Guía SOA: Examen SRM

Facultad de Ciencias UNAM

Jasiel Antonio Coria Galán

Índice

El fin principal de la presente guía para obtener el título de actuario como alternativa a los talleres presenciales pospuestos por la pandemía es ayudar a los estudiantes de la facultad de ciencias que planean presentar el examen SRM de la SOA en los próximos meses a que tengan una visión de la prueba desde la perspectiva de un estudiante de la misma facultad y esperando que de alguna manera les ayude a aclarar el contenido del examen y descubrir su mejor manera de abordarlo.

Al lector:

La presente guía va a servirle querido lector como complemento de la guía: *ACTEX – Exam SRM Study Manual. Fall 2018 Edition*, misma que podrán encontrar en la hemeroteca de la facultad.

El contenido de los siguientes problemas es una explicación y con ejemplos gráficos de algunas de las preguntas teóricas de la guía. Este será un buen complemento para aquellos estudiantes cuya forma de aprendizaje sea visual para que el día del examen recuerden las gráficas y ejercicios aquí mostrados. De igual manera va dirigido a aquellos estudiantes que las preguntas teóricas no quedaron cien por ciento claras mientras avanzaban con los ejercicios de la guía, entonces espero que los ejemplos aquí mostrados les ayuden a alcarar sus dudas y lleguen con tranquilidad el día de su examen.

Tambien algo valioso que se llevarán de este trabajo será el código de cada uno de los problemas aquí mostrados. Todo se ha programado en R. Podrán encontrar el código en el siguiente link:

https://github.com/JasielCG/SRM_Code

Encontrarán el código de como ajustar una regresión lasso, una regresión ridge, los ensemble models, los códigos para los gráficos de ggplot, entre otros.

Como aclaración, hay secciones de la presente guía en las que doy consejos y opiniones del examen pero todos ellos están dados desde mi punto de vista y mi experiencia con el mismo. Comentaré la estructura que noté en el examen en la fecha que lo presenté, aclarando que pudo haber sufrido modificaciones con el paso del tiempo. El examen lo presenté en Septiembre de 2019 y tenía alrededor de un año de haber sido incluido como un nuevo examen de la SOA por lo tanto a partir de dicha fecha pudo haber sufrido modificaciones en su estructura y temas abordados.

Introducción.

Querido lector, toda la información relacionada con el examen vas a poder encontrarla en:

https://www.soa.org/education/exam-req/edu-exam-srm-detail/.

Como consejo te recomiendo explorar la página con detenimiento ya que cuenta con recursos que serán útiles para tu examen y con algunos consejos e instrucciones para el día de tu prueba.

Lo primero que vas a encontrar en dicha página es un syllabus por cada mes de aplicación del examen en cada uno de los cuales se detalla el contenido del examen que se aplicará en esa fecha en particular. Como lo mencioné anteriormente el contenido del examen puede variar un poco de mes a mes, aquí yace la importancia que te asegures que estás leyendo el syllabus correcto. Por ejemplo si vas a presentar el examen del mes de septiembre 2021 debes asegurarte que estás revisando el syllabus para el mes en cuestión y cerciorarte que la guía que estés usando para estudiar (aunque sea para exámenes anteriores) abarca todo el contenido del examen que estás a punto de presentar. Como último comentario acerca del syllabus, en la última página del mismo podrán encontrar un link que los va a llevar a una sección de preguntas ejemplo, las cuales serán buenos ejercicios para practicar una vez que ya has estudiado o sin en cambio apenas estás conociendo el examen estas preguntas te ayudarán a descubrir cómo es que está estructurado y qué tipo de preguntas esperarte para cada uno de los temas.

Lo siguiente que encontrarán en la página de la SOA es el calendario de meses en los que vas a poder presentar el examen durante el año así como el costo del mismo. Tener en cuenta que al ser estudiantes mexicanos tenemos un descuento en el precio de cada examen, el cual podrán encontrar en la sección de Fees.

Y por último en la página podrán encontrar consejos para el día del examen, recursos que la misma página recomienda para estudiar y el reglamento para la aplicación del examen en el cuál vienen los modelos de calculadoras permitidos por la SOA para presentar el examen. La calculadora que yo recomiendo para este examen es la Texas Instruments BA II Plus, en cualquiera de sus dos variantes: *BA II Plus* o *BA II Plus Professional*. La razón por la que les recomiendo alguna de estas dos calculadoras es porque en el examen podría requerirse hacer una regresión lineal simple a partir de datos en crudo y estas calculadoras te facilitan el cálculo de los parámetros de la regresión solo con ingresar los datos de la forma (x, y) lo cual se resume en ahorrar tiempo y esto en un examen de la SOA es de lo más valioso. ¡Cada segundo cuenta!.

Acerca del Examen.

El examen *Statistics for Risk Modeling* (SRM) es un nuevo examen de la Society of Actuaries aplicado por primera vez en Septiembre 2018. El examen consta de 35 preguntas y se tiene un tiempo límite de 3 horas y media para resolverlo.

El contenido del examen se basa principalmente en una introducción al análisis de datos, lo cual incluye modelos de regresión, modelos de series de tiempo, análisis de componentes principales, árboles de decisión, métodos de selección de modelos y análisis de clusters.

Las preguntas del examen SRM son de los siguientes dos formatos:

- *Ejercicios prácticos*: Este tipo de ejercicios se basan en la parte práctica de los modelos estadísticos y requieren realizar cálculos para llegar al resultado correcto. Como ejemplo se puede pedir calcular una regresión lineal (simple o múltiple) para un conjunto pequeño de datos (10 observaciones), hacer pruebas de hipótesis a partir de outputs de modelos previamente ajustados con el programa R, o calcular intervalos de confianza o de predicción para ciertos modelos.
- *Ejercicios teóricos*: Estas preguntas se encargan de medir que usted conoce cómo funcionan los modelos estadísticos, la motivación, las ventajas y desventajas de cada uno de ellos. A diferencia de otros exámenes impartidos por la SOA, el examen SRM comprende un mayor porcentaje dedicado a preguntas teóricas, las cuales según mi experiencia consistieron de un alrededor de un 30% de mi examen.

Recomendaciones.

En los cursos de la facultad se enseña todo lo necesario para aprobar este examen. Gran parte de su contenido se abarca en los cursos 2 y 3 de estadística los cuales complementados por un seminario en estadística que se enfoque al análisis de datos, análisis multivariado, ciencia de datos o machine learning, pueden darte todo el conocimiento necesario para aprobar el examen, únicamente requiriendo de la guía para familiarizarte con la metodología del examen y obtener velocidad en la resolución de ejercicios. Por lo tanto querido lector, podrías estar estudiando para tu examen SRM a la par que tomas tu seminario de estadística en séptimo u octavo semestre de la licenciatura.

Además es importante mencionar que para este examen no es importante saber programar en R pero sí será necesario saber cómo interpretar los outputs de este programa al ajustar un modelo estadístico. Esto lo puedes aprender en los cursos mencionados de la facultad o se explica el mínimo necesario en la guía.

Problemas

Problem 3.5.1

[**1**] (Options for handling outliers) Suppose that an outlier has been identified in a multiple linear regression analysis.

Which of the following is not recommended as a natural response?

(A) Immediately discard the outlier form the analysis.
(B) Discard the outlier, if it is discovered that the observation is caused by errors.
(C) Include the observation, but comment on its effects on the model.
(D) Carry the regression in two rounds, one round with the observation included and one round without the observation.
(E) None of the above.

Solution.

Un outlier en cualquier modelo nunca debe ser descartado sin antes hacer un análisis del mismo. Un outlier puede ser causado por un error de captura en los datos como de una observación real que sacarla del análisis puede desviar los resultados.

Mostrémoslo con un ejemplo. Sea χ la variable aleatoria de pago de un seguro de robo de autos (sin deducible ni límite máximo) tal que $\chi \sim LogNormal(12, 1)$. (Figure 1)

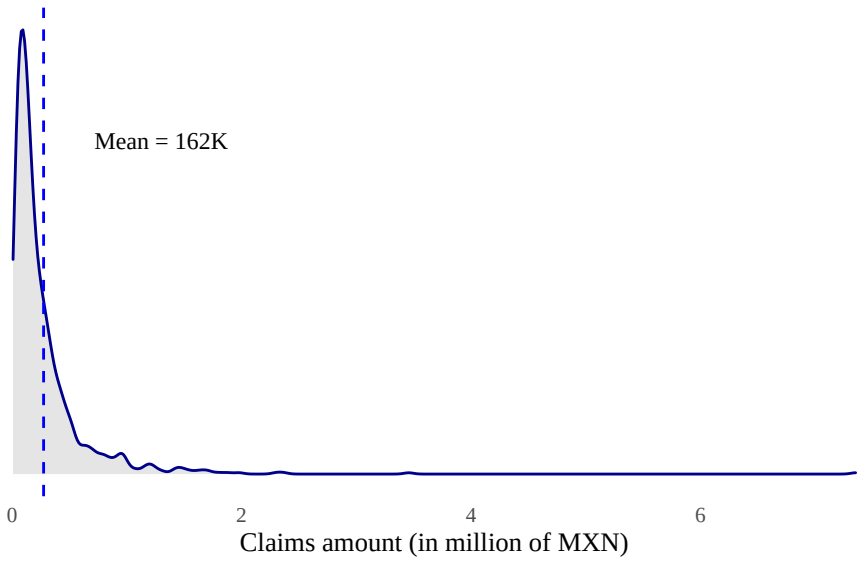

Figure 1: LogNormal Claims Density

Bajo estos supuestos esperamos un siniestro promedio de 162 mil pesos ($E[X] = e^{12}$).

Tomamos una muestra de 1,000 observaciones (siniestros). En nuestra muestra nos encontramos con un outlier, un siniestro de 7.3 millones de pesos.

Analicemos este outlier:

- ¿Existen autos de 7.3 millones de pesos?
- ¿La aseguradora cubre autos tan costosos?
- ¿Será un auto de 730 mil pesos pero hubo un error en el registro del mismo?

- ¿Tenemos más información como el modelo y el año del auto para poder investigar en el mercado su valor?

¿A qué se debe que no eliminemos el siniestro directamente de nuestro análisis/modelo que estemos haciendo? Principalmente debido a que este tipo de siniestros en una aseguradora deben ser tratados con mucho cuidado ya que un siniestro de este tamaño equivale a 45 siniestros promedios, por lo tanto si después de analizar este siniestro resulta que no es un error en los datos debemos tratarlo como que es posible que tengamos más siniestros de este estilo y plantear un límite máximo de retención o la posibilidad de un reaseguro.

Por lo tanto la respuesta es (A).

Problem 4.5.4

[1] (k-fold CV vs. LOOCV) You are given the following statements about different resampling methods.

I. k-fold cross-validation is a special case of the leave-one-out cross-validation (LOOCV).

II. k-fold cross-validation has a higher bias than LOOCV when k < n.

III. LOOCV tends to overestimate the test error rate comparison to k-fold cross-validation.

Determine which of the following statements are correct.

(A) I only.

(B) II only.

(C) II and III only.

(D) I, II and III.

(E) The correct answer is not given by (A), (B), (C) or (D).

Solution.

El inciso I es falso porque el LOOCV es un caso especial de k-fold CV, es equivalente a un n-fold CV.

k-fold cross-validation en efecto tiene un bias (sesgo) mayor que LOOCV para k<n y además k-fold cross-validation tiende a sobreestimar el test error en comparación a LOOCV. Por lo tanto, el inciso II es correcto y el III es falso. Veamoslo con un ejemplo sencillo.

Sea DF un conjunto de datos con dos variables, x una secuencia del 1 al 100 y y una muestra de 100 observaciones de una variable aleatoria $\chi \sim N(0,1)$. De esta manera tenemos un conjunto de datos (x,y) de 100 observaciones al rededor de la constante 0. El mejor modelo de regresión lineal (y en función de x) que se le puede ajustar a este conjunto de datos es la constante 0.

Sobre este modelo hagamos un k-fold CV para k $\in \{2, 5, 10, 15, ..., 90, 95, 100\}$. Notemos que para k=100 es equivalente a LOOCV.

Observemos como a medida que el valor del número de folds (k) incrementa, el error se hace cada vez menor, encontrando el menor error en K=100 que es el correspondiente a LOOCV. En este caso en particular dado que la varianza es constante por construcción (Var=1), esta disminución en el RMSE es debida a una disminución en el bias (sesgo).

La razón por la que el sesgo de un k-fold (k<n) sea mayor al sesgo de LOOCV es porque k-fold se entrena con menos datos, entonces es más probable que tengamos un error más grande por el simple hecho que el algoritmo aprendió de un conjunto de datos más pequeño.

Si ya vimos que LOOCV nos da un mejor estimador del error del modelo, ¿Por qué no usar siempre LOOCV?

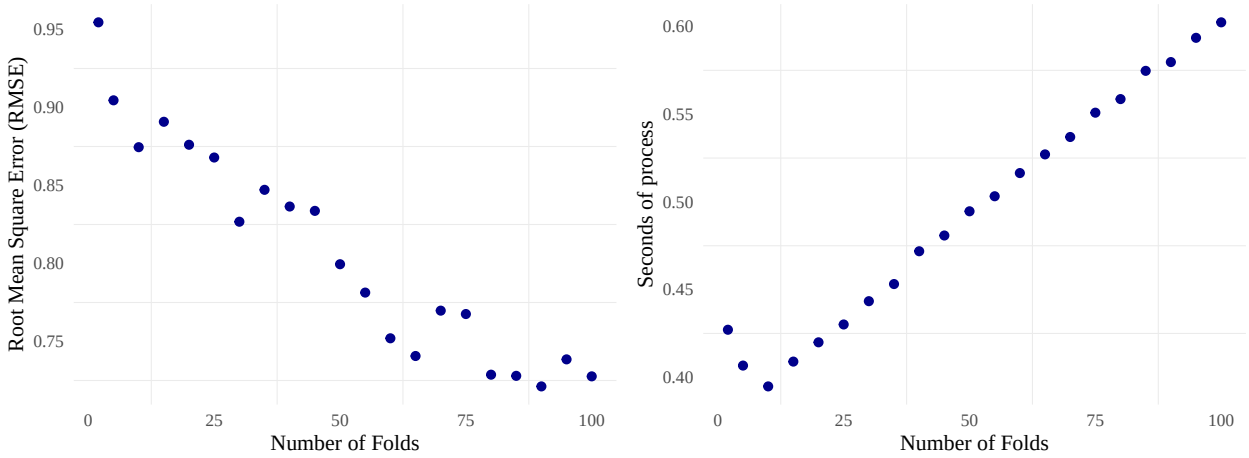

Figure 2: RMSE & Computational cost per Number of folds

La razón es por el costo computacional, si simplemente seguimos el algoritmo por definición de LOOCV el tiempo en el que tengamos el resultado puede ser muy grande, como se puede ver en (Figure 2) el tiempo incrementó de manera casi lineal en función del número de folds.

Claro que hay ciertos trucos como para regresión lineal que tenemos fórmulas para calcular LOOCV sin necesidad de entrenar todos los modelos pero en este caso lo quisimos mostrar de manera general, cómo aplicaría para un modelo arbitrario, no necesariamente que fuera lineal.

Problem 4.5.16

[1] (Effects of s on ridge regression/lasso) You estimate the regression coefficients in a linear model by minimizing:

$$RSS = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_1 + ... + \beta_k x_k)]^2$$

Subject to $\sum_{j=1}^{k} |\beta_j| \leq s$ for some $s \geq 0$.

Which of the following statements is/are correct?

I. This shrinkage method is known as ridge regression.

II. As s increases, the training error will increase.

III. As s increases, the test error will increase.

(A) I only.

(B) II only.

(C) III only.

(D) II and III only.

(E) The correct answer is not given by (A), (B), (C) or (D).

Solution.

Primero, es importante mencionar que el problema de minimizar que se nos presenta:

$$RSS = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_1 + ... + \beta_k x_k)]^2$$

Sujeto a $\sum_{j=1}^{k} |\beta_j| \leq s$ para algún $s \geq 0$.

Es equivalente a minimizar el siguiente problema:[1]

$$RSS = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_1 + ... + \beta_k x_k)]^2 + \lambda \sum_{j=1}^{k} |\beta_j|$$

Para algún $\lambda \geq 0$.

Dicho esto, veamos las respuestas para cada inciso.

I. Este método se conoce como Lasso.

II. Cuando λ es muy pequeña estamos quitando la restricción sobre $\hat{\beta}$, entonces cuando λ crece es cuando el efecto de la Lasso se hace mas evidente al poner mayor peso a la penalización sobre $\hat{\beta}$.

III. El test error muestra una forma de U en función de los valores de λ.

Aprovechemos este problema para explicar la regresión Lasso a grandes rasgos.

La motivación de la regresión Lasso (Least absolute shrinkage and selection operator) es construir un modelo de entrenamiento seleccionando únicamente un subconjunto del total de variables de la tabla de datos con el objetivo de mejorar las predicciones (usando a nuestro favor el Bias-Variance Tradeoff) y la interpretabilidad del modelo, incluir menos variables significa que nos estamos quedando de alguna manera con las variables más "importantes" del conjunto de datos o con mayor poder predictivo.

El modelo Lasso podría ser más útil cuando nuestra base de datos sufre de multicolinealidad, es decir, una o más variables pueden ser expresadas como combinación lineal del resto de variables, ya que el modelo con su selección de variables tiende a eliminar las variables colineales al no aportar mayor información para el entrenamiento. También puede ser buena opción cuando tenemos sospecha de que tenemos variables en nuestra base de datos que no aportan información predictiva.

Veamos la regresión LASSO con un ejemplo.

Trabajaremos con el dataset "College" de la librería ISLR, el cual es una recopilación de datos de 777 colegios de Estados Unidos de 1995. El problema que se quiere resolver es si podemos predecir el número de aplicaciones que recibieron cada uno de estos colegios en función de las demás variables disponibles.

Primero entrenaremos el modelo Lasso. Tal como se muestra en Figura 4 se realiza un 10-fold cross validation para muchos valores de λ para encontrar la λ óptima para nuestra regresión lasso.

Es importante saber interpretar el diagrama mostrado, para los primeros valores de $log(\lambda)$ el modelo usa las 17 variables y son las que nos arrojan el menor error, prácticamente estamos haciendo una regresión lineal. A medida que $log(\lambda)$ crece y llega a valores entre 5 y 6 podemos observar que el modelo se queda únicamente con 3 variables y el error prácticamente no incrementó, este es el beneficio de la regresión lasso que queremos aprovechar. Si seleccionamos la lambda que nos dé el error mínimo estaremos quedándonos con un modelo con 17 variables y será muy parecido a la regresión lineal, por esto la paquetería nos permite seleccionar la máxima λ que se encuentra a menos de un error estándar de la λ mínima, en nuestro caso $log(\lambda) = 5.72$ que nos permite quedarnos con 3 variables en el modelo.

Veamos la comparación del modelo lasso con la regresión lineal multiple ajustada por mínimos cuadrados.

Como se puede observar la regresión Lasso eliminó una gran cantidad de variables de nuestro modelo quedándonos con 3 de ellas más el intercepto lo cual es un modelo mucho más sencillo de interpretar y como

[1] http://personal.cimat.mx:8181/~mrivera/cursos/aprendizaje_maquina/ridge_lasso/Ridge_Lasso.html

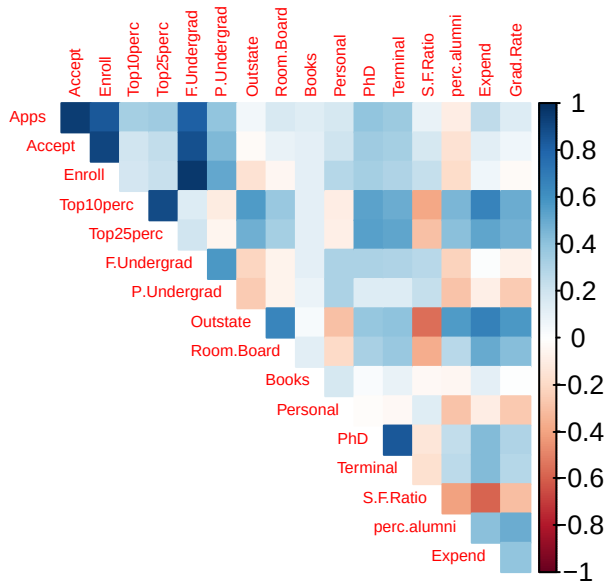

Figure 3: College: Matriz de correlación

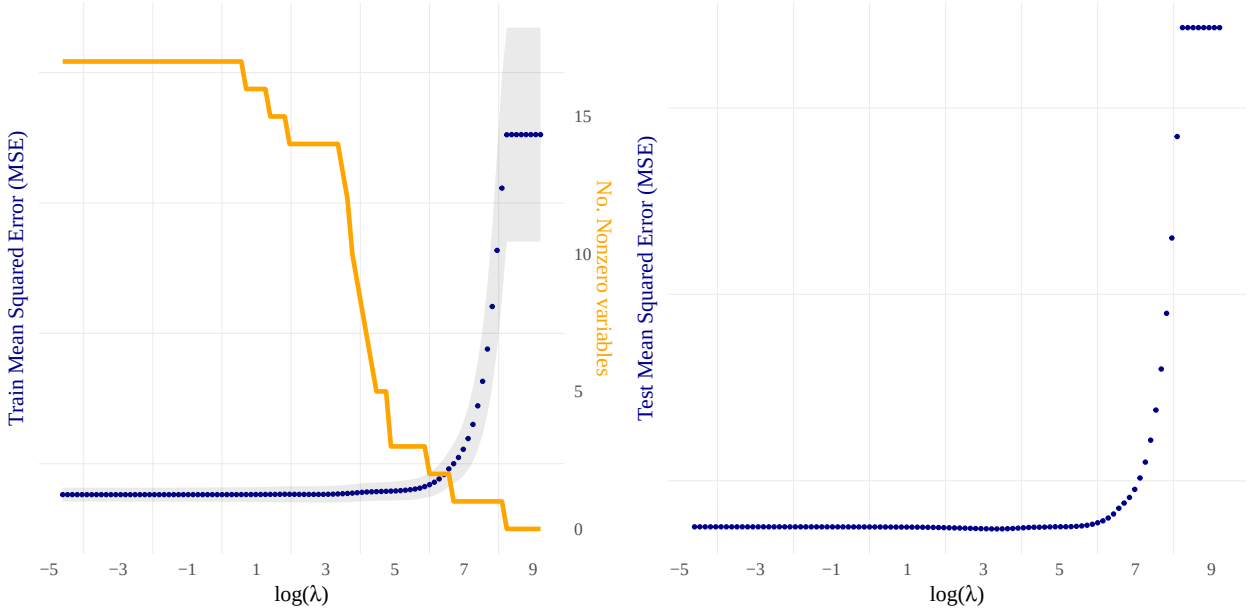

Figure 4: LASSO: Error Train & Test para distintos valores de lambda

consecuencia más sencillo de explicar a alguien que no esté muy familiarizado con estos temas. Lo que nos debemos estar preguntando es ¿a qué costo hemos eliminado esas variables?, ¿cuánto ha sido el incremento en nuestro error?

Table 1: LASSO: Comparación coeficientes regresión lineal (OLS) y regresión LASSO

	Multiple Linear Regression	LASSO Regression
(Intercept)	-445.084	-343.331
PrivateYes	-494.149	0.000
Accept	1.586	1.336
Enroll	-0.881	0.000
Top10perc	49.926	18.489
Top25perc	-14.234	0.000
F.Undergrad	0.057	0.000
P.Undergrad	0.044	0.000
Outstate	-0.086	0.000
Room.Board	0.151	0.000
Books	0.021	0.000
Personal	0.031	0.000
PhD	-8.678	0.000
Terminal	-3.331	0.000
S.F.Ratio	15.390	0.000
perc.alumni	0.179	0.000
Expend	0.078	0.014
Grad.Rate	8.668	0.000

El Test RMSE (Root mean square error) de la regresión lineal es de 1115.4.

El Test RMSE (Root mean square error) de la regresión lasso es de 1143.8.

Lo cual quiere decir que el error en la regresión Lasso es un 2.5% más grande que en la regresión lineal pero hemos eliminado 14 variables del modelo, haciéndolo mucho más interpretable.

¿Ha valido el trabajo?

Depende, en este caso la regresión lasso no nos ayudó a aprovechar el bias-variance tradeoff y disminuir el error pero nos dio un modelo mucho más interpretable seleccionando un subconjunto de variables con alto poder predictivo con un pequeño costo, un incremento del 2.5% en el error de predicción. Nos debemos preguntar, si lo que buscábamos era disminuir el error, este modelo no nos sirvió y podemos buscar algún otro, pero si estábamos buscando mayor interpretabilidad en el modelo lo hemos conseguido a un ligero costo de 2.5%.

Problem 4.5.17

[1] (Effects of λ on ridge regression/lasso) You estimate the regression coefficients in a linear model by minimizing:

$$RSS = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_1 + ... + \beta_k x_k)]^2 + \lambda \sum_{j=1}^{k} \beta_j^2$$

For some $\lambda \geq 0$.

Which of the following statements is/are correct?

I. This shrinkage method is known as ridge regression.

II. As λ increases, the training error will increase.

III. As λ increases, the test error will increase.

(A) I and II only.

(B) I and III only.

(C) II and III only.

(D) I, II and III.

(E) The correct answer is not given by (A), (B), (C) or (D).

Solution.

De la misma manera veamos las respuestas para cada inciso.

I. Verdadero.

II. Cuando λ es muy pequeña estamos quitando la restricción sobre $\hat{\beta}$, entonces cuando λ crece es cuando el efecto de la Ridge se hace mas evidente al poner mayor peso a la penalización sobre $\hat{\beta}$.

III. El test error muestra una forma de U en función de los valores de λ.

Aprovechemos este problema para explicar la regresión Ridge a grandes rasgos.

La motivación de la regresión Ridge es ajustar un modelo a nuestros datos que resuelvan el problema de una base de datos con multicolinealidad aprovechando el bias – variance tradeoff. La manera en que la regresión ridge lidia con este tipo de bases de datos es similar a la aproximación de modelo lasso que se explicó en el problema anterior, se pone una restricción al valor del vector de coeficientes $\hat{\beta}$ de nuestra regresión lineal, de esta manera el problema no solo tiene que reducir el error sino que tiene que encontrar el mejor modelo que mantenga un equilibrio entre disminuir el error del modelo y al mismo tiempo se evite que las $\hat{\beta}$ sean grandes sin que necesariamente se vuelvan cero como en la regresión lasso.

Veamos esto con el mismo ejemplo de la regresión lasso para poder entender mejor las diferencias entre estos dos modelos.

De igual manera trabajaremos con el dataset *College* de la librería ISLR. El cuál como se puede ver en la matriz de correlaciones (Figura 3) tiene algunos problemas de multicolinealidad entre sus variables, por ejemplo, las variables "enroll" y "accept" con una correlación muy cercana a 1.

Primero entrenaremos el modelo Ridge. Tal como se muestra en la figura 5, se realiza un 10-fold cross validation para muchos valores de λ para encontrar la λ óptima para nuestra regresión Ridge.

De nueva cuenta es importante interpretar el diagrama aquí mostrado. A diferencia de la regresión lassso este modelo siempre incluye las 17 variables para todos los valores de λ, dichas variables podrían tener una β muy cercana a cero pero no necesariamente se vuelven cero y las "eliminan" del modelo como la regresión lasso. Para los primeros valores de $log(\lambda)$ el modelo nos arroja el menor error en nuestros datos de entrenamiento, prácticamente estamos replicando una regresión lineal ajustada por mínimos cuadrados. A medida que $log(\lambda)$ crece y llega de nuevo a valores entre 4 y 6 podemos observar que el modelo comienza una tendencia alcista, entre estos valores podría estar nuestra mejor lambda ya que en esta parte del modelo el sesgo incrementa un poco (como se observa en el error de entrenamiento) pero si la varianza del modelo baja entonces el error podría disminuir tal como lo estamos buscando. Analizando la gráfica del Test error podemos decir que el modelo que minimiza el errror se encuentra al rededor de $log(\lambda) \approx 5$

Si seleccionamos la λ que nos dé el MSE mínimo nos quedaríamos con un modelo muy similar a la regresión lineal OLS, por esto la librería de R nos permite seleccionar la máxima λ que se encuentra a menos de un error

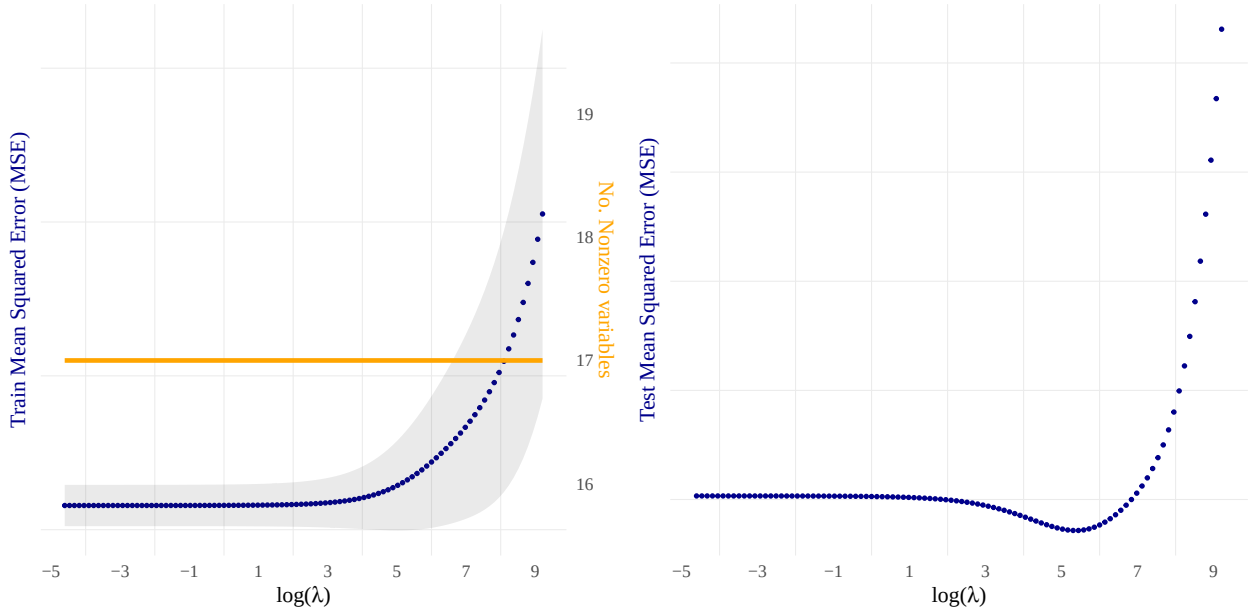

Figure 5: Ridge: Error Train & Test para distintos valores de lambda

estándar de la λ mínima, en nuestro caso $log(\lambda) = 5.02$ parecido a la predicción que hicimos anteriormente solo con ver el gráfico.

Veamos la comparación entre nuestro modelo ridge con la regresión lineal múltiple ajustada por mínimos cuadrados.

Table 2: Ridge: Comparación coeficientes regresión lineal (OLS) y regresión Ridge

	Multiple Linear Regression	Ridge Regression
(Intercept)	-445.084	-1468.326
PrivateYes	-494.149	-527.878
Accept	1.586	1.005
Enroll	-0.881	0.431
Top10perc	49.926	25.806
Top25perc	-14.234	0.550
F.Undergrad	0.057	0.073
P.Undergrad	0.044	0.024
Outstate	-0.086	-0.024
Room.Board	0.151	0.199
Books	0.021	0.129
Personal	0.031	-0.008
PhD	-8.678	-4.028
Terminal	-3.331	-4.811
S.F.Ratio	15.390	13.022
perc.alumni	0.179	-8.545
Expend	0.078	0.076
Grad.Rate	8.668	11.267

Como se puede observar la regresión ridge ha puesto límites en algunas variables haciendo que los coeficientes de algunas de ellas sean más pequeños, por ejemplo en la variable "Top10perc".

Ahora las preguntas que nos debemos hacer son: ¿Qué tan bueno resultó ser nuestro modelo?, ¿Cuál fue el costo/beneficio de haber restingido las β para estas variables?, ¿Cuánto ha sido el incremento/disminución en nuestro error?.

El RMSE de la regresión lineal es de 1,115.4.

El RMSE de la regresión ridge es de 1,055.6.

Lo cual quiere decir que el error en la regresión ridge es un 5.35% más pequeño que en la regresión lineal.

¡Hemos mejorado el error en nuestro modelo!

De nuevo nos preguntamos ¿Ha valido el trabajo? Depende, en este caso la regresión ridge nos ayudó a aprovechar el bias-variance tradeoff y disminuir el error, por lo tanto si es lo que buscábamos lo hemos conseguido, pero a diferencia de la regresión lasso no hemos conseguido un modelo más interpretable que la regresión lineal OLS ya que mantenemos las mismas 17 variables.

Solo para ponerlo a discusión, en estos ejemplos de la regresión lasso y ridge no hemos estandarizado las variables pero podría ser recomendable ya que la penalización de las β afecta de manera proporcional a la escala en la que se encuentren cada una de las variables, un buen ejercicio para el lector es tomar el código de los dos modelos anteriores y modificarlos para agregar la normalización de las variables.

(Hint: No debería tomar más de dos líneas de código con *dplyr* escalar las variables)

Problem 5.4.24

[1] (Theoretical knowledge about AIC and BIC) Determine which of the following GLM selection consideration is true.

(A) The model with the largest AIC is always the best model in model selection process.

(B) The model with the largest BIC is always the best model in model selection process.

(C) The model with the largest deviance is always the best model in model selection process.

(D) Other things equal, when the number of observations >1000, AIC penalizes more for the number of parameters used in the model than BIC.

(E) Other things equal, when the number of observations >1000, BIC penalizes more for the number of parameters used in the model than AIC.

Solution.

El mejor modelo es el que tiene el menor AIC, BIC y devianza, por lo tanto (A), (B) y (C) son falsos.

Mostremos esto con un ejemplo sencillo.

Consideremos una muestra de tamaño 10,000 de 3 variables aleatorias independientes idénticamente distribuidas (v.a.i.i.d) $X_i \sim Normal(2, 2)$ y definamos nuestra variable respuesta como:

$Y = X_1 + 2X_2 + 3X_3 + e$

Ruido blanco: $e \sim Normal(0, 1)$

Para el primer modelo consideraremos solo la variable respuesta en función del intercepto ($Y \sim 1$), el cual es el modelo más sencillo y con menor poder predictivo en este ejemplo. El segundo modelo será el que la variable respuesta está en función de todas las variables.

Table 3: Comparación AIC, BIC, Devianza

	Intercept Model	Full Model
AIC	68840.57	28355.279
BIC	68854.99	28391.331
Deviance	571557.42	9966.565

El mejor modelo es el segundo, por construcción y en efecto muestra el menor error y cuenta con el menor AIC, BIC y devianza.

Para los incisos D y E, de las penalizaciones del AIC y BIC se deduce que:

$pln(n) > 2p \Leftrightarrow n > e^2 = 7.3891$

Donde, p = número de parámetros y n = número de observaciones.

De lo cual se observa que el correcto es el inciso E.

Problem 8.7.2

[1] (Sample problem) Consider the following statements.

I. Pruning a classification tree always leads to a decrease of the training error.

II. Pruning a classification tree always leads to an increase of the training error.

III. Pruning a classification tree always leads to a decrease of the test error.

IV. Pruning a classification tree always leads to an increase of the test error.

Which of these statements is correct?

(A) I only.

(B) II only.

(C) III only.

(D) IV only.

(E) None of the statements is correct.

Solution.

Podar un árbol siempre nos lleva a un árbol mas pequeño y como consecuencia el error de entrenamiento siempre será mayor o igual al árbol original. El efecto de podar un árbol esperamos que nos lleve a una disminución del Test error pero esto es incierto, no es una regla en general, en ocasiones puede aumentar o en otras disminuir. Por lo tanto, la respuesta correcta es la (B). En la figura 6 mostramos un ejemplo.

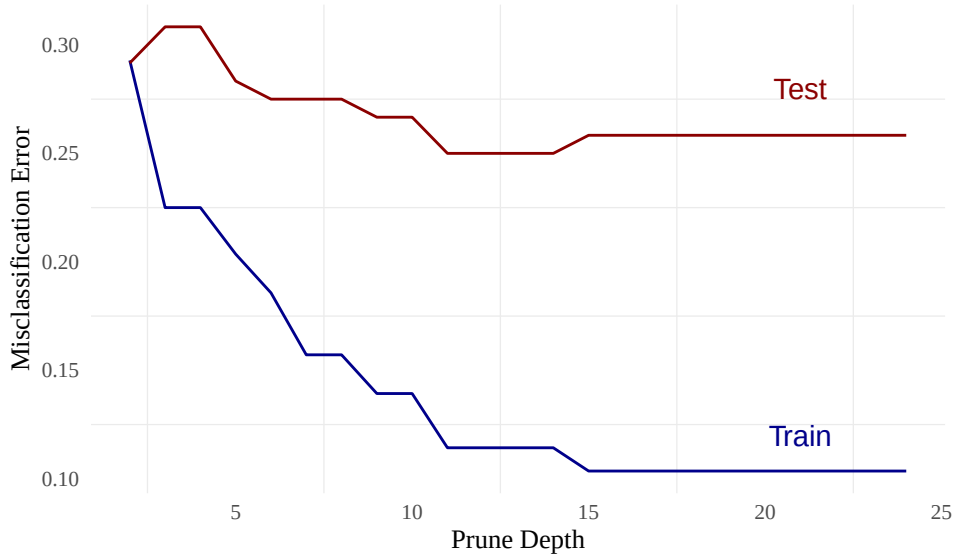

Figure 6: Misclassification Error per Prune Depth

Problem 9.6.1

[1] (Sample problem) Consider the following statements about ensemble learning methods. Determine which of these statements is/are true.

I. Increase the number of individual trees used to construct a bagged tree, increases the probability of overfitting the data.

II. Increasing the number of individual trees used to construct a random forest, increases the probability of overfitting the data.

III. Increasing the number of individual tress used to construct a boosted tree, increases the probability of overfitting the data.

(A) I only.

(B) II only.

(C) III only.

(D) I, II and III.

(E) The correct answer is not given by (A), (B), (C) or (D).

Solution.

Antes de resolver este problema les paso por un breve resumen de que trata cada uno de los algoritmos para que recuerdes sus diferencias.

Bagged tree: El modelo es entrenado para B diferentes arboles de decisión, luego el output del modelo es el promedio de todas estas predicciones para el caso de regresión o para el caso de clasificación la respuesta será la cual dicte la mayoría de los árboles. Es importante mencionar que cada uno de estos modelos es entrenado con un bootstrapped data set(muestra con reemplazo del dataset original), de tal manera que si los datos son muy similares entre si o las muestras son muy parecidas entonces las predicciones se volverán similares entre cada árbol y el bagging habrá perdido su poder de predicción.

Random forest: Descorrelacionando los árboles del modelo. Un randon forest considera B arboles de decisión individuales donde cada uno de ellos únicamente considera un subconjunto de P' variables explicativas. Es comun el uso de P' = $\sqrt{P}$. Aquí el secreto está en encontrar la P' que nos baje la correlación entre los árboles

e incremente el poder predictivo. Como resumen, es la misma idea que en el Bagged Tree pero en este caso tomamos una selección del total de variables para cada uno de los árboles a entrenar. Bagged Tree es un caso especial de Random Forest donde P' = P, hace uso de todas las variables explicativas.

Boosted tree: Esta es una aproximación diferente con el mismo objetivo, mejorar el modelo de predicción del árbol de decisión. Lo que se hace es crear un modelo $f_1(x)$ de árbol de decisión sobre el training set, luego se calculan sus residuos $e_1 = y - f_1(x)$, posteriormente se calcula un nuevo modelo $f_2(x)$ con variable respuesta los residuaos e_1. De esta manera podemos continuar entrenando modelos $f_k(x)$ con variable respesta e_{k-1} hasta que nosotros decidamos. El modelo final está dado por $F(x) = f_1(x) + f_2(x) + \ldots + f_n(x)$ donde n es el número de modelos entrenados.

Comúnmente se les agrega a estos modelos un parámetro de convergencia λ para que controlemos la velocidad de convergencia al modelo "óptimo" ($F(x) = f_1(x) + \lambda f_2(x) + \ldots + \lambda f_n(x)$), más lento puedes controlar mejor el riesgo de overfitting, pero para llegar a un error pequeño tendremos que entrenar más modelos (nuestra n será mas grande). A comparación, una λ más grande (es dificil decir qué es "grande") te lleva a disminuir el error más rápido por lo que llegas a un error 0 con una menor cantidad de modelos entrenados, el problema con hacerlo más rápido es que el riesgo de overfitting incrementa rápidamente.

Una forma en la que podremos encontrarnos Boosted trees en foros en internet será la versión mas sencilla, cada uno de los modelos $f_i(x)$ es un árbol de decisión de dos hojas haciendo que el modelo general dependa únicamente del número de modelos que entrenemos y no de la produndidad de los mismos.

Una vez aclarado este breve resumen sobre cada uno de los modelos mostremos las respuestas.

I. Para Bagged tree incrementar el número de árboles en la Bag reduce la probabilidad de overfitting ya que los datos es menos probable que sean similares entre sí.

II. Para Random forest incrementar el número de árboles en la Bag reduce la probabilidad de overfitting ya que los datos es menos probable que sean similares entre sí.

III. Para Bossted tree incrementar el número de árboles del modelo incrementa la probabilidad de overfitting porque llegaremos a un error de entrenamiento 0 pero esto no necesariamente se reflejará en una disminución en el Test error.

Mostremoslo con un ejemplo.

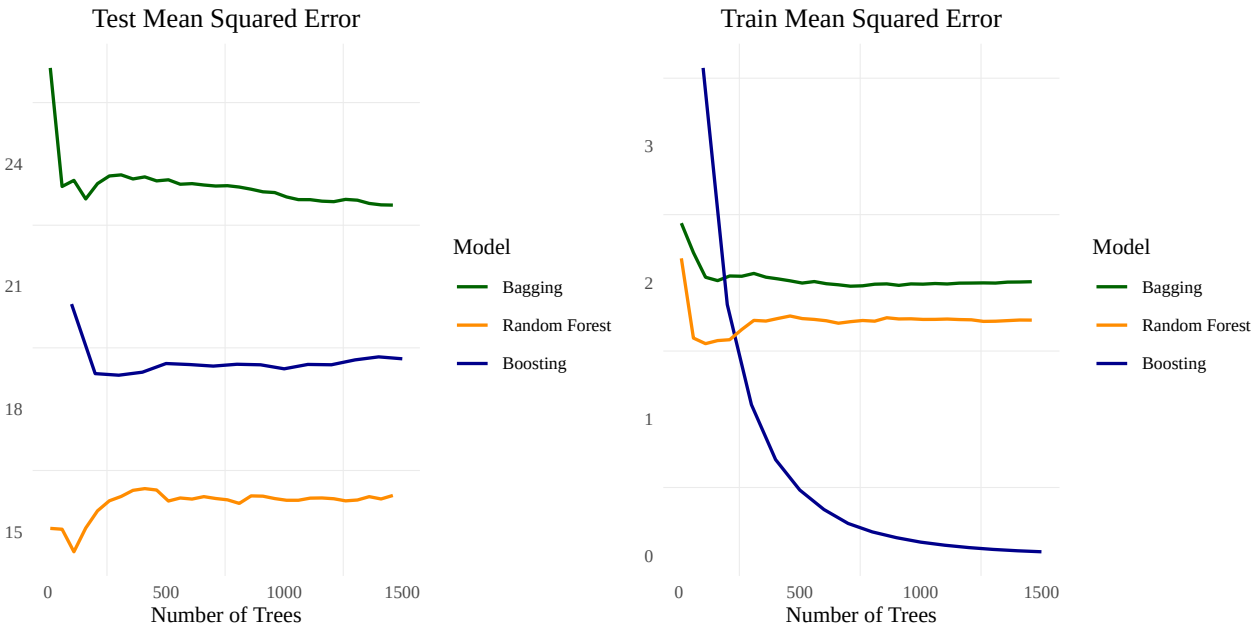

Figure 7: Train & Test Errors per Model

Se ajustó al mismo conjunto de datos cada uno de los tres modelos. Como se puede observar en figura 7 para bagging y random forest cuando incrementa el número de árboles en los modelos no reduce el error de entrenamiendo, de hecho se mantiene constante en cierto punto, así mismo se ve este mismo patrón en el test error. Por otra parte para el modelo de boosting a medida que el número de árboles incrementa el error de entrenamiento cae rápidamente a cero, sin embargo el test error no se ve afectado de la misma manera, esta es la manera en la que se ve el sobreajuste (overfitting), un train error que tiende a cero y un test error que ni se afecta.

Problem 9.7.2

[1] Consider the following statements about decision trees and linear regression models.

I. Decision trees are more flexible than linear regression models.

II. Decision models are more accurate than linear regression models.

III. Decision trees are easier to interpret than linear regression models.

(A) I is only correct.

(B) II is only correct.

(C) III is only correct.

(D) I, II and III are correct.

(E) None of the statements are correct.

Solution.

I. En efecto, los árboles de decisión son un modelo más flexible, enteniendo por flexible que es capaz de aprender de patrones mas complejos, no necesariamente lineales tal como se muestra en la figura 8.

II. Falso. No hay algo como un algoritmo más preciso, lo que podemos encontrar es el algoritmo más preciso para un conjunto de datos en específico. Para mostrar que a pesar de que el árbol de decisión es un modelo mas flexible no significa que siempre es la mejor alternativa. Como se muestra en la figura 9 se puede observar como la regresión lineal tiene un mejor ajuste para este conjunto de datos que el árbol de decisión.

III. Un árbol de decisión con pocas ramificaciones puede ser interpretable pero puedes llegar a un punto en el que para que puedas reducir el test error tienes que hacer un árbol muy profundo que pierde por completo su interpretabilidad. Si sumado a esto le agregas los métodos de ensemble learning: bagging, boosting o random forest; la interpretabilidad se pierde por completo. Por otra parte, los modelos lineales son un poco más sencillos de interpretar, ya que se basan en que si la variable independiente incrementa x, entonces la variable respuesta incrementará en βx.

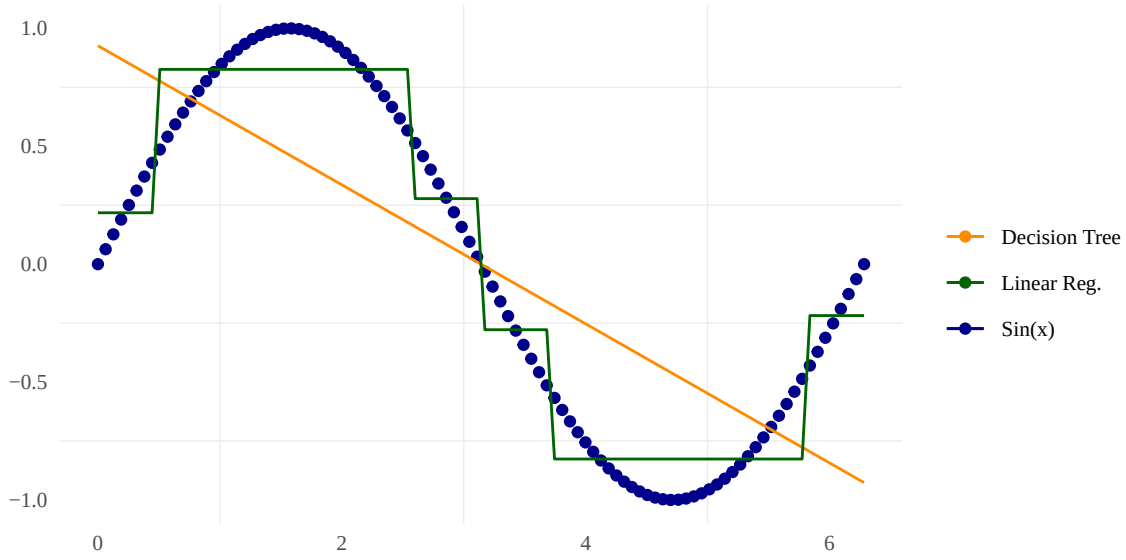

Figure 8: Flexibility of Decision Tree vs Linear Model I

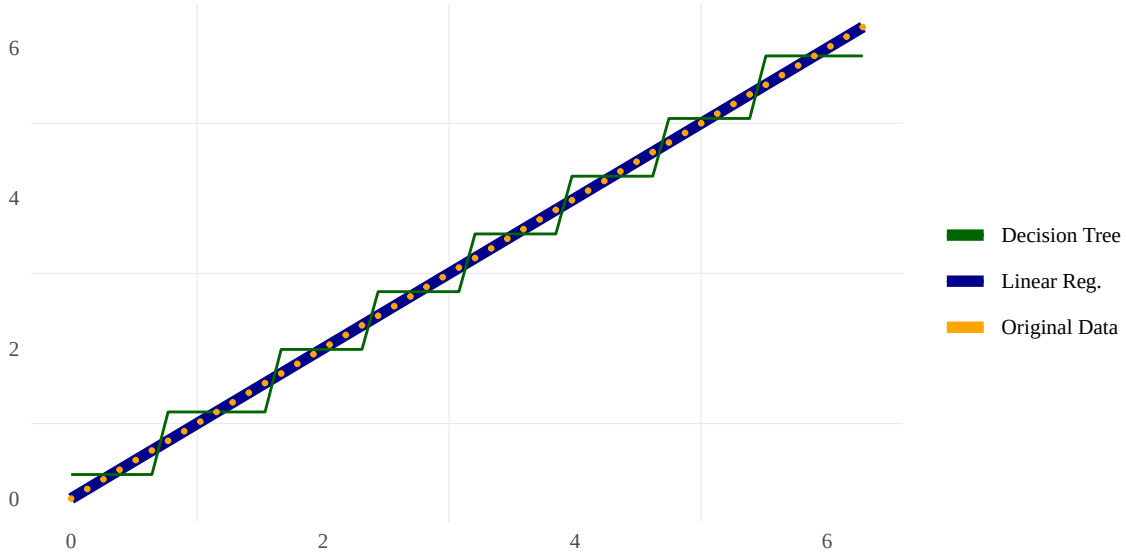

Figure 9: Flexibility of Decision Tree vs Linear Model II

Problem 10.2.4

[**1**] Consider the following statements:

I. Principal Component Analysis (PCA) provide low-dimensional linear surfaces that are closest to the observations.

II. The first principal component is the line in p-dimensional space that is closest to the observations.

III. PCA finds a low dimension representation of a data set that contains as much variation as possible.

IV. PCA serves as a tool for data visualization.

Determine which of the statements are correct.

(A) Statements I, II and III only.

(B) Statements I, II and IV only.

(C) Statements I, III and IV only.

(D) Statements II, III and IV only.

(E) Statements I, II, III and IV are all correct.

Solution.

Los componentes principales tienen dos interpretaciones:

1. Los vectores propios (loading vectors) representan las direcciones en que el espacio original tiene mayor varianza.
2. Los componentes principales dan las superficies lineales de menor dimensión que son mas cercanas a los datos.

De esta manera se sigue que III es cierto, por construccion. I es cierto por la segunda interpretación gráfica de PCA y II es un caso especial de I.

Para IV, lo mostramos en un ejemplo. Sea $N = 1000$ nuestro tamaño de muestra y sean x_1, x_2 y x_3 v.a.i.i.d. que se distribuyen $Normal(0, 1)$. Sea $x_4 = 0$ si $x_1 < 0$ y $x_4 = 1$ en otro caso.

Dicho esto nos podemos imaginar al conjunto de datos para las variables x_1, x_2 y x_3 como una nube esférica de puntos aleatorios, centrada en (0,0,0) y de radio 1. La variable x_4 viene a separar la esfera en dos conjuntos de datos y está estrechamente relacionada con x_1. Lo que esperamos de una representación en dos dimensiones sería una nube de puntos que de alguna manera se muestre esta separación. Como lo observamos en la figura 10, la representación con dos componentes principales nos brinda esta idea, que los puntos son una nube y además se separan de alguna manera en dos conjuntos de datos. Adicional, como observación, los loadings de x_1 y x_4 están estrechamente relacionados, casi apuntando en la misma dirección.

Por lo tanto IV tambien es correcto. La respuesta correcta es el inciso (E).

Problem 10.2.5

[**1**] (Sample Question) Consider the following statements:

I. The proportion of variance explained by an additional principal component increases as more principal component are added.

II. The cumulative proportions of variance explained increases as more principal components are added.

III. Using all possible principal components provides the best understanding of the data.

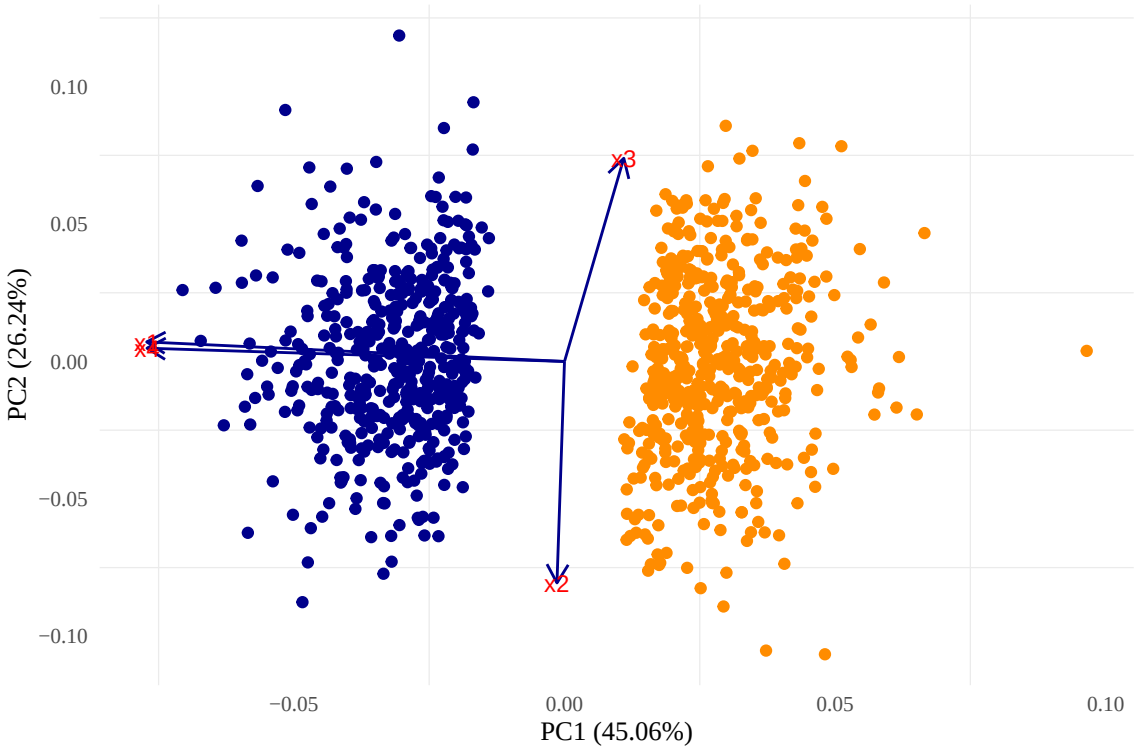

Figure 10: PCA as a Data visualization tool

IV. A scree plot provides a method for determining the number of principal components to use.

Determine which of the statements are correct.

(A) Statements I and II only.

(B) Statements I and III only.

(C) Statements I and IV only.

(D) Statements II and III only.

(E) Statements II and IV only.

Solution.

Para darte una mejor idea de los siguientes puntos, como recomendación observa a la par la figura 11.

I. Incorrecto. La proporción de varianza disminuye con cada componente principal añadido.

II. Correcto. La varianza acumulada en efecto aumenta con cada componente principal añadido.

III. Incorrecto. El objetivo de los componentes es usar el menor número de componentes posibles explicando la mayor cantidad de varianza. Entonces usar todos los componentes no necesariamente da un mejor entendimiento de los datos.

IV. Correcto. Es un plot de la proporción de varianza explicada.

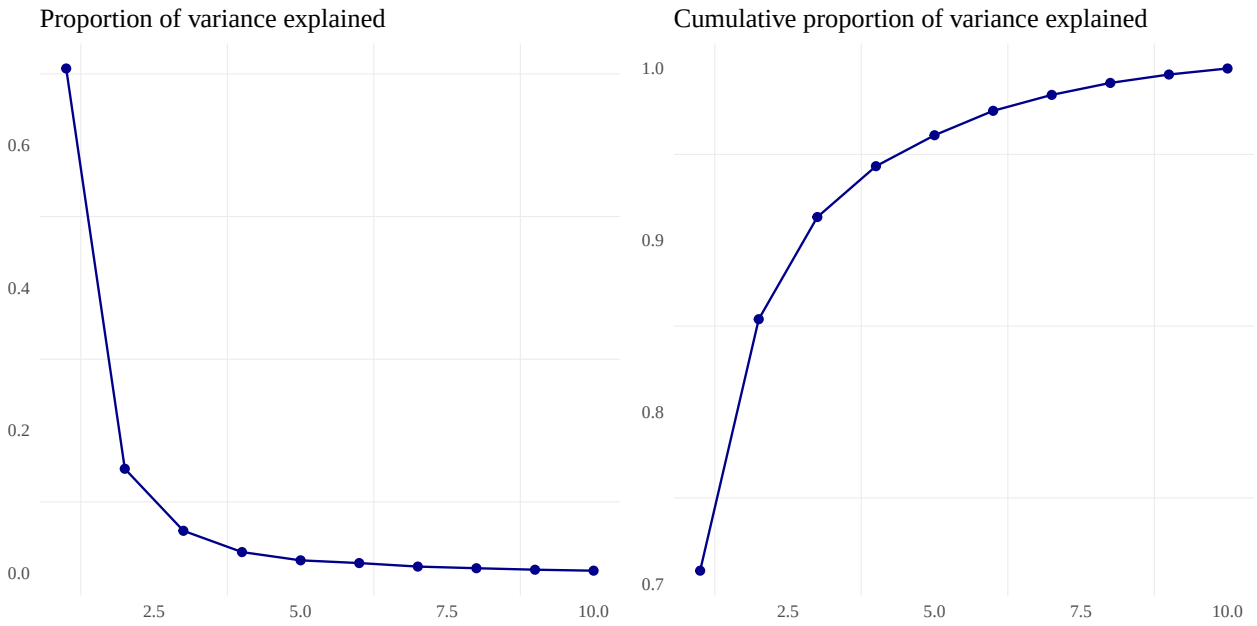

Figure 11: PCA Scree Plot

Problem 10.2.6

[**1**] Consider the following statements.

I. There is no well-accepted objective way to decide how many principal components are enough in a principal component analysis.

II. The principal components in a principal component regression represent the direction in which original predictors/variables show the most variation.

III. In a PCR, the directions in which original predictors/variables show the most variation are the same as the directions that are most associated with the response variable.

IV. PCR is a type of feature selection method.

(A) Statements I and II only.

(B) Statements I and III only.

(C) Statements I and IV only.

(D) Statements II and III only.

(E) Statements II and IV only.

Solution.

Expliquemos cada una de las respuestas.

I. Verdadero. Es común usar el Scree Plot como criterio pero no hay un consenso del umbral a elegir para determinar el mínimo número de componentes principales a usar. Es a criterio propio el número de componentes a usar.

II. Verdadero. Por definición de componentes principales.

III. Falso. Veámoslo con un contrajemplo. Consideremos las variables x_1 y x_2 como se muestran en la figura 12. Les realizamos un análisis de componentes principales (PCA) y obtenemos los dos componentes que se muestran en rojo y en verde con sus respectivas direcciones. Posteriormente con un poco de

álgebra lineal creamos el plano que dependa únicamente de la dirección de PC2 (1,-1) y creamos la variable respuesta $y = x_1 - x_2$. Ajustamos una regresión lineal al conjunto de datos que acabamos de crear, tal como se muestra en la Tabla 4 confirmamos con los vectores propios y los coeficientes de la regresión que la variable respuestá esta asociada con el PC2 que es el componente de menor peso, es decir, la dirección en donde las variables predictoras tienen la mayor varianza no están relacionadas con la dirección que mejor explica la variable respuesta.

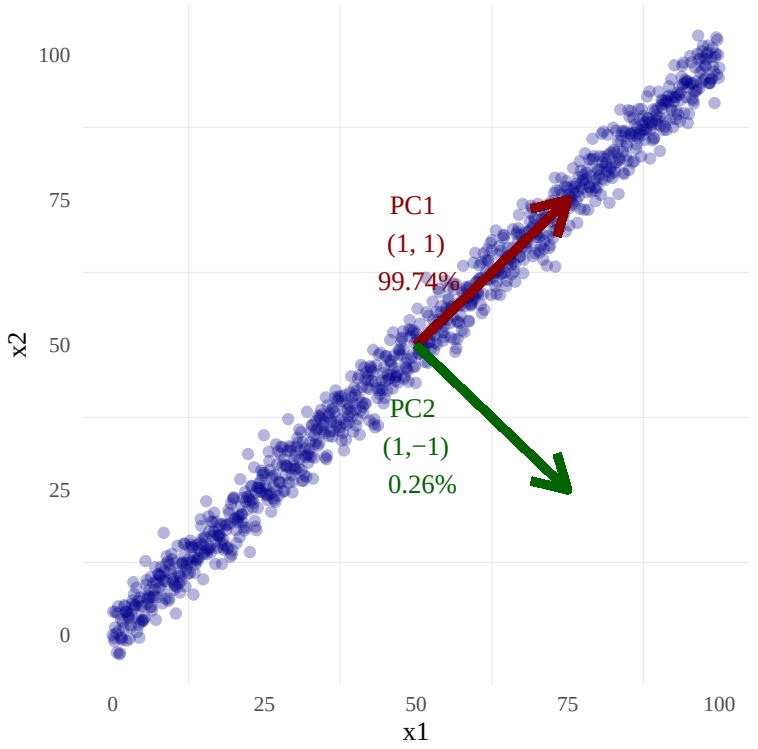

Figure 12: Vectores de carga para Contraejemplo 10.2.6 III

Table 4: PCA Loading vectors & Linear Regression

	PC1	PC2	Linear Reg.
x1	1	1	1
x2	1	-1	-1

IV. Falso. PCR no es un modelo de selección de variables porque recordemos que cada componente principal es una combinación lineal de todas las variables originales, es por ello que no necesariamente los componentes principales tienen interpretaciones de caracteristicas especiales de las variables originales.

Problem 11.4.1

[**1**] (Sample Question 2) Determine which of the following statements is/are true when deciding the number of clusters.

I. The number of clusters must be pre-specified for both K-means clustering and hierarchical clustering.

II. The K-means clustering algorithm is less sensitive to the presence of outliers than the hierarchical clustering algorithm.

III. The number of clusters may be determined using validation data.

(A) I only.

(B) II only.

(C) III only.

(D) I, II and III.

(E) The correct answer is not given by (A), (B), (C) or (D).

Solution.

I. Solo se especifica a priori el número de clusters para K-means.

II. Ambos algoritmos son muy sensibles a la presencia de outliers, tal como se puede observar en la figura 13, ambos algoritmos obligan al outlier a pertenecer obligatoriamente a un cluster.

III. Una forma de determinar el número de clusters es separar el conjunto de datos en entrenamiento y test. Primero usamos el conjunto de entrenamiento para determinar los clusters y posteriormente usamos el conjunto test para medir la precisión de los clústers.

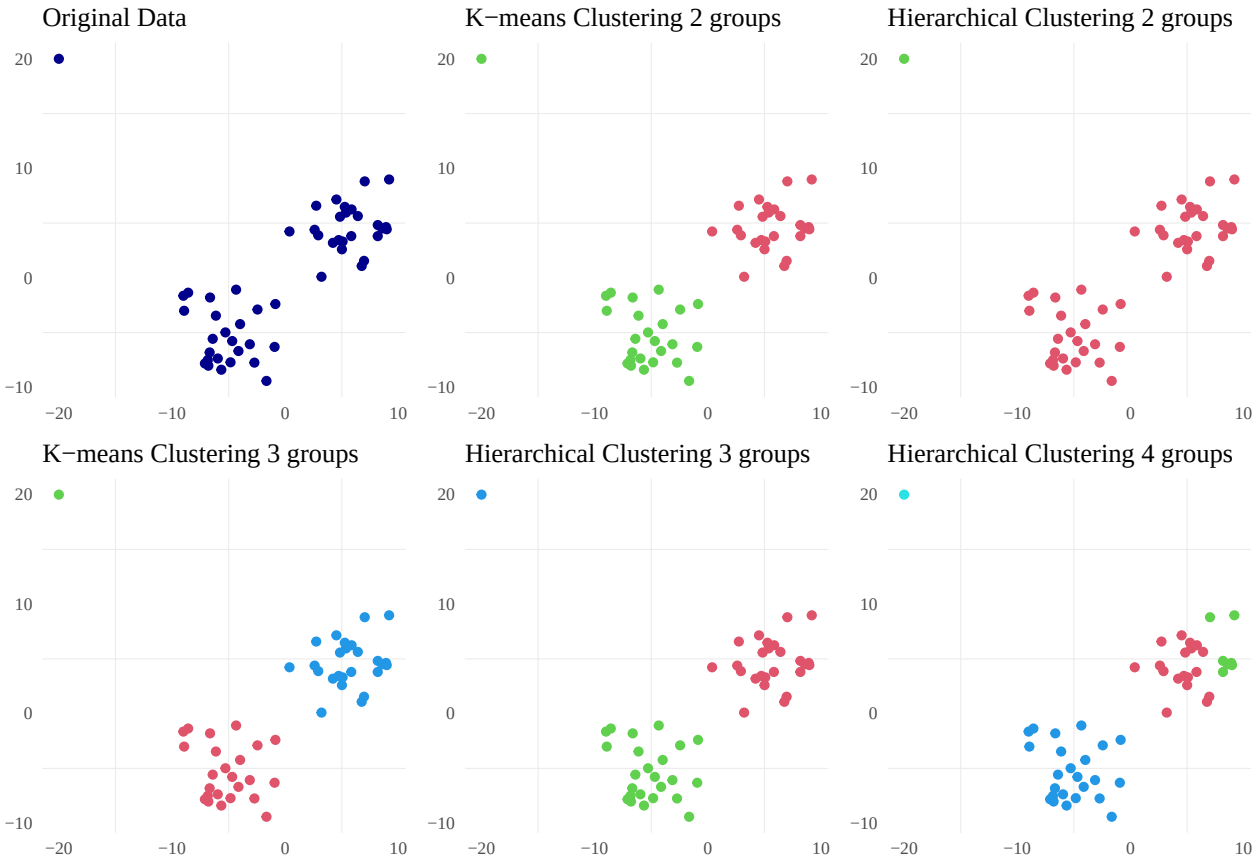

Figure 13: Outliers in Cluster Algorithms

Bibliografía

Todos los ejercicios mostrados en la presente guía provienen de las siguientes fuentes:

- **[1]** SOA Exam SRM Study Manual, Runhuan Feng, Daniël Linders, Ambrose Lo, 2018, USA: ACTEX
- **[2]** An Introduction to Statistical Learning, with Applications in R, James, Witten, Hastie, Tibshirani, 2013, New York: Springer.

Muchas gracias y les deseo el mejor de sus éxitos en su examen y en la vida.

Jasiel CG