Regularización, Ridge y LASSO

Modelo Lineal

María Eugenia Szretter Noste

Instituto de Cálculo

FCEyN, UBA

Otros estimadores – Métodos de penalización o regularización

¿Por qué querríamos usar otros procedimientos de estimación además de mínimos cuadrados?
Para obtener mejoras en

- la precisión de la predicción
- la interpretabilidad del modelo

Precisión de la predicción

Si la verdadera relación entre la respuesta y los predictores es aproximadamente lineal, los estimadores LS tendrán un sesgo bajo.

- Si $n \gg p$ entonces las estimaciones de mínimos cuadrados tienden a tener también una varianza baja, y por lo tanto darán buenas predicciones en las observaciones de testeo.
- Sin embargo, si n no es mucho más grande que p, entonces puede haber mucha variabilidad en el ajuste de mínimos cuadrados, resultando en un sobreajuste del estimador de mínimos cuadrados, y, en consecuencia, malas predicciones sobre observaciones de testeo.Y si $p > n$, entonces ya no hay una estimación única del coeficiente de mínimos cuadrados: la varianza es infinita, OLS no se puede usar. Una solución consiste en restringir a los estimadores: conseguimos estimadores sesgados, que pueden redundar en una reducción de varianza y mejoras sustanciales en la precisión con la que podemos predecir la respuesta para las observaciones no utilizadas en el entrenamiento modelo.

Interpretabilidad del modelo

A veces, una o varias variables utilizadas en un modelo de regresión múltiple no están asociadas con la respuesta. Incluir tales variables irrelevantes conduce a innecesarias complejidad en el modelo resultante. Al eliminar estas variables (ajustando a cero las estimaciones del coeficiente correspondiente) obtenemos un modelo más fácil de interpretar. OLS no suele producir estimaciones de coeficientes que son exactamente cero.

Existen algunos enfoques para hacer esta selección automática:

- selección de variables
- métodos de penalización o regularización

Ridge Regression

Cuando hay muchas variables correlacionadas en un modelo de regresión lineal:

- los coeficientes pueden ser muy mal estimados por LS: un coeficiente positivo extremadamente grande en una variable puede ser compensado por un coeficiente negativo del mismo orden de magnitud en una variable muy correlacionada
- los estimadores de LS tienen alta variabilidad

Ejemplo Archivo `simulaciondatoscorrelacionados.R`

Veamos un ejemplo. Simulamos los datos con tamaño de muestra $n = 100$,
$X_1 \sim U(0;5)$
$X_2 \sim U(0;5)$
$X_3 = 2X_1 + U_3$, con $U_1 \sim U(-0{,}1; 0{,}1)$
$X_4 = -X_1 + U_4$, con $U_4 \sim U(-0{,}1; 0{,}1)$
$Y = 8X_1 - 5X_2 + X_3 + 4X_4 + 5 + \varepsilon$, con $\varepsilon \sim N(0,1)$
$X_1, X_2, U_3, U_4, \varepsilon$ son variables independientes
Es decir, la respuesta Y depende de dos covariables independientes, X_1, X_2 y las otras dos X_3, X_4 son casi colineales con X_1. Repetimos el experimento $B = 1000$ veces. Cada vez ajustamos los dos modelos lineales por LS

$$E(Y|\mathbf{X}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 \quad (1)$$

$$E(Y|\mathbf{X}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \quad (2)$$

Covariables muy correlacionadas

```
> cor(cbind(x1,x2,x3,x4))
            x1          x2          x3          x4
x1  1.00000000  0.02381522  0.99981176 -0.99921380
x2  0.02381522  1.00000000  0.02381281 -0.02252836
x3  0.99981176  0.02381281  1.00000000 -0.99916009
x4 -0.99921380 -0.02252836 -0.99916009  1.00000000
```

Resultados de la simulación

Para cada una de las repeticiones, contamos cuáles de los coeficientes estimados resulta significativamente distinto de cero, a nivel 0.05. Obtuvimos

Coeficientes significativos en 1000 repeticiones

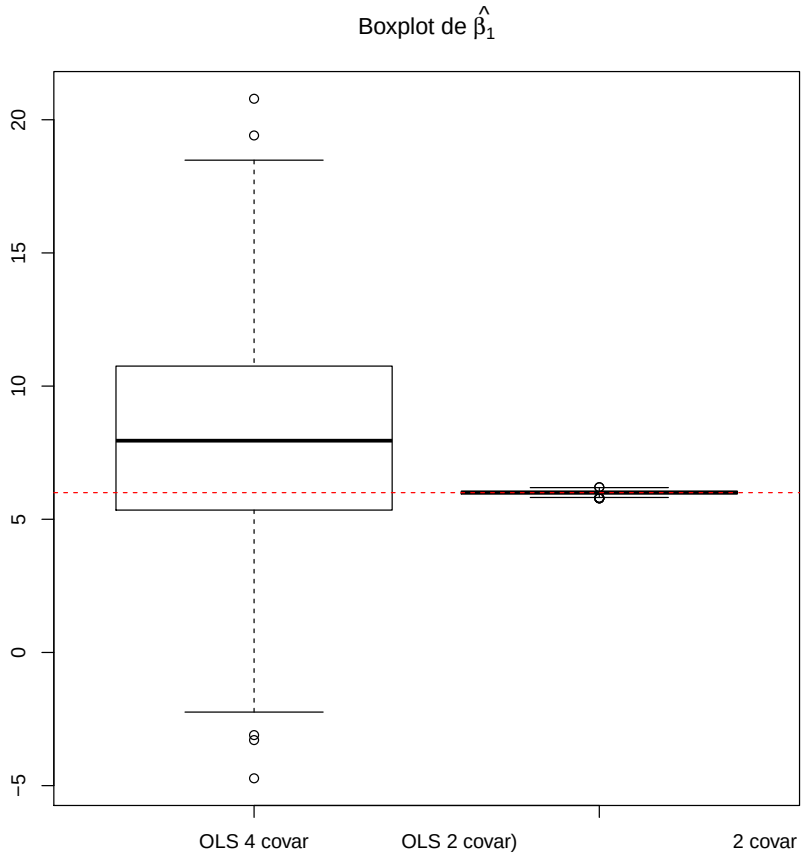
β	Modelo (1)	Modelo (2)
0	1000	1000
1	514	1000
2	1000	1000
3	80	0
4	595	0

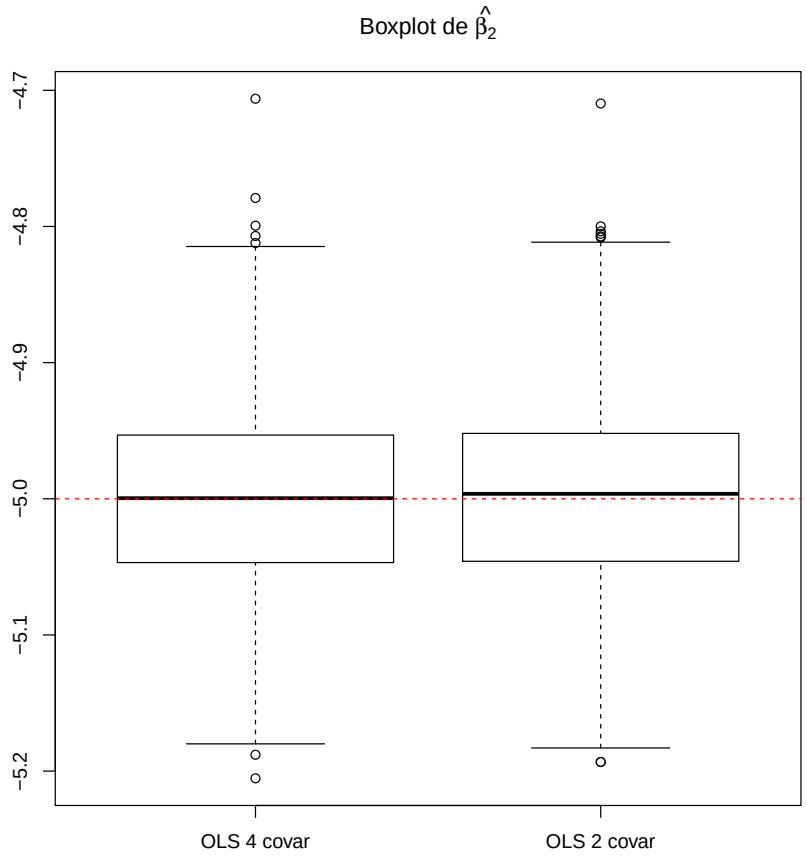
Resultados de la simulación

Desvíos estándares de los estimadores de cada coeficiente obtenidos en ambos ajustes

	sd (4 covar)	sd (2 covar)
$\widehat{\beta}_0$	0.27	0.27
$\widehat{\beta}_1$	4.06	0.07
$\widehat{\beta}_2$	0.07	0.07
$\widehat{\beta}_3$	1.76	.
$\widehat{\beta}_4$	1.78	.

Comparemos los estimadores de los coeficientes





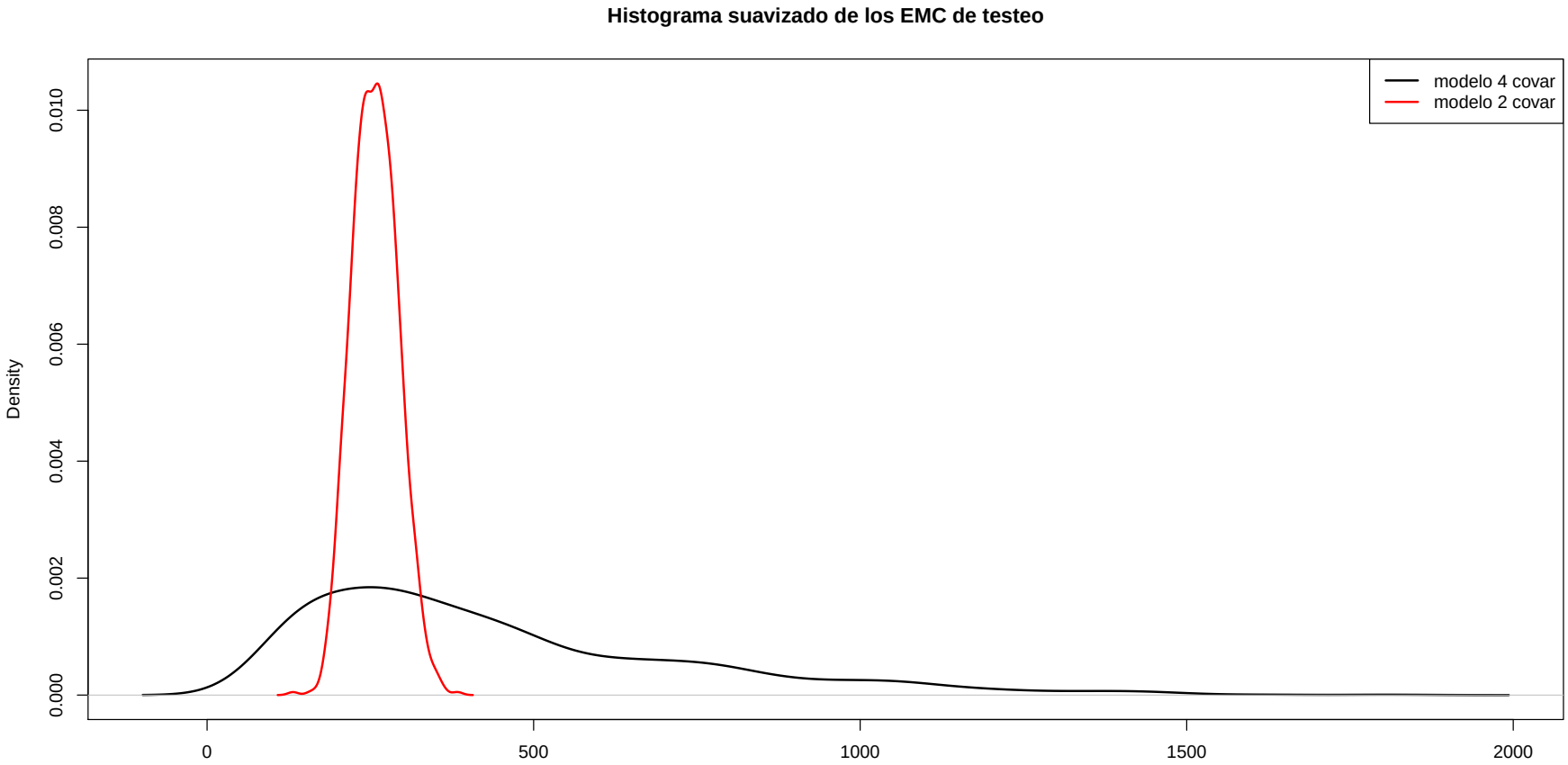
Resultados de la simulación

Otra manera de evaluar la simulación es calculando el error cuadrático medio de predicción en una muestra de testeo independiente de la utilizada en el ajuste.
Para cada repetición, $j = 1, \ldots, B = 1000$

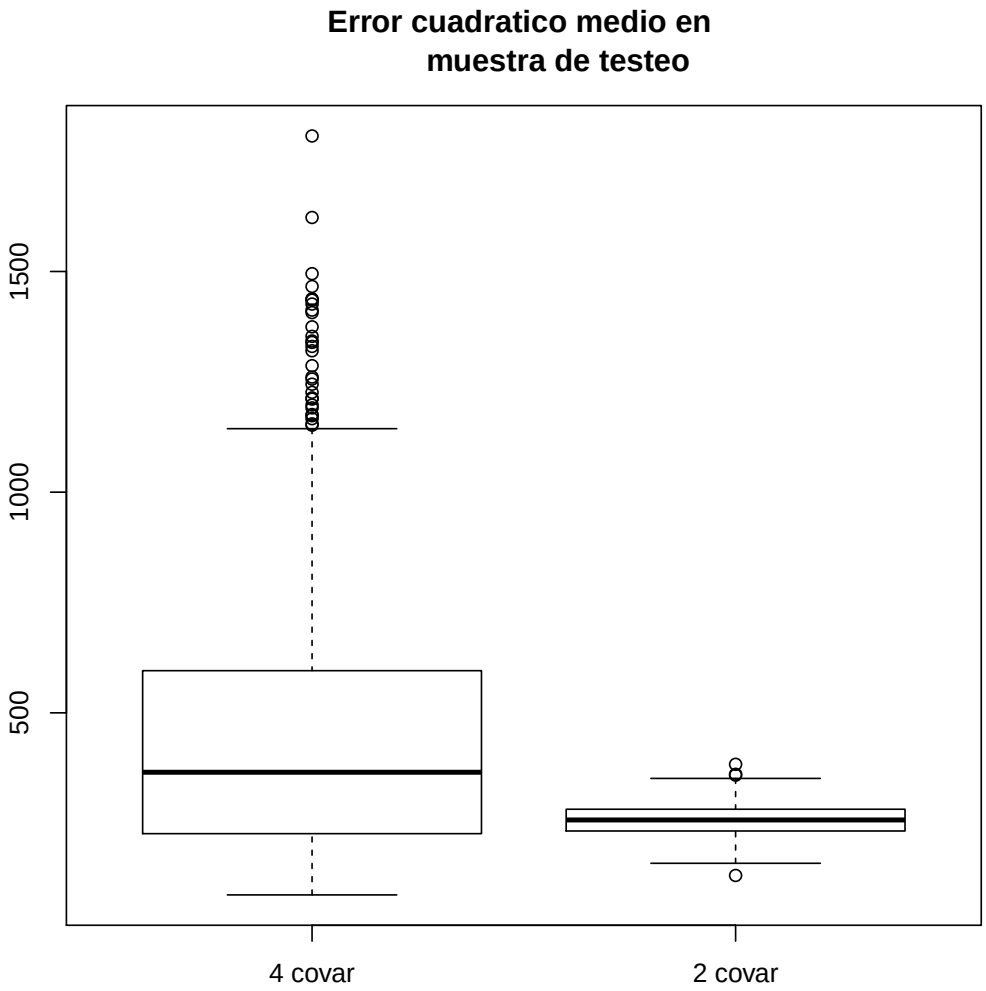
- Generamos 100 datos $\left\{ \left(\mathbf{x}_i^{(j)}, y_i^{(j)} \right) \right\}_{1 \leq i \leq n=100}$ la *muestra de entrenamiento* con la que estimamos a los $\widehat{\boldsymbol{\beta}^{(j)}}$
- Generamos una nueva *muestra de testeo* $\left\{ \left(\mathbf{x}_i^{(j)n}, y_i^{(j)n} \right) \right\}_{1 \leq i \leq n}$ siguiendo el modelo antes descripto, independiente de la muestra de entrenamiento
- Calculamos el $ECMpred^{(j)}$, el error cuadrático medio de predicción para la j-ésima repetición, como

$$ECMpred^{(j)} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i^{(j)n} - \mathbf{x}_i^{(j)n} \widehat{\boldsymbol{\beta}^{(j)}} \right)^2$$

Resultados de la simulación



Resultados de la simulación



Limitar el presupuesto

Una solución posible: imponer una restricción al tamaño en los coeficientes. Puede pensarse que se dispone de un "presupuesto" total limitado (o norma del vector de los coeficientes), y que se quiere hacer el mejor uso de este presupuesto para repartirlo entre los coeficientes involucrados. Mientras más restringido el presupuesto, menos posibilidad de "ajustar demasiado bien a los datos", luego al limitar el presupuesto, limitamos el "sobreajuste"

Estimadores Ridge

Definimos los estimadores Ridge por

$$\begin{aligned} \widehat{\boldsymbol{\beta}}^{\text{ridge}} &= \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2, \\ &\quad \text{sujeto a } \sum_{j=1}^{p} \beta_j^2 \leq t^2. \end{aligned} \tag{3}$$

El término de restricción $\sum_{j=1}^{p} \beta_j^2 = \|(\beta_1, \ldots, \beta_p)\|_2^2$ trata a todos los coeficientes como iguales. Esta penalización (o restricción) es más natural si todas las variables son medidas en la misma escala. Típicamente se usa la regresión ridge sobre datos transformados para que tengan desvío estándar igual a uno y media muestral 0.
Con esto, logramos que los coeficientes de regresión $\beta_1, \ldots, \beta_p$ estén en escalas comparables. También asumimos que $\overline{y} = 0$. Los estimadores ridge no son equivariantes bajo reescalamientos de las covariables.

Estimadores Ridge: multiplicadores de Lagrange

Alternativamente, el problema de regresión ridge puede ser dado por

$$\widehat{\boldsymbol{\beta}}^{\text{ridge}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}. \qquad (4)$$

donde $\lambda \geq 0$ es un parámetro que se determina de forma separada de los $\boldsymbol{\beta}$. Miremos la nueva función objetivo

$$G(\boldsymbol{\beta}) = \left\{ \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

Estimadores Ridge: función objetivo G

$$G(\boldsymbol{\beta}) = \left\{ \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

Encontrar un valor de $\boldsymbol{\beta}$ que minimice a esta función se consigue balanceando dos criterios:

- el primer sumando es la RSS, minimizar RSS equivale a elegir $\boldsymbol{\beta}$ que ajuste bien a los datos,
- el segundo sumando, $\lambda \sum_{j=1}^{p} \beta_j^2$ se denomina término de penalización (*shrinkage penalty*: penalización por contracción), se hace más chico cuanto más chica sean los $\boldsymbol{\beta}$, mejor dicho, cuanto más chica sea la norma 2 del vector $\boldsymbol{\beta}$, excluyendo el término independiente.

Estimadores Ridge: función objetivo G

El λ controla el impacto que tienen estos dos criterios en la selección del $\boldsymbol{\beta}$. Por eso a λ se lo llama *tuning parameter* (parámetro de sintonización). Con

- $\lambda = 0$, recuperamos el estimador de mínimos cuadrados (OLS)
- con $\lambda \to +\infty$ el impacto del término de penalización aumenta, al minimizar a G se encuentran estimadores ridge cercanos a cero.
- diferentes valores de λ dan lugar a distintos valores de estimadores ridge, en realidad la notación correcta es $\widehat{\boldsymbol{\beta}}_{\lambda}^{\text{ridge}}$. Por eso, seleccionar un buen valor para λ es clave
- Discutiremos cómo hacerlo más adelante, donde usamos validación cruzada.

Estimadores Ridge: función objetivo G en forma vectorial

Escribimos la función objetivo de (4) de forma vectorial

$$G(\boldsymbol{\beta}) = \|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|^2$$

Diferenciando a G obtenemos que

$$\begin{aligned} \widehat{\boldsymbol{\beta}}_{\lambda}^{\text{ridge}} &= \left(X^T X + \lambda I\right)^{-1} X^T \mathbf{y} \\ &= \left(X^T X + \lambda I\right)^{-1} X^T X \widehat{\boldsymbol{\beta}} \end{aligned}$$

Luego $\widehat{\boldsymbol{\beta}}_{\lambda}^{\text{ridge}}$ es una versión "contraída" de $\widehat{\boldsymbol{\beta}}$, (es decir, del estimador de mínimos cuadrados ordinarios $\widehat{\boldsymbol{\beta}} = \widehat{\boldsymbol{\beta}}_{OLS}$), cuánto más grande sea λ, más se contrae a $\widehat{\boldsymbol{\beta}}$.

Estimadores Ridge: covarianza

Aun en el caso en el que $X^T X$ es singular, el estimador de ridge se basa en invertir una matriz no singular. Esta fue inicialmente la motivación para la introducción de la regresión ridge en estadística (Hoerl and Kennard, 1970).
Puede probarse que,

$$Cov\left(\widehat{\beta}_{\lambda}^{\text{ridge}}\right) = \sigma^2 \left(X^T X + \lambda I\right)^{-1} X^T X \left(X^T X + \lambda I\right)^{-1}$$

Más aún, bajo el supuesto de errores gaussianos, $\widehat{\beta}_{\lambda}$ tiene distribución normal multivariada con la esperanza y varianza recién calculadas.

Ejemplo: Research and development expenditure

En economía, suelen aparecer variables explicativas muy correlacionadas
El *gasto interno bruto en investigación y desarrollo* se define como el gasto total (corriente y capital) en I + D realizado por todas las empresas residentes en un país, sus institutos de investigación, laboratorios universitarios y gubernamentales, etc. Incluye la I + D financiada desde el extranjero, pero excluye los fondos nacionales para la I + D realizada fuera de la economía nacional. Este indicador se mide como porcentaje del PBI.
Fuentes: `data.worldbank.org` y `data.oecd.org`

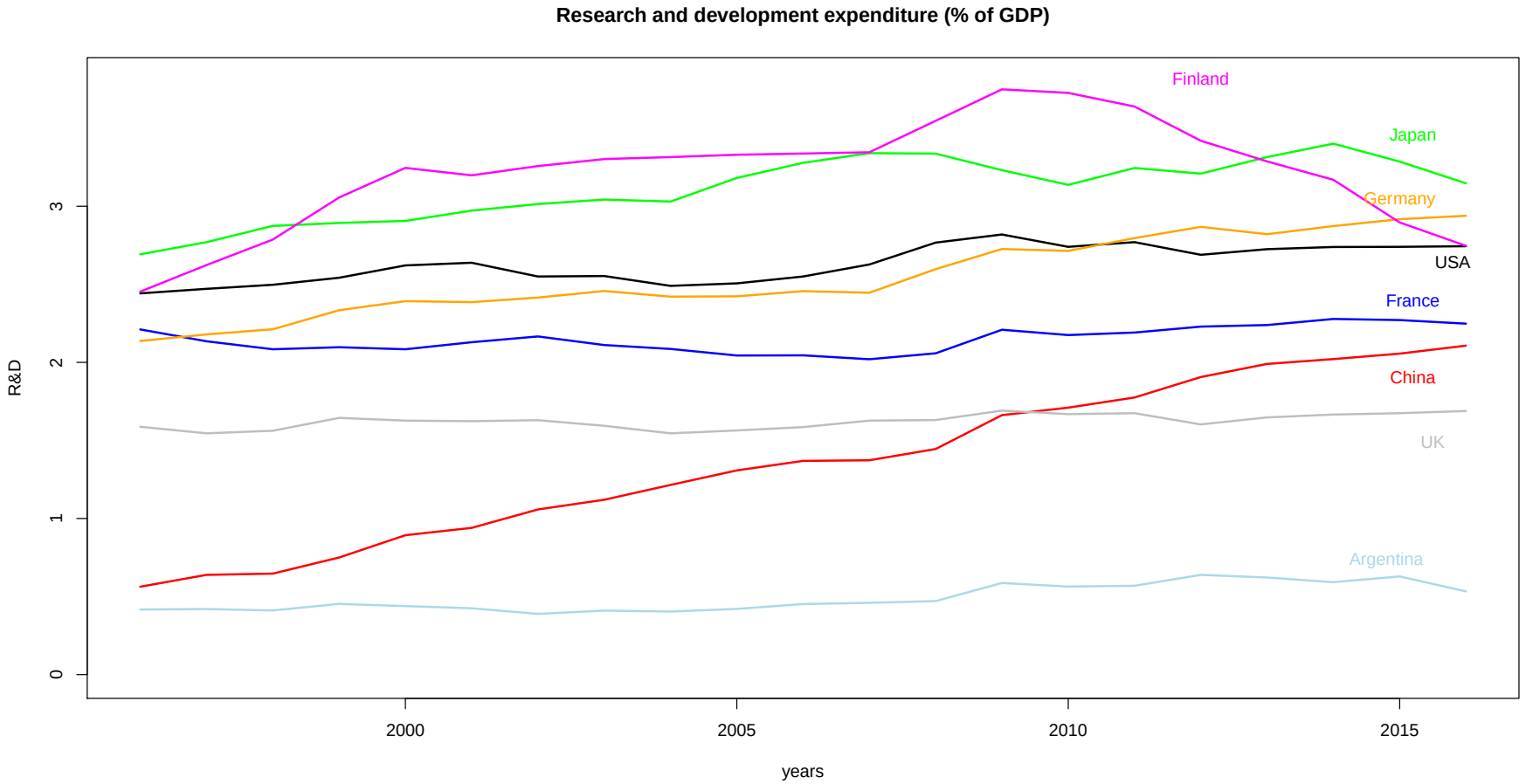
Ejemplo: Research and development expenditure

La base de datos

year	argen	ger	china	japan	france	uk	finl	usa
1996	0.42	2.14	0.56	2.69	2.21	1.59	2.45	2.44
1997	0.42	2.18	0.64	2.77	2.13	1.55	2.62	2.47
1998	0.41	2.21	0.65	2.87	2.08	1.56	2.79	2.50
1999	0.45	2.33	0.75	2.89	2.10	1.64	3.06	2.54
2000	0.44	2.39	0.89	2.91	2.08	1.63	3.25	2.62
⋮		⋮		⋮				
2016	0.53	2.94	2.11	3.15	2.25	1.69	2.75	2.74

Queremos explicar el % de gasto en I + D de Estados Unidos, usando como covariables los % de gasto en I + D de otros países: Argentina, Alemania, China, Japón, Francia, Reino Unido, Finlandia, entre 1996-2016.

Covariables: Research and development expenditures



¿Están correlacionadas las variables?

Matriz de correlaciones

	argen	ger	china	japan	france	uk	finland	usa
argen	1.00	0.89	0.86	0.64	0.73	0.68	0.31	0.81
ger	0.89	1.00	0.98	0.77	0.66	0.74	0.41	0.87
china	0.86	0.98	1.00	0.85	0.58	0.66	0.42	0.82
japan	0.64	0.77	0.85	1.00	0.17	0.50	0.61	0.72
france	0.73	0.66	0.58	0.17	1.00	0.58	-0.16	0.51
uk	0.68	0.74	0.66	0.50	0.58	1.00	0.33	0.86
finland	0.31	0.41	0.42	0.61	-0.16	0.33	1.00	0.55
usa	0.81	0.87	0.82	0.72	0.51	0.86	0.55	1.00

En R

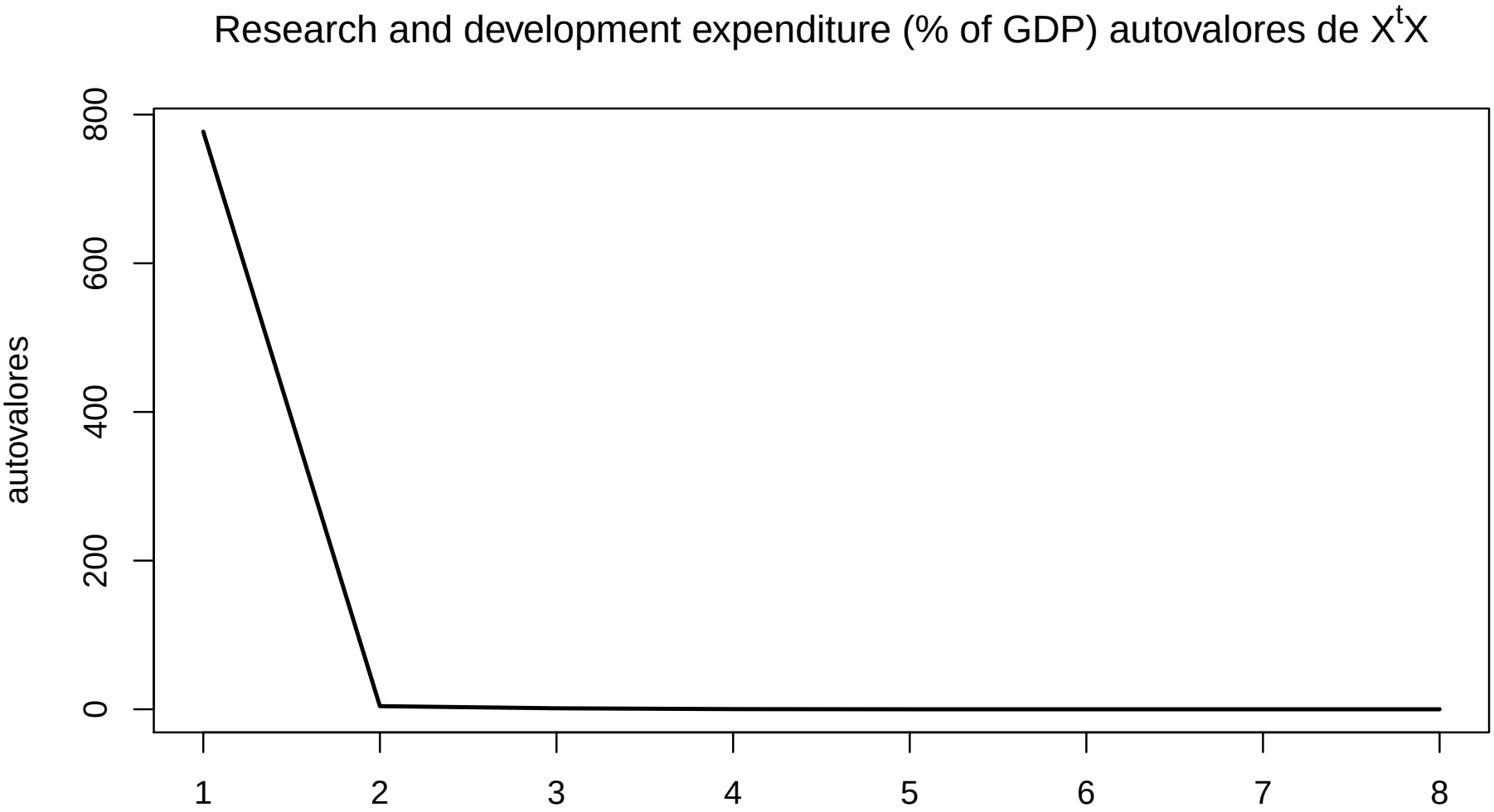
Archivo: `ridge.R`

```
ajustels<-lm(usa~.,data = rdpercentGDP[,-1] )
XX<-model.matrix(ajustels)  #matriz de disenio
t(XX)%*%XX
> autoval<-eigen(t(XX)%*%XX)$values

> #numero de condicion
> autoval[1]/autoval[8]
[1] 367265.9

> autoval[8]
[1] 0.002116002
```

Autovalores ordenados de la matriz de diseño $X^T X \in \mathbb{R}^{8\times 8}$

	1	2	3	4	5	6	7	8
autovalores	777.14	4.22	1.36	0.22	0.03	0.02	0.01	0.00

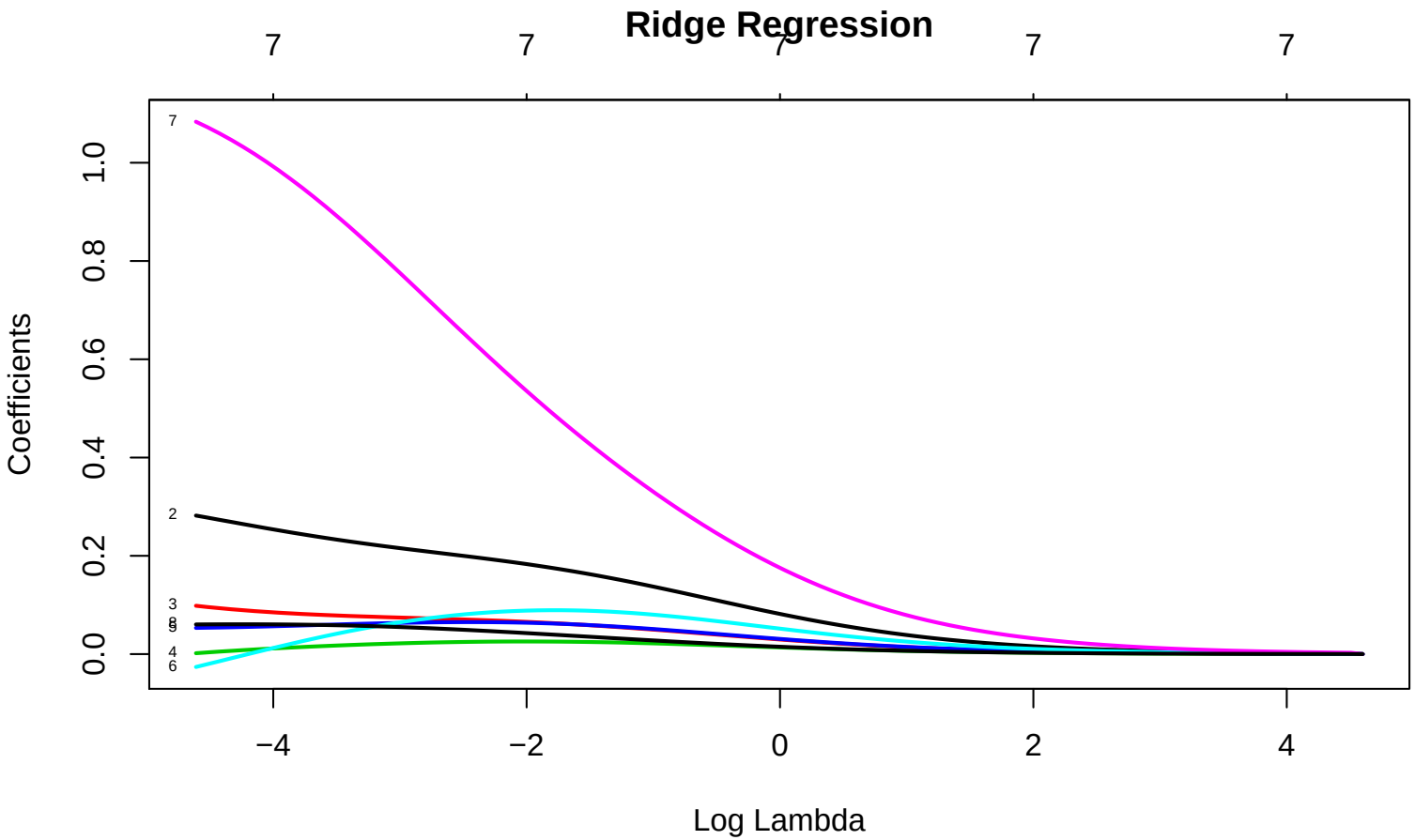
Ejemplo: Autovalores de $X^T X$



library glmnet

```
library(glmnet)

XX<-model.matrix(ajustels)   #matriz de disenio
# poner alpha = 0 para que haga el ajuste ridge

ajuste.ridge = glmnet(XX,usa,alpha=0)
plot(ajuste.ridge,label = T,xvar="lambda",lwd=2)
```

R & D: coeficientes para distintos λ en Ridge




```
> names(rdpercentGDP)
[1] "year" "argen" "ger" "china" "japan" "france" "uk"
[8] "finl" "usa"
```

Research and development expenditures

Ajuste por mínimos cuadrados

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.3106	0.8551	-0.36	0.7223
argen	0.3248	0.2691	1.21	0.2491
germany	0.3988	0.2633	1.51	0.1537
china	-0.1596	0.1378	-1.16	0.2675
japan	0.1251	0.1427	0.88	0.3965
france	-0.1240	0.2737	-0.45	0.6579
uk	1.0576	0.3449	3.07	0.0090
finland	0.0444	0.0401	1.11	0.2883

Coeficientes de ridge, en función de λ

```
> rid.l1<-glmnet(XX, usa,alpha=0,lambda = 1)
> rid.l2<-glmnet(XX, usa,alpha=0,lambda = 2)
> rid.l4<-glmnet(XX, usa,alpha=0,lambda = 4)
> rid.l08<-glmnet(XX, usa,alpha=0,lambda = 0.08453212)
> cbind(coef(rid.l08),coef(rid.l1),coef(rid.l2)
,coef(rid.l4))
                      s0          s0           s0           s0
(Intercept) 0.73511044 1.95537033 2.225203602 2.404638483
argen       0.19890620 0.08170706 0.049699041 0.027850802
ger         0.07032570 0.02971715 0.018078091 0.010130148
china       0.02485736 0.01357721 0.008362876 0.004715379
japan       0.06508696 0.03085011 0.018822180 0.010558955
france      0.08132040 0.05177145 0.032315034 0.018354525
uk          0.64634447 0.17547743 0.102601972 0.056296279
finl        0.04914211 0.01493489 0.008732780 0.004787035
```

¿Qué ganamos con la regresión Ridge?

- Mínimos cuadrados es lo mismo que ajustar ridge con $\lambda = 0$,

```
ajustels<-lm(usa~.,data = rdpercentGDP[,-1])
rid.ols<-glmnet(XX, usa,alpha=0,lambda = 0)
```

- Como se esperaba, ninguno de los coeficientes es cero: ¡la regresión ridge no selecciona variables!
- Los estimadores ridge son sesgados.
- La ventaja de la regresión ridge sobre OLS radica en el trade–off sesgo-varianza. A medida que aumenta λ, la flexibilidad del ajuste ridge disminuye, dando lugar a una disminución de la varianza, pero un mayor sesgo.

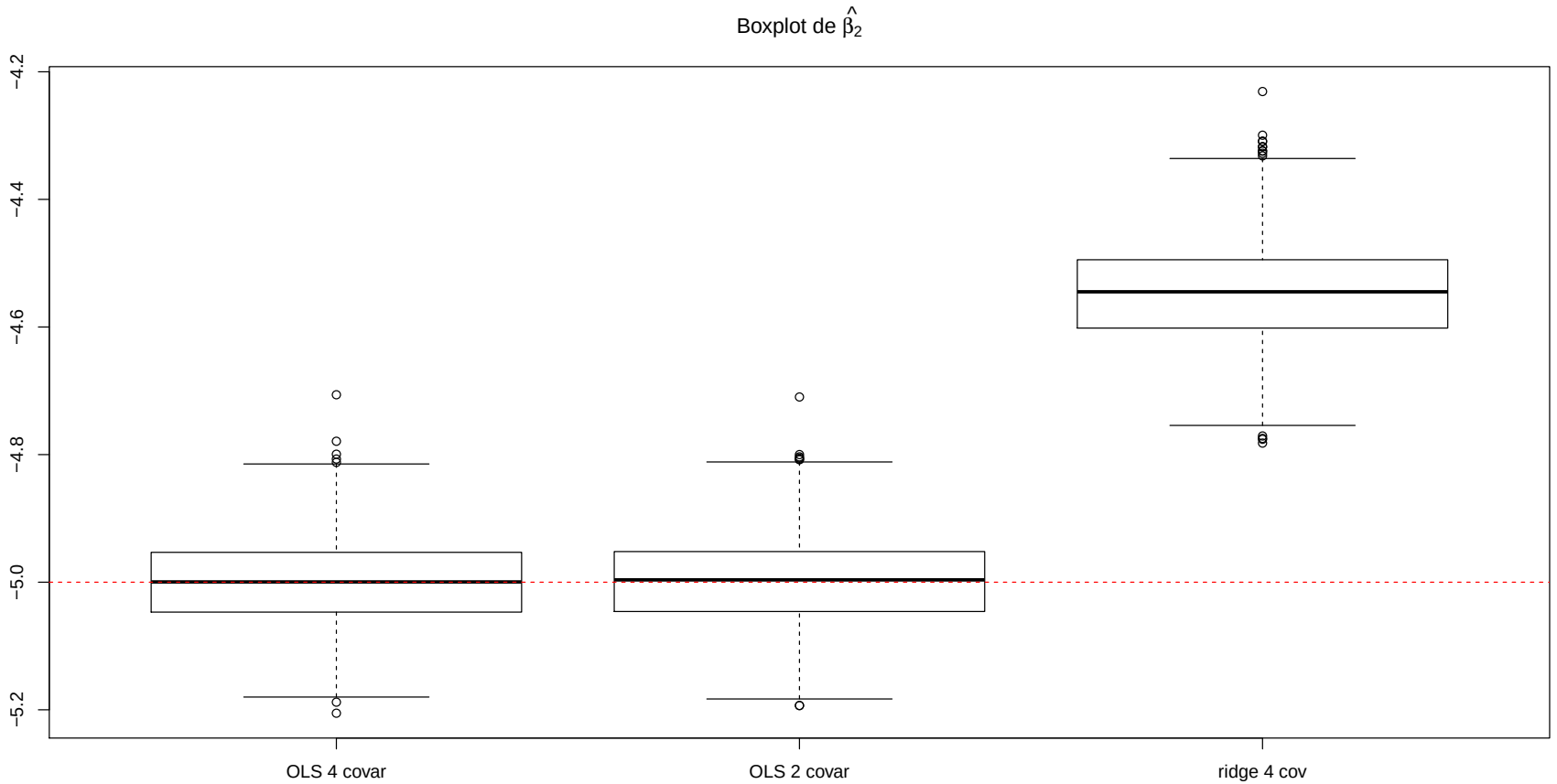
Resultados de la simulación: comparación con Ridge

Volvemos al ejemplo de datos correlacionados. Comparemos los ajustes de mínimos cuadrados con cuatro covariables, dos covariables y el ajuste ridge con las 4 covariables. Desvíos estándares de los estimadores de cada coeficiente obtenidos en los 3 ajustes

	sd (4 covar)	sd (2 covar)	sd ridge
$\widehat{\beta}_0$	0.27	0.27	0.28
$\widehat{\beta}_1$	4.06	0.07	0.03
$\widehat{\beta}_2$	0.07	0.07	0.08
$\widehat{\beta}_3$	1.76	.	0.02
$\widehat{\beta}_4$	1.78	.	0.04

Comparemos los estimadores de los coeficientes, β_2

En rojo, el valor verdadero.



Ridge da sesgado, varianzas comparables

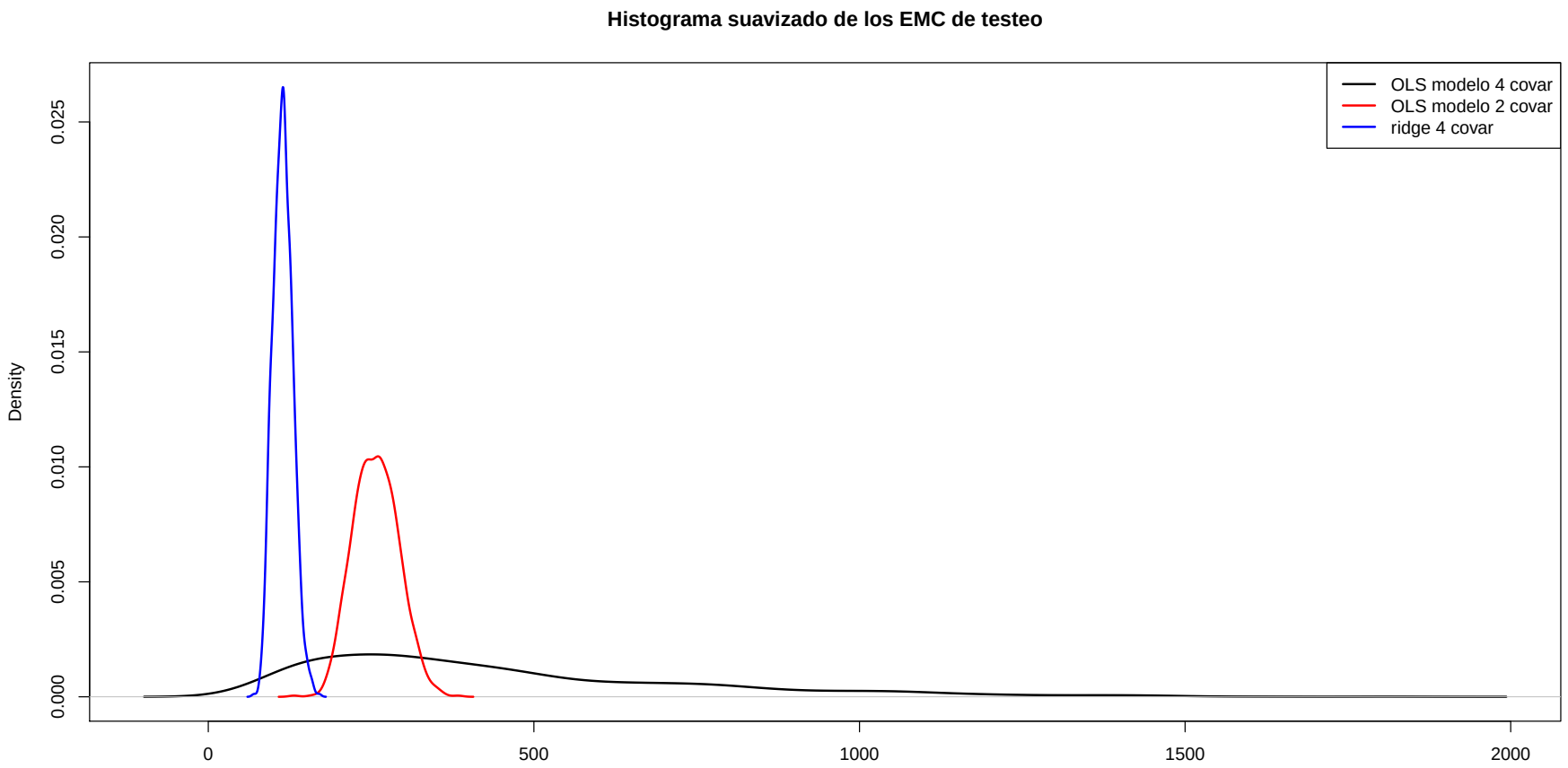
Resultados de la simulación

Otra manera de evaluar la simulación es calculando el error cuadrático medio de predicción en una muestra de testeo independiente de la utilizada en el ajuste.
Para cada repetición, $j = 1, \ldots, B = 1000$

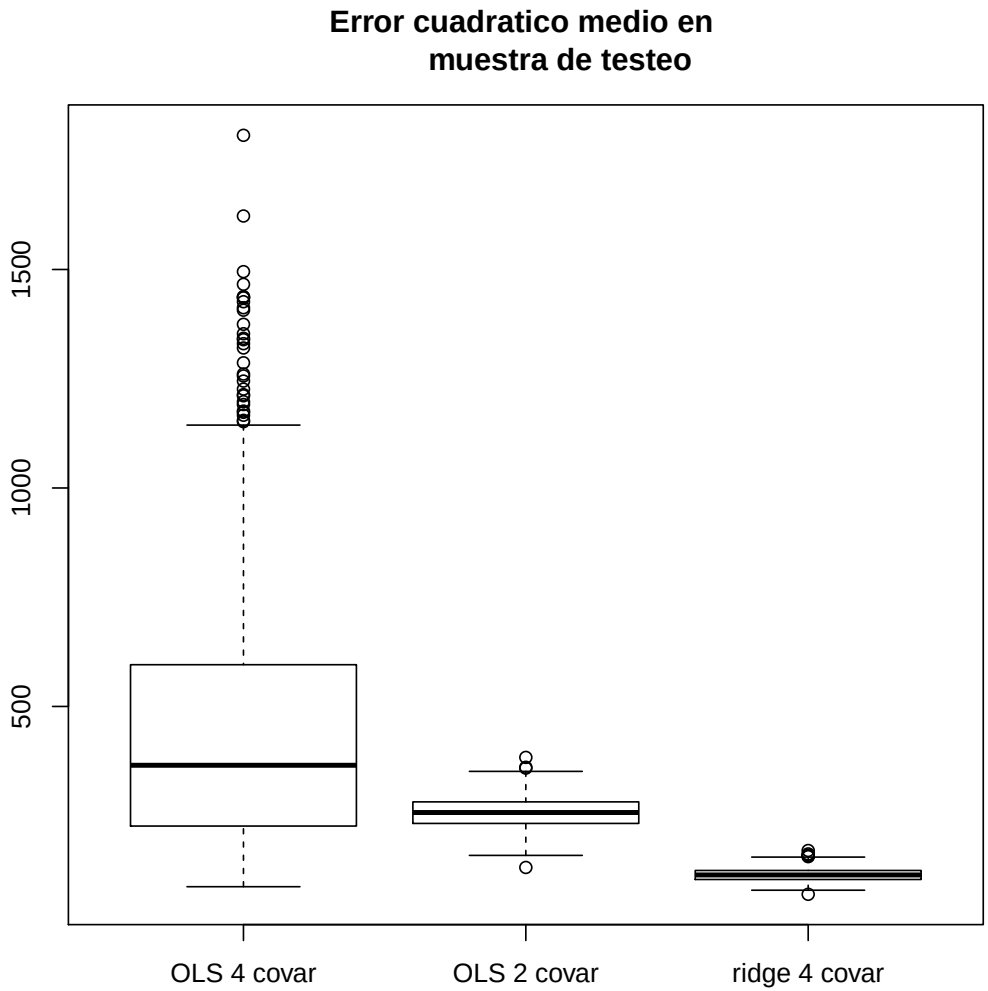
- Generamos 100 datos $\left\{ \left(\mathbf{X}_i^{(j)}, Y_i^{(j)} \right) \right\}_{1 \leq i \leq n=100}$ la *muestra de entrenamiento* con la que estimamos a los $\widehat{\boldsymbol{\beta}^{(j)}}$
- Generamos una nueva *muestra de testeo* $\left\{ \left(\mathbf{X}_i^{(j)n}, Y_i^{(j)n} \right) \right\}_{1 \leq i \leq n}$ siguiendo el modelo antes descripto, independiente de la muestra de entrenamiento
- Calculamos el $ECMpred^{(j)}$, el error cuadrático medio de predicción para la j-ésima repetición, como

$$ECMpred^{(j)} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i^{(j)n} - \mathbf{X}_i^{(j)n} \widehat{\boldsymbol{\beta}^{(j)}} \right)^2$$

Resultados de la simulación, EMCpred con muestra de testeo



Resultados de la simulación, EMCpred con muestra de testeo



LASSO

La regresión Ridge no selecciona variables, porque no tiene ningún incentivo para estimar como exactamente cero a ningún coeficiente β. ¿Cómo conseguimos esto? El LASSO resuelve un criterio similar a Ridge.

$$\begin{aligned} \widehat{\boldsymbol{\beta}}^{\text{lasso}} &= \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2, \\ &\quad \text{sujeto a } \sum_{j=1}^{p} |\beta_j| \leq t. \end{aligned} \tag{5}$$

El término de penalización se puede escribir en forma sintética $\sum_{j=1}^{p} |\beta_j| = \|(\beta_1, \ldots, \beta_p)\|_1$, donde $\|\cdot\|_1$ denota la norma 1 o norma l_1.

Comparación gráfica de "bolas"

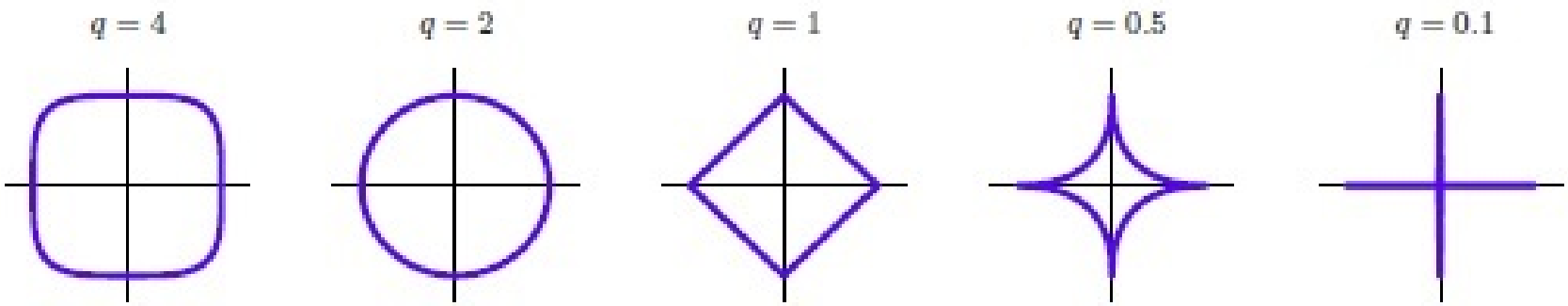


Figure 2.6 *Constraint regions* $\sum_{j=1}^{p} |\beta_j|^q \leq 1$ *for different values of* q. *For* $q < 1$, *the constraint region is nonconvex.*

Fuente: Trevor Hastie, Robert Tibshirani, Martin Wainwright. SLS Statistical Learning with Sparsity The Lasso and Generalizations.

La forma de la región de la restricción determina el punto donde se alcanza el mínimo.

¡Lasso sí selecciona variables! Argumento gráfico

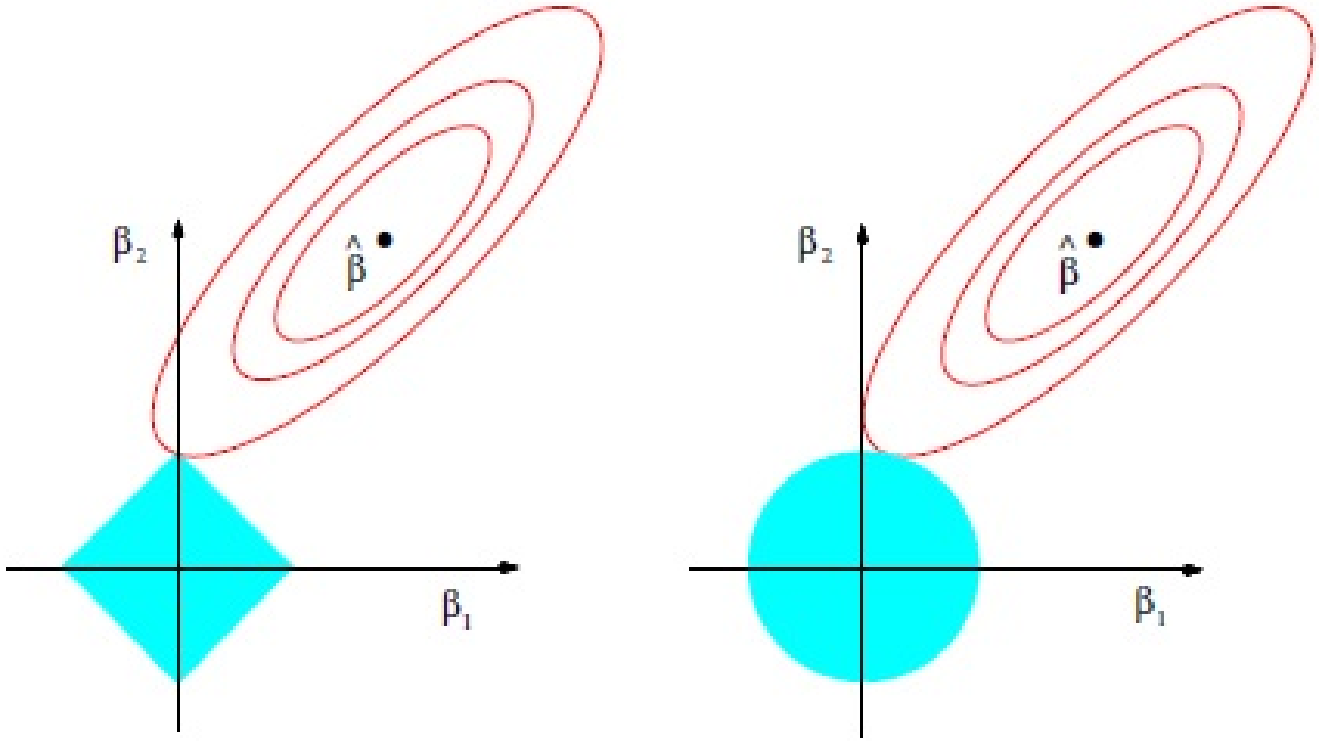


Figure 2.2 *Estimation picture for the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions* $|\beta_1|+|\beta_2| \leq t$ *and* $\beta_1^2+\beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the residual-sum-of-squares function. The point* $\widehat{\beta}$ *depicts the usual (unconstrained) least-squares estimate.*

Fuente: ESL, Hastie, Tibshirani, Friedman

¡Lasso sí selecciona variables!

La función a optimizar (tanto en lasso o ridge) tiene curvas de nivel que son elipses centradas en el estimador OLS del parámetro $\boldsymbol{\beta}$. Las restricciones de ambos criterios son ($p = 2$):

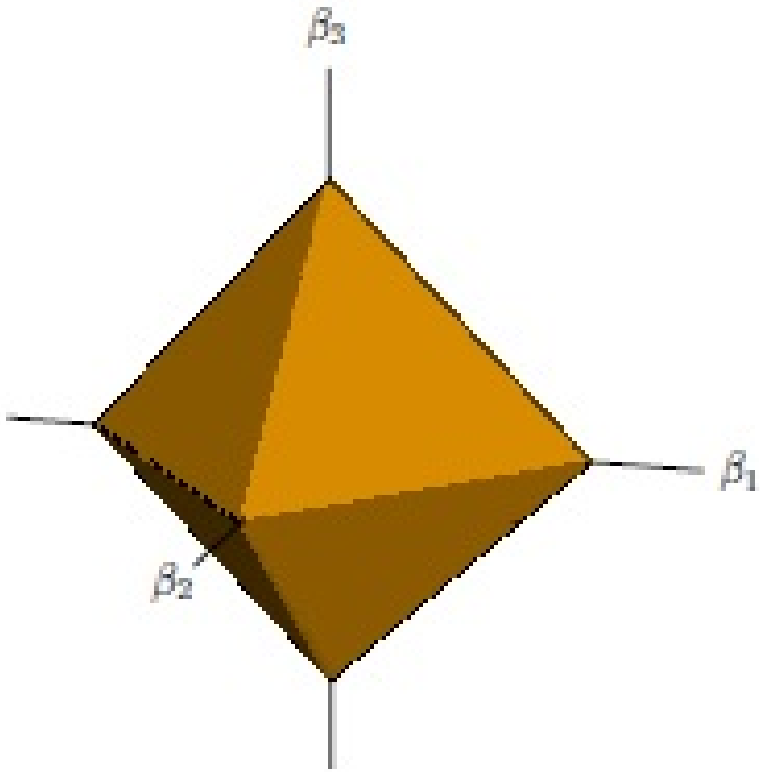
- (ridge) $\beta_1^2 + \beta_2^2 \leq t^2$ (disco)
- (lasso) $|\beta_1| + |\beta_2| \leq t$ (rombo)

Ambos métodos buscan el punto de la restricción que corta a la elipse más cercana al estimador OLS posible. Pero... el rombo ¡tiene esquinas!, por lo que, si la solución ocurre en una esquina, alguno de los coeficientes estimados será igual a cero.

El libro de Bertsekas, D., *Nonlinear Programming* (1995), capítulo 5 es una buena referencia donde se prueba que bajo ciertas hipótesis, el óptimo de una función convexa sobre un conjunto convexo (como (5) se alcanza en la frontera del conjunto.

¡Lasso sí selecciona variables!

- ¿Cómo son las restricciones de lasso para $p > 2$?



Más oportunidades para que los parámetros estimados valgan cero.

Regresión Lasso

Una forma alternativa de escribir el problema de regresión lasso es

$$\widehat{\boldsymbol{\beta}}^{\text{lasso}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\} \tag{6}$$

para algún $\lambda > 0$, o matricialmente,

$$\widehat{\boldsymbol{\beta}}^{\text{lasso}} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - X\boldsymbol{\beta}\|_2^2 + \lambda \|(\beta_1, \ldots, \beta_p)\|_1 \right\}. \tag{7}$$

En algunas versiones, el primer sumando viene acompañado de un factor $1/n$ o $1/2n$, eso sólo reparametriza el valor de λ. La rutina `glmnet` de R usa el factor $1/2n$ en la función G que optimiza, tanto para ridge como para lasso.

Regresión Lasso

La función objetivo

$$G(\boldsymbol{\beta}, \lambda) = \|\mathbf{y} - X\boldsymbol{\beta}\|_2^2 + \lambda \|(\beta_1, \ldots, \beta_p)\|_1 \tag{8}$$

- No es derivable
- Es convexa, encontrar los estimadores lasso es un problema de optimización convexa
- Por lo que puede resolverse de forma numéricamente eficiente cuando n y p son grandes
- Asumimos que los valores de y_i han sido centrados, es decir, $\bar{y} = 0$, y en tal caso se puede omitir β_0 de la optimización
- Una vez obtenido el vector $\left(\widehat{\beta}_1, \ldots, \widehat{\beta}_p\right)$ para los datos centrados, el estimador para los datos originales se completa tomando

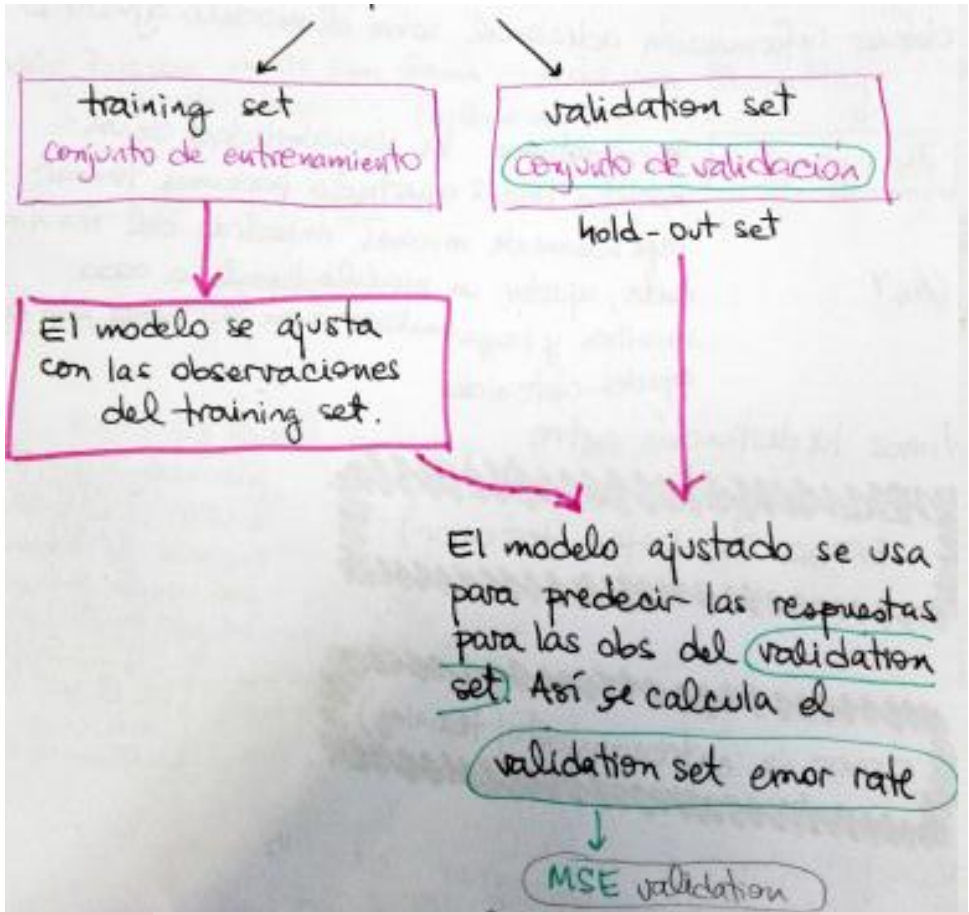
$$\widehat{\beta}_0 = \bar{y} - \sum_{j=1}^{p} \bar{x}_j \widehat{\beta}_j$$

Regresión Lasso

La optimización se consigue a través de teoría de análisis convexo. Puede encontrarse a través de subgradientes una condición necesaria y suficiente para resolver (7). En R, se consigue con el paquete `glmnet`, el comando `glmnet` con la opción `alpha=1`.

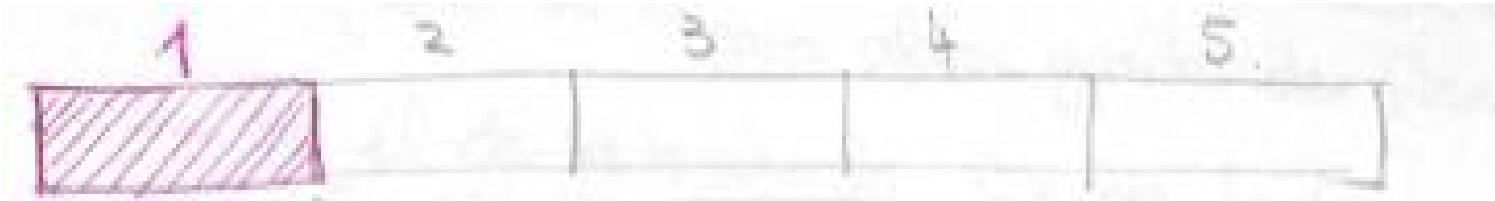
Validación cruzada

La validación cruzada (*cross-validation*) es una técnica estadística que se usa tanto para evaluar el comportamiento de un estimador como para seleccionar un modelo entre varios posibles. Consiste en la división aleatoria de la muestra en dos partes: una parte de *entrenamiento* y otra de *validación*: la primera se utiliza para ajustar el modelo, la segunda para validarlo.



Validación cruzada para elegir λ

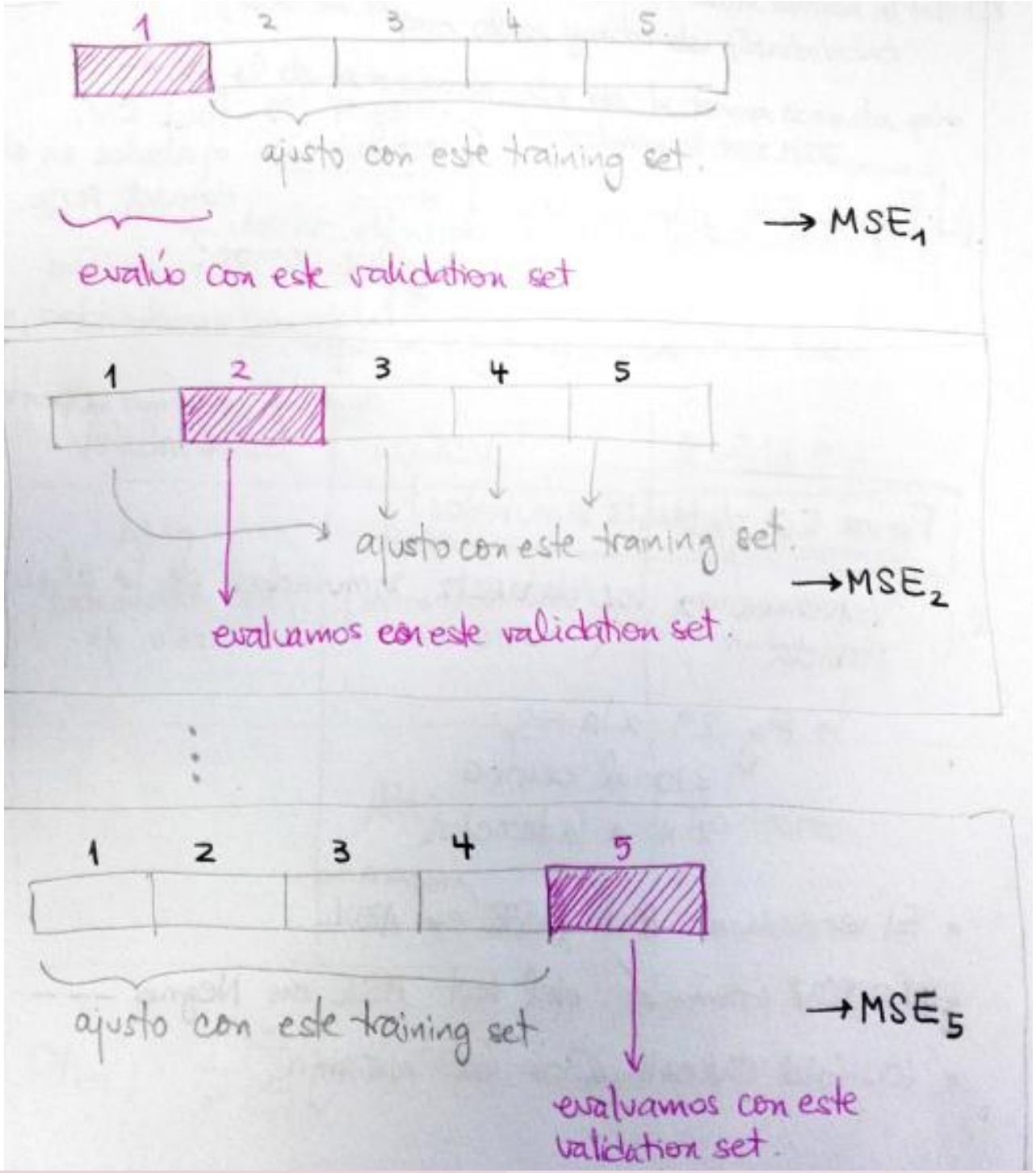
1. Dividimos a la muestra en K partes (o *folds*) de igual tamaño, elegidas de forma aleatoria, $\bigcup_{k=1}^{K} B_k = \{1, 2, \ldots, n\}$. Típicamente $K = 5$ ó 10. A veces también n. Ejemplo, con $K = 5$



   Fijamos una grilla de valores para λ.

2. Para cada *fold* o parte, $k = 1, \ldots, K$,
   1. **(Entrenamiento)** Para cada λ en la grilla, calculamos los estimadores lasso a partir de las observaciones de la **muestra de entrenamiento**, es decir, las que **no** pertenecen a B_k.
   2. **(Validación o testeo)** Para cada observación $(\mathbf{x}_j, y_j)$ tal que $j \in B_k$, es decir para cada observación del grupo de testeo, predecimos el valor de y_j con los estimadores lasso calculados. Y luego calculamos el error cuadrático medio de predicción para cada λ en la grilla, $ECM_k(\lambda)$.

Por ejemplo, si $K = 5$



Validación cruzada para elegir λ

1. Dividimos a la muestra aleatoriamente en K partes $\bigcup_{k=1}^{K} B_k = \{1, 2, \ldots, n\}$ con $B_k \cap B_l = \emptyset$.
2. Para cada parte, $k = 1, \ldots, K$, y para cada λ en la grilla,
   1. **(Entrenamiento)** Calculamos el estimador lasso basado en la muestra de entrenamiento
   2. **(Validación o testeo)** Predecimos a y_j del grupo de testeo, con los estimadores lasso calculados. Calculamos el $ECM_k(\lambda)$.
3. Para cada λ en la grilla promediamos estas K estimaciones del error de predicción, lo que produce una curva de error de validación cruzada

$$CV(\lambda) = \frac{1}{K} \sum_{i=k}^{K} ECM_k(\lambda)$$

cuyo desvío estándar muestral resulta ser

$$SD(\lambda) = sd\left(ECM_1(\lambda), \ldots, ECM_K(\lambda)\right). \tag{9}$$

Validación cruzada para elegir λ: mínimo error de CV

Para cada λ en la grilla, promediamos estas K estimaciones del error de predicción, lo que produce una curva de error de validación cruzada $CV(\lambda)$ y tenemos el $SD(\lambda)$ que suele denominarse error estándar.
El criterio que suele utilizarse para elegir el valor de λ por validación cruzada sería tomar el

$$\widehat{\lambda}_{cross} = \arg\min_{\lambda} CV(\lambda). \tag{10}$$

Finalmente el modelo se vuelve a ajustar usando todas las observaciones y el valor seleccionado para el parámetro de tuneo $\lambda = \widehat{\lambda}_{cross}$.

Validación cruzada para elegir λ: regla de un desvío estándar

A veces el criterio que se usa para elegir el parámetro de tuneo es la regla de "un desvío estándar": elegir el modelo más parsimonioso cuya función de performance diste menos de un error estándar de la mínima. Esta regla, sugerida por ESL (Sección 7.10) toma en cuenta que la curva de CV está estimada con error, y se basa en el hecho de que cuánto menor sea la cantidad de parámetros a estimar con los mismos datos, mejor será la calidad de la estimación obtenida (i.e. será menor la varianza de los mismos). El criterio que utilizaremos para elegir el λ siguiendo *la regla de un desvío estándar* será

$$\widehat{\lambda}_{cross1de} = \text{máx}\left\{\lambda : CV\left(\lambda\right) < CV\left(\widehat{\lambda}_{cross}\right) + SE\left(\widehat{\lambda}_{cross}\right)\right\}. \qquad (11)$$

donde $\widehat{\lambda}_{cross}$ lo definimos en (10).

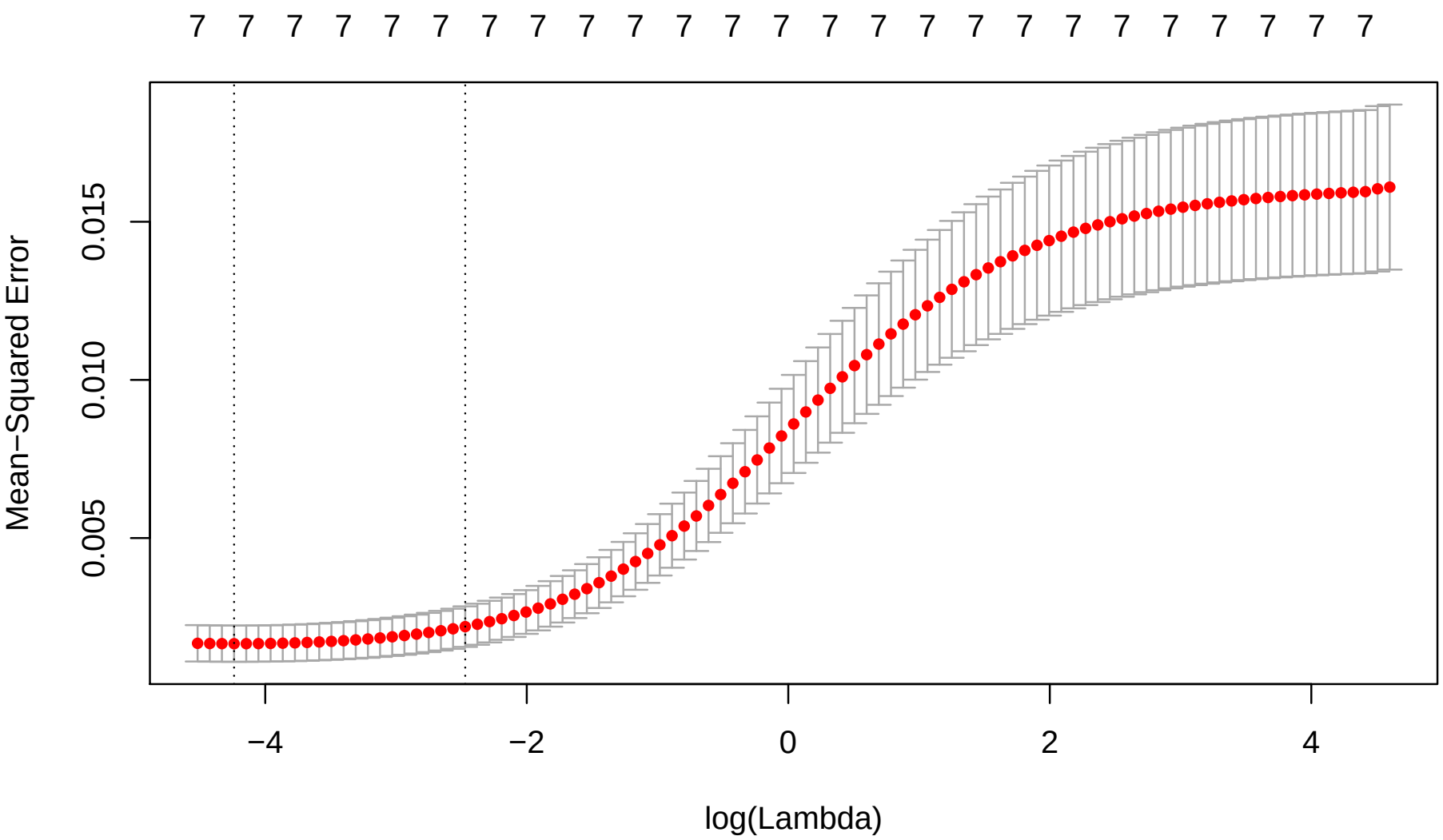
Seleccionamos λ para ridge en R

Seleccionamos el valor de λ para ridge por cross validation con $K = 5$. Obtenemos el mínimo y el mínimo con el criterio de un desvío estándar.

```
> rid.crossval<-cv.glmnet(XX, usa,nfolds = 5, alpha=0)
> names(rid.crossval)
[1] "lambda"   "cvm"  "cvsd"   "cvup"   "cvlo"
[6] "nzero"   "name"  "glmnet.fit" "lambda.min" "lambda.1se"
> rid.crossval$lambda.min
[1] 0.01443261
> rid.crossval$lambda.1se
[1] 0.08453212
plot(rid.crossval)
```

`rid.crossval` es un objeto con muchas cosas, vemos los valores de λ mínimo. Y el gráfico.

Ejemplo R & D: Seleccionamos λ para ridge



Ejemplo R & D: Seleccionamos λ para ridge

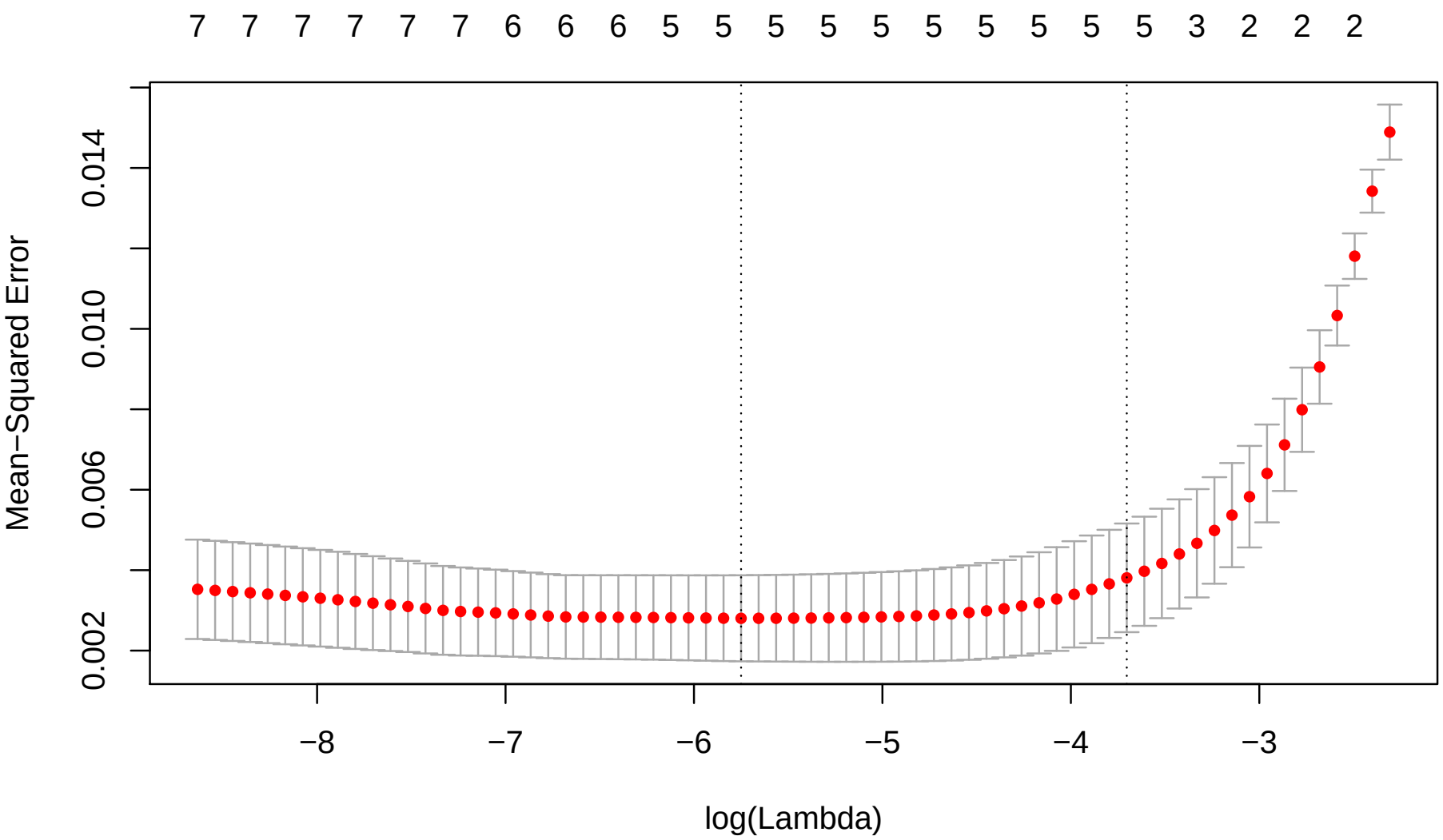
El gráfico anterior muestra los valores de $CV(\lambda)$ (en rojo) en función del logaritmo de λ, para los valores de λ de una grilla. Las líneas verticales que acompañan cada punto rojo son las indicadoras de $+/-$ un desvío estándar, estimado como en (9). Las dos líneas verticales punteadas marcan el mínimo y el mínimo con el criterio de un desvío estándar definidos por (10) y (11), respectivamente. Arriba, en un renglón, aparece la cantidad de variables con coeficientes no nulos en el modelo ajustado con el λ considerado.

Ejemplo R & D: Seleccionamos λ para lasso en R

Seleccionamos el valor de λ para lasso por cross validation con $K = 5$.
Obtenemos el mínimo y el mínimo con el criterio de un desvío estándar.
Con `alpha = 1` tenemos el ajuste lasso.

```
> las.crossval<-cv.glmnet(XX, usa,nfolds = 5, alpha=1)
> las.crossval$lambda.min
[1] 0.00318258
> las.crossval$lambda.1se
[1] 0.02464156
> log(las.crossval$lambda.min)
[1] -5.750063
> log(las.crossval$lambda.1se)
[1] -3.703321
```

Ejemplo R & D: Seleccionamos λ para lasso

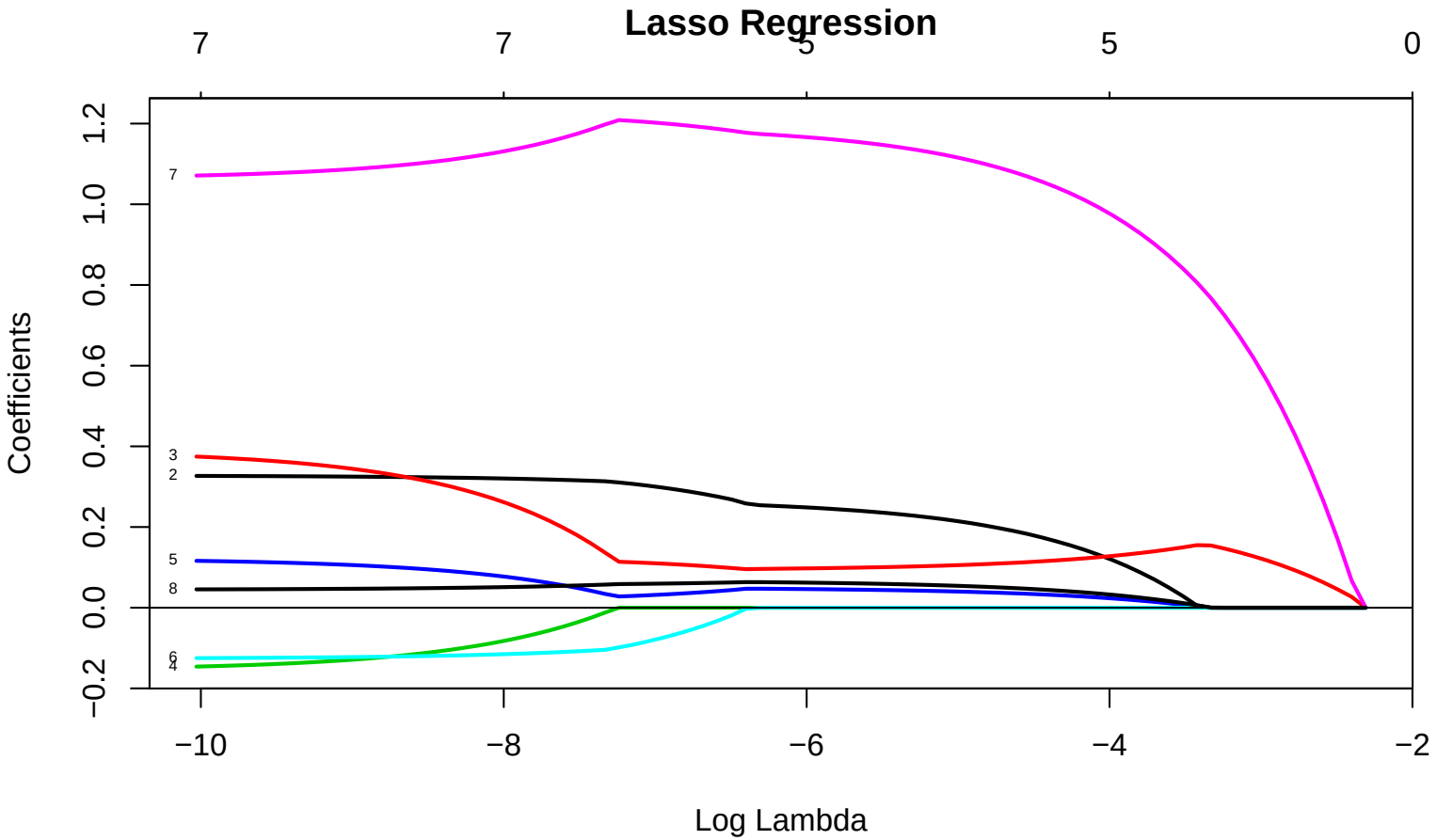


Ejemplo R & D: Ajuste para lasso en R

```
> las.1se<-glmnet(XX, usa,alpha=1,
                        lambda = las.crossval$lambda.1se)
> coef(las.1se)
9 x 1 sparse Matrix of class "dgCMatrix"
                      s0
(Intercept) 0.66408884
(Intercept) .
argen       0.07269915
ger         0.13872767
china       .
japan       0.01466151
france      .
uk          0.90169279
finl        0.02097493
```

Lasso sí selecciona variables.

R & D: coeficientes para distintos λ en Ridge




```
> names(rdpercentGDP)
[1] "year" "argen" "ger" "china" "japan" "france" "uk"
[8] "finl" "usa"
```

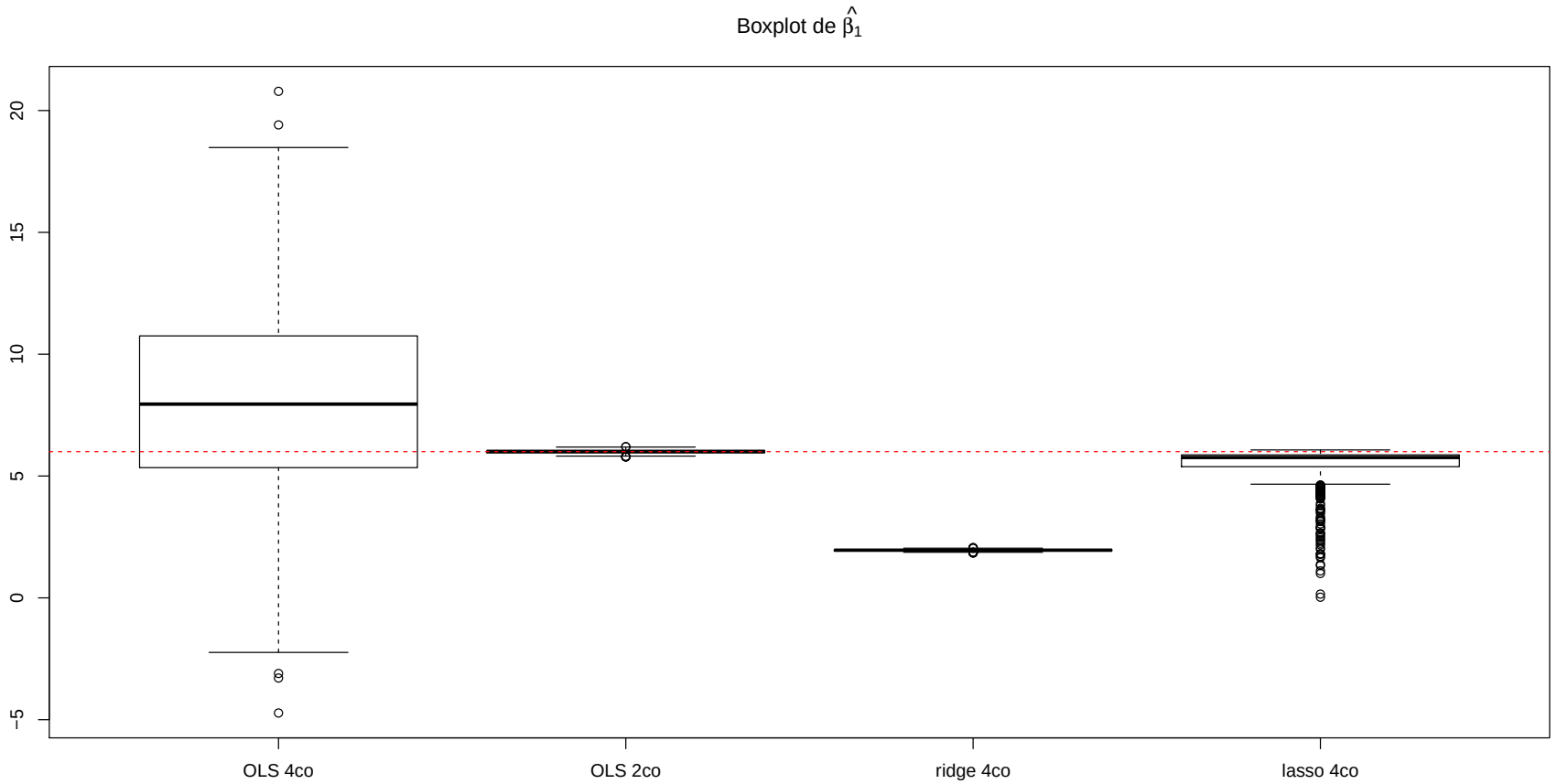
Resultados de la simulación: comparación con Ridge y con lasso

Volvemos al ejemplo de datos correlacionados. Comparemos los ajustes de mínimos cuadrados con cuatro covariables, dos covariables, el ajuste ridge con las 4 covariables y el de lasso, desvíos estándares de los estimadores de cada coeficiente obtenidos.

	sd (4 covar)	sd (2 covar)	sd ridge	sd lasso
$\widehat{\beta}_0$	0.27	0.27	0.28	0.28
$\widehat{\beta}_1$	4.06	0.07	0.03	0.74
$\widehat{\beta}_2$	0.07	0.07	0.08	0.07
$\widehat{\beta}_3$	1.76	.	0.02	0.36
$\widehat{\beta}_4$	1.78	.	0.04	0.02

Comparemos los estimadores de los coeficientes, β_1

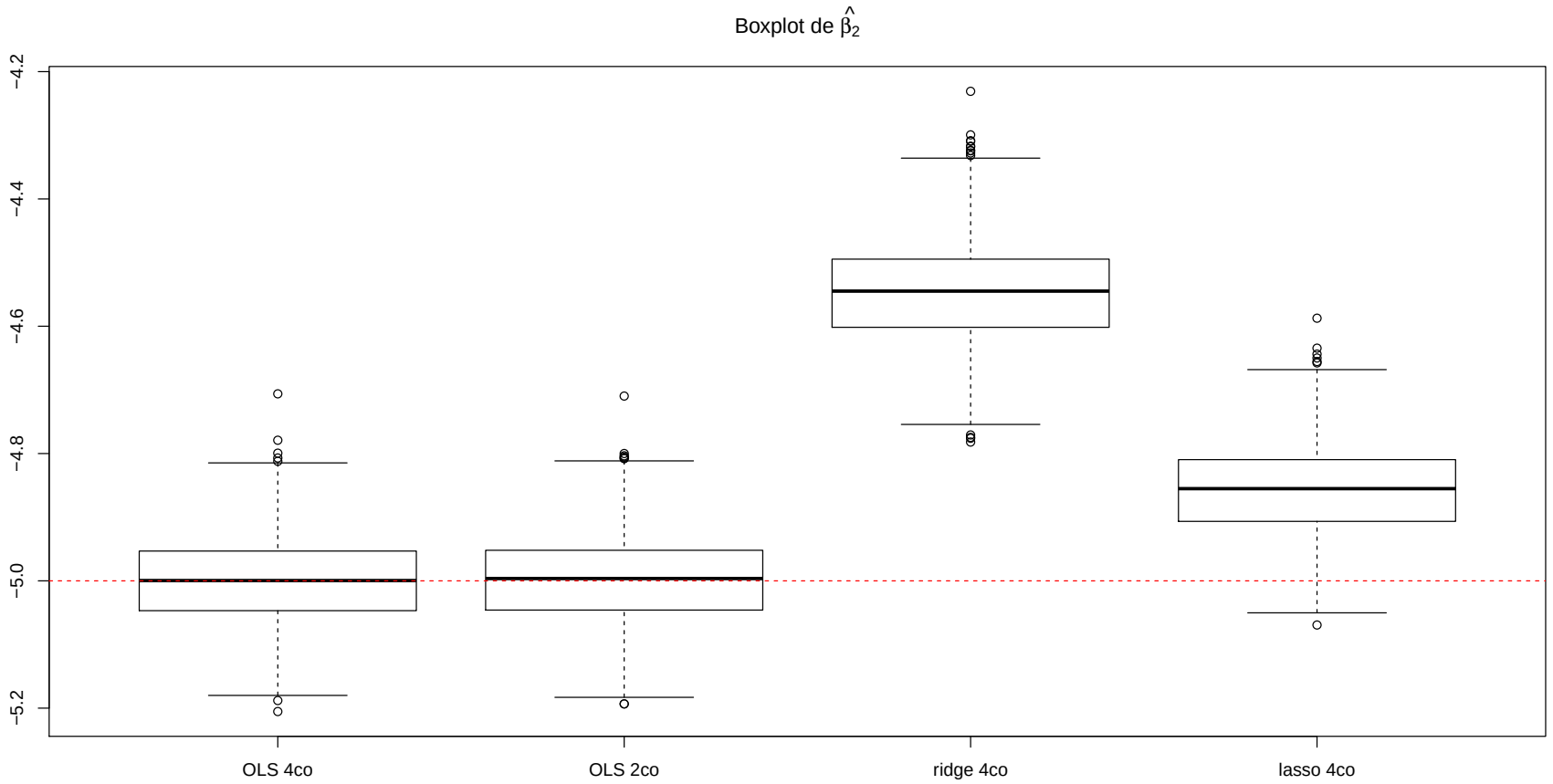
En rojo, el valor verdadero.



Ridge da sesgado (mas chico), lasso da bien, varianzas comparables a OLS 2 covariables.

Comparemos los estimadores de los coeficientes, β_2

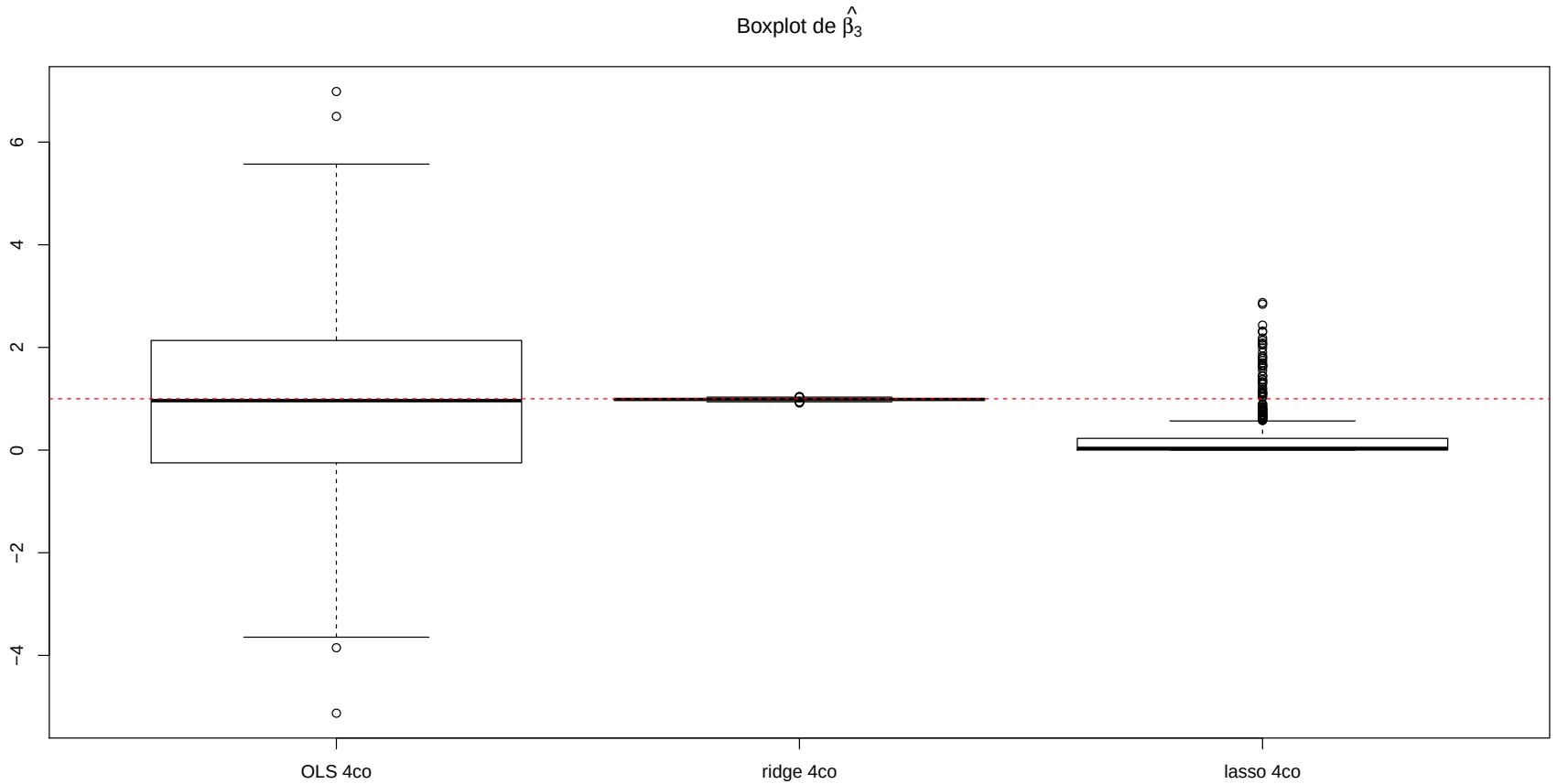
En rojo, el valor verdadero.



Ridge y lasso dan sesgados, varianzas comparables

Comparemos los estimadores de los coeficientes, β_3

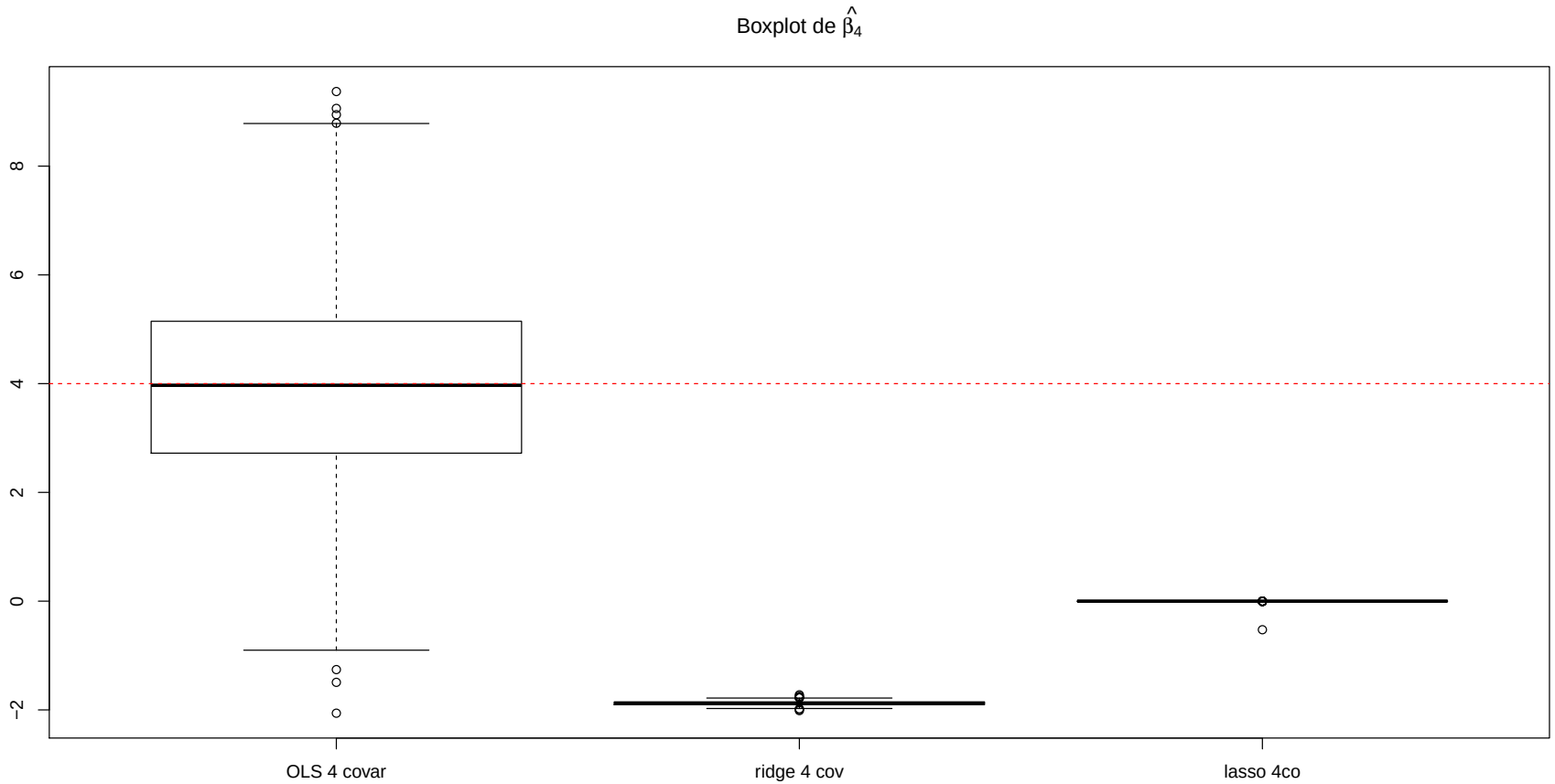
En rojo, el valor verdadero.



Lasso lo estima por cero.

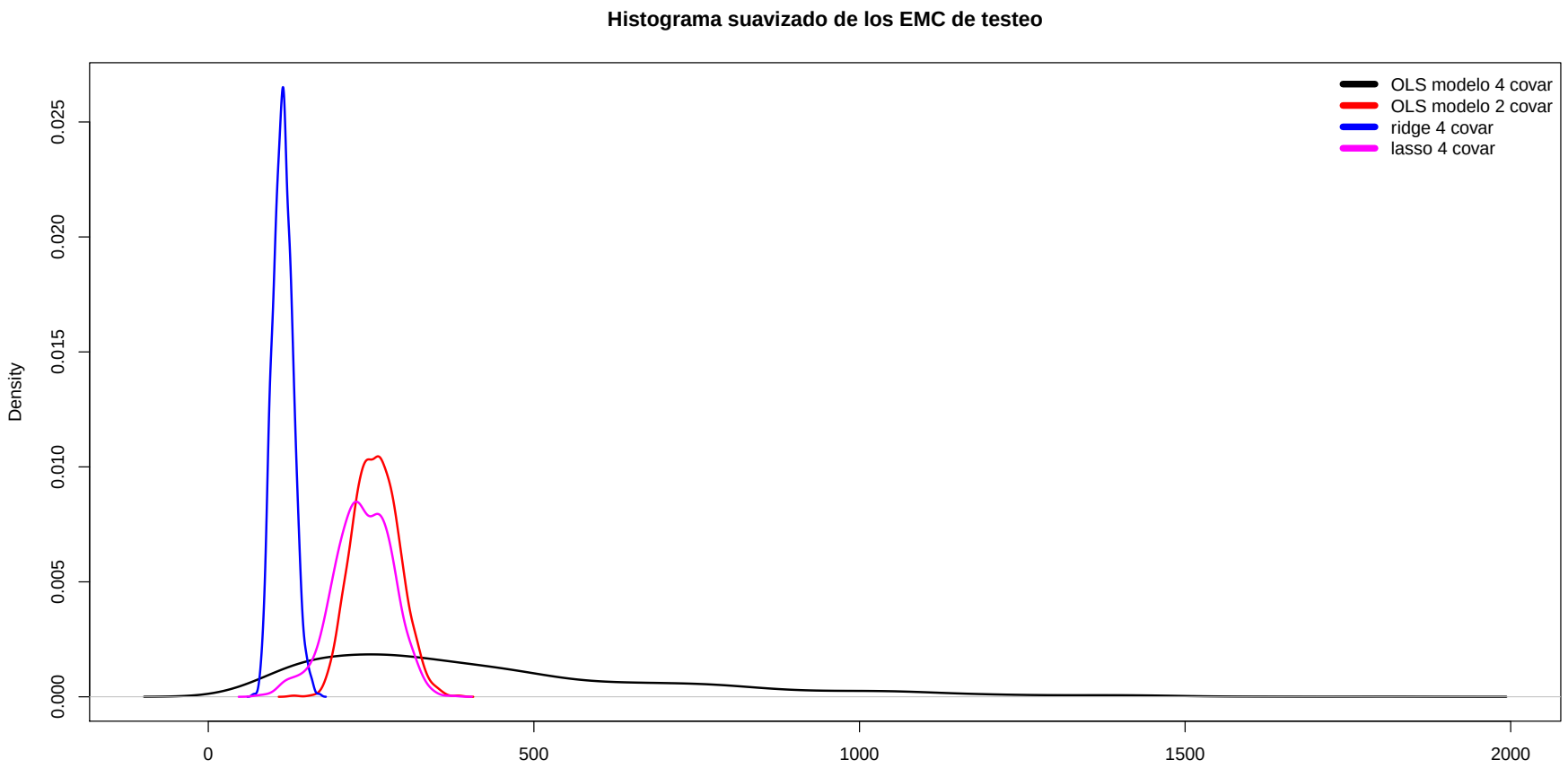
Comparemos los estimadores de los coeficientes, β_4

En rojo, el valor verdadero.

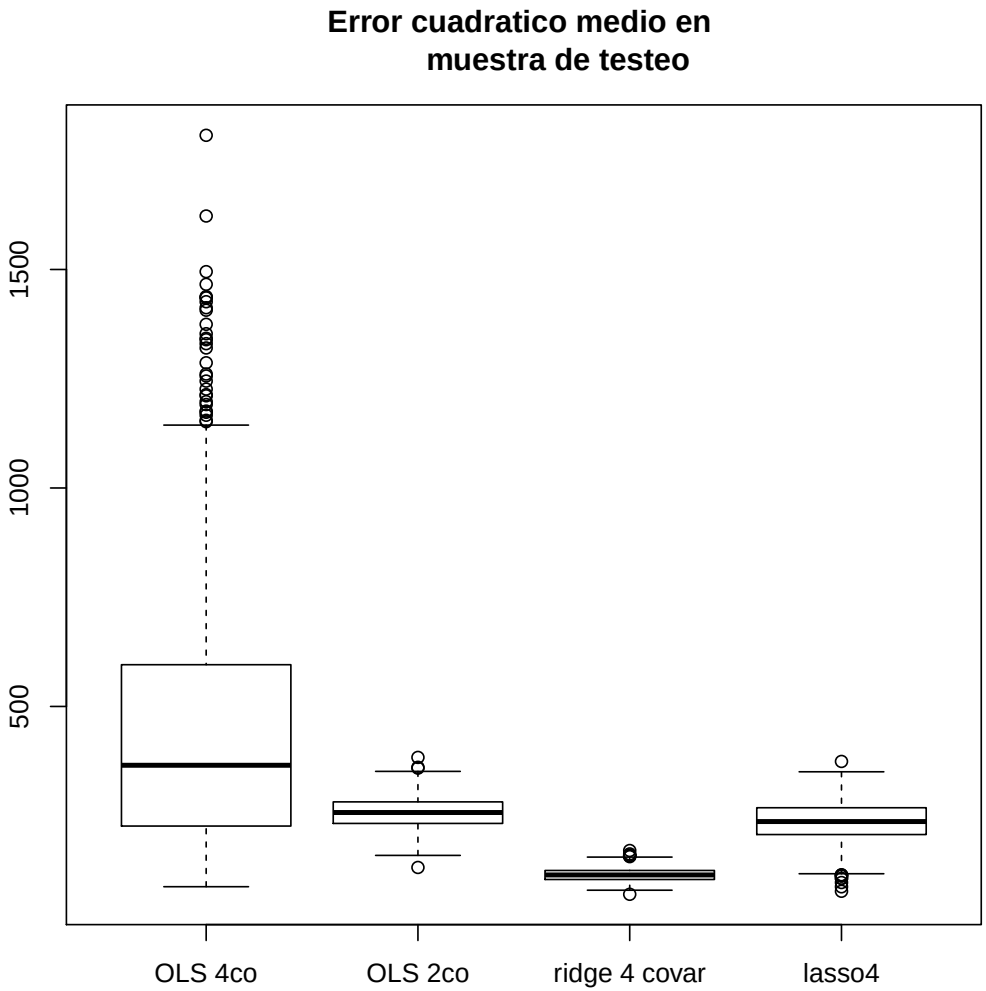


Lasso lo estima por cero.

Resultados de la simulación, EMCpred con muestra de testeo



Resultados de la simulación, EMCpred con muestra de testeo



Generalización: Elastic net

El estimador elastic net hace un compromiso entre las penalidades de lasso y ridge (Zou and Hastie 2005), resuelve el problema convexo

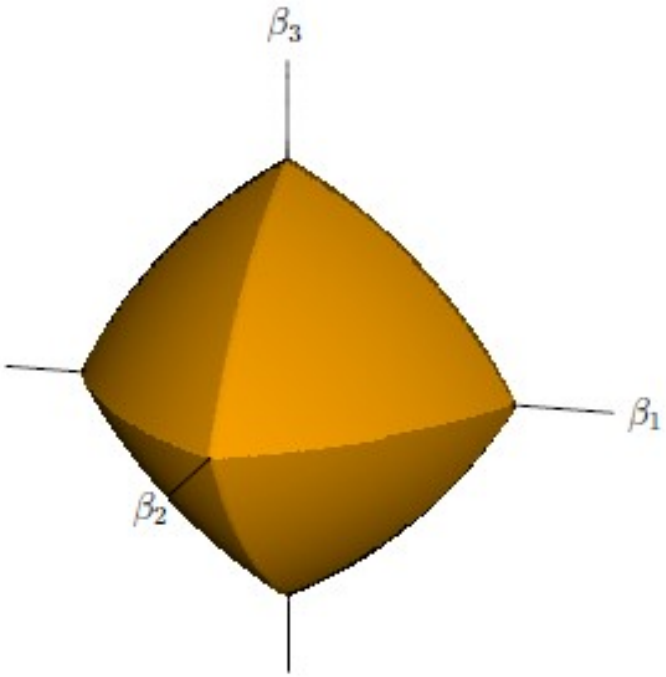
$$\underset{(\beta_0,\boldsymbol{\beta})\in\mathbb{R}\times\mathbb{R}^p}{\text{mín}} \left\{ \frac{1}{2}\sum_{i=1}^{n}\left(y_i-\beta_0-\mathbf{x}_i^T\boldsymbol{\beta}\right)^2+\lambda\left[\frac{1}{2}(1-\alpha)\|\boldsymbol{\beta}\|_2^2+\alpha\|\boldsymbol{\beta}\|_1\right]\right\}$$

donde el parámetro α puede ser elegido entre 0 y 1, incluyéndolos.
Por construcción, la penalización aplicada a un coeficiente individual (más allá del coeficiente de regularización λ) está dada por

$$\frac{1}{2}(1-\alpha)\beta_j^2+\alpha\,|\beta_j|$$

Cuando $\alpha=1$, es la penalización de norma 1, o sea el estimador resultante es el estimador lasso, cuando $\alpha=0$, la penalizadción resultante es el cuadrado de la norma 2 y por lo tanto se obtiene el estimador ridge.

La "bola" de elastic net con $\alpha = 0{,}7$



Bibliografía

- Buhlmann, Peter, van de Geer, Sara. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Series in Statistics.
- Trevor Hastie, Robert Tibshirani, Martin Wainwright. (2016). *Statistical Learning with Sparsity The Lasso and Generalizations* (SLS). CRC Presss.
- Hastie, T., Tibshirani, R. and Friedman, J. (2009), The *Elements of Statistical Learning: Data Mining, Inference and Prediction*, (ESL) second edn, Springer Verlag, New York.
- Gareth James, Daniela Witten, Robert Tibshirani, Trevor Hastie. (2013) *An Introduction to Statistical Learning with Applications in R* (ISL)