
1	1	1	1	1	1	3	3	10014	1053
0	0	0	0	0	0	1	1	310	229
0	0	1	1	1	1	1	2	892	390
0	0	1	1	1	1	1	2	798	347

bili	crea	harnst	lact	thromb	na	quirck	fibrino	hb	ecoli	vasopressors
3,8	1,5	63		225	147	73	63	14,7	0	1
0,3	5,1	196,2		172	137	57	586	11,2	0	1
			3,3	132	139			6,9	0	0
	4,9	147	1,8	191	144	44	1082	9,5	0	1
0,2	0,5	26	1,5	36	153	47	882	9,3	0	1
1,6	2,6	153	4,3	289	126	51	696	10,1	0	1
2,5	0,6	53	5,2	263	135	49	1143	10	0	1
1,7	1,3	44	5,9	137	108	40	1145	15,6	0	1
0,4	0,6	9	0,5	222	138	75	503	9	0	0
	0,5	25		297	132	95	705	8,2	0	0
4,58	1,9	141	2	367	129	68		6,8	1	1
3,8	5,7	134	11	234	124	58	859	14,6	0	1
0,61	1	58		166	123	83	493	12,7	0	1
1,3	1,5	71	4,3	123	130	93	947	17,2	0	0
1,3	0,8	28		242	136	82		14,7	0	1
0,65	1,94	94	9,5	591	124	66	964	8,7	1	1
	4,1	196	2,4	509	138	15		7,7	0	1
	4,5	128,4	3,1	151	123	70		13,1	0	1
0,6	3,2	174		320	143	101		10,3	0	1
0,5	2	166		340	144	95		11	1	1
13,59	5	75		307	131	76	594	9,3	0	1
1,26	0,4	30	5,74	105	136	72	360	9,1	0	1
	0,4	16		380	135	96		9,9	1	0
	0,92	33		181	128	101		14	0	0
0,68	1,2	43	1,6	175	134	68	706	8,1	0	1
0,28	1,8	240		95	135	42		11,8	0	1
0,41	1,1	86		271	127	77	924	14,6	0	1
0,77	1,8	160	2,6	229	156	83	950	11,3	0	1
2,23	3,34	132		65	139	55	763	9,5	0	1
0,67	1	72		382	133	61	612	12,2	1	0
0,4	3,4	124	1	367	141	79	881	10,1	1	1
0,2	2	108		316	124	58	799	11,8	0	1
2,5	1,9	127	3,2	80	141	80	660	9,1	1	1
	0,7	26		300	145	60	878	10,2	0	0
10,58	2,7	73		42	141	55	399	10,4	1	1
0,9	6,5	117	10,3	101	131	68	679	14,3	0	1
	1,2			830	131	76		11,1	0	1
1,1	1	33	11,7	108	150	21	96	8,7	0	1
0,2	1,6	89		443	132	85	1438	14	0	1

bili	crea	harnst	lact	thromb	na	quirck	fibrino	hb	ecoli	vasopressors
0,79	3,16	207	34,4	141	134	48	830	12,6	0	1
0,7	0,33	17		168	132	78	755	11,8	1	1
1,9	0,7	37		118	140	84	677	12,1	0	0
0,2	0,6			10	142	101	727	11	0	0
	0,9	75		28	139	65	373	10,7	1	1
0,2	1	28		260	140	80		14,9	0	0
0,9	1	53	1,58	235	145	108	806	9,5	1	1
	1,43							11,8	0	0
0,5	0,7	21		431	138	108	668	9,4	0	1
0,51	3,08	112	50	29	130	51		13,3	0	1
0,48	0,9	64	1,4	367	135	95	823	11,4	0	0
1,7	2,17	79	1,2	385	135	63	729	13,1	0	1
	2,6	211		207	141	75	883	9,6	0	1
0,6	1	27	7,99	138	138	90		9,4	0	1
				256	139			12,8	0	0
0,6	1	35,8		230	132	75	1010	15,7	0	0
0,56	0,6	9		93	136	114		11,4	0	0
0,7	1,3	106,2		311	130	57		12,9	1	1
	1,2	13		191	135	79		14,6	0	0
0,8	1,5	172		174	131	72	502	10,5	0	0
11,58	2,9	47	14,1	43	136	15	96	12,6	1	1
2,6	5	171	5,6	36	139	40	212	5,7	0	1
0,23	0,8	21	2,2	619	138	108	574	12,7	0	0
4,08	2,49	50	8,7	120	132	53	819	16	0	1
1,3	5,32	137		209	138	48	523	12,5	0	1

got	gpu
24	13
13	16
43	
13	10
35	18
139	59
56	40
24	47
29	41
47	13
80	33
52	59
18	54
37	53
131	33
36	23
99	36
241	143
41	39
137	42
41	19
15	
142	
49	21
39	22
30	37
97	52
670	364
90	62
117	80
53	18
53	54
84	29
172	56
135	46
64	18
13	16

got	gpu
196	88
27	26
17	22
30	26
11	10
4	7
16	15
74	150
55	29
19	19
71	46
229	66
20	15
14	20
8	7
66	29
11	14
106	136
893	121
413	179
18	17
186	203
	25

Bibliografía

[1] Davison, A. C. y Hinkley, D. V. (1997). *Bootstrap methods and their application.* Cambridge: Cambridge university press.

[2] Diciccio T. y Efron B. (1992). More accurate confidence intervals in exponential families. *Biometrika.* **79**, 231-245.

[3] Diciccio T. y Efron B (1996). Bootstrap Confidence Intervals. *Statistical Science.* **11**, 133-137.

[4] Efron, B. y Tibshirani, R. J. (1993). *An Introduction to the Bootstrap.* New York: Chapman and Hall.

[5] Efron, B. (1987). Better Bootstrap Confidence Intervals. *Journal of the american statistical association.* **82**, 171-185.

[6] Efron, B. y Gong, G. (1983). A leisurely look at the bootstrap, the jacknife and the cross validation. *The American Statitian.* **3**, 36-48.

[7] Efron, B. y Tibshirani, R. (1986). Bootstrap Methods for Standard Errors, Confidence Intervals and Other Measures of Statistical accuracy. *Statistical Science.* **1**, 54-77.

[8] Fox, J. (2002). *An R and S-PLUS companion to applied regression.* Sage publications, Inc.

[9] Fox, J. y Weisberg, S. (2012). Bootstrapping Regression Models in R. An Appendix to An R Companion to Applied Regression, second edition.

[10] Freedman, D. A. (2005). *Statistical models. Theory and practice.* Cambridge university press.

[11] George, E. (2012). Bayesian Variable Selection: Past, Present and Future Developments. Lecture I - The Variable Selection Problem. *University of Pennsylvania. Erasmus MC.*

[12] Gelman, A. y Hill J. (2007). *Data analysis using regression and multilevel/hierchical models.* Cambridge university press.

[13] Hall, P. (1992). *The Bootstrap and Edgeworth Expansion.* Springer science.

[14] Hinkley, D. y Kuonen, D. (2005). An introduction to the bootstrap with application in R. *Statistical computing and statistical graphics newsletter.* **13**, 6-11.

[15] Hosmer, D. W. y Lemeshow, S. (1989) *Applied logistic regression.* John Wiley & Sons, Inc.

[16] Kehmeier E., Petersen M., Galonska A., Zeus T., Verde P. y Kelm M. (2014) Diagnostic Value of the Six-Minute Walk Test (6MWT) in Grown-Up Patients with Congenital Heart Disease (GUCH): Comparison with Clinical Status and Functional Exercise Capacity. *Int J Cardiol IF 6.175.*

[17] Krieg A., Dizdar L., Verde P. E. y Knoefel W. T. (2014). Predictors of mortality for necrotizing soft-tissue infections: a retrospective. *Langenbecks Arch Surg.* **4**, 333-341.

[18] Stine, P. (1989). An introduction to the bootstrap Method. Examples and ideas. *Sociological Methods and research.* **18**, 243-291.

[19] Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society.* **58**, 267-288.

[20] Verde, P. E. (2014). Statistical inference with computer simulation: an introduction to bootstrap analysis with R. *Coordination Center for Clinical Trials. University of Duesseldorf.*