

**DigitalHouse** >  
Coding School

# DATA SCIENCE

UNIDAD 2  
MÓDULO 3

Regularización

Septiembre 2017

1

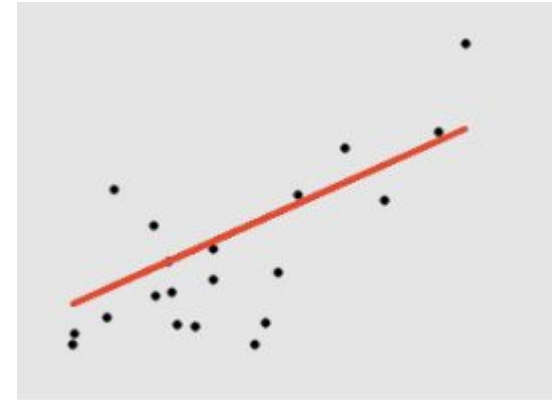
**Supuestos de Gauss Markov y Multicolinealidad**

2

**Regresiones Ridge y Lasso: extensión del modelo lineal.**

3

**Aprender a hacer validación cruzada para ajustar los hiper-parámetros de regularización**



# Supuestos Gauss Markov



## Multicolinealidad

- Los supuestos de Gauss Markov son una serie de condiciones sobre los datos que, si se cumplen, sugieren fuertemente que la regresión lineal es un buen approach.
- No son una condición necesaria para aplicar este modelo, pero si podemos probar que se cumplen permiten derivar varias propiedades de la estimación de mínimos cuadrados.
- De estas propiedades, la más importante son los intervalos de confianza para la predicción.

1. El modelo está correctamente especificado.
2. Debe ser lineal en los parámetros.
3. **El valor de la media condicional es cero.**
4. **Hay homocedasticidad**
5. No existe correlación entre las perturbaciones.
6. La covarianza entre  $u_i$  y  $x_i$  es cero.
7. El número de observaciones es mayor que el de parámetros.
8. No hay multicolinealidad perfecta.

$x$  y  $u$  son dos variables aleatorias. Veamos cómo se relacionan.  
Esta es la definición del problema de reg. simple

$$y = \beta_0 + \beta_1 x + u.$$

$$E(u) = 0.$$

La estimación que hacemos a partir de este modelo, requiere que además de tener media cero, los errores sean independientes de  $x$ .

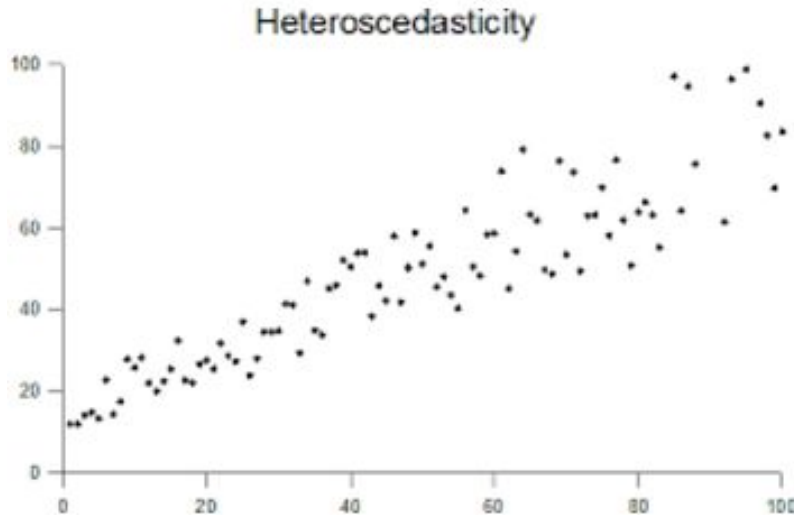
$$E(y|x) = \beta_0 + \beta_1 x.$$

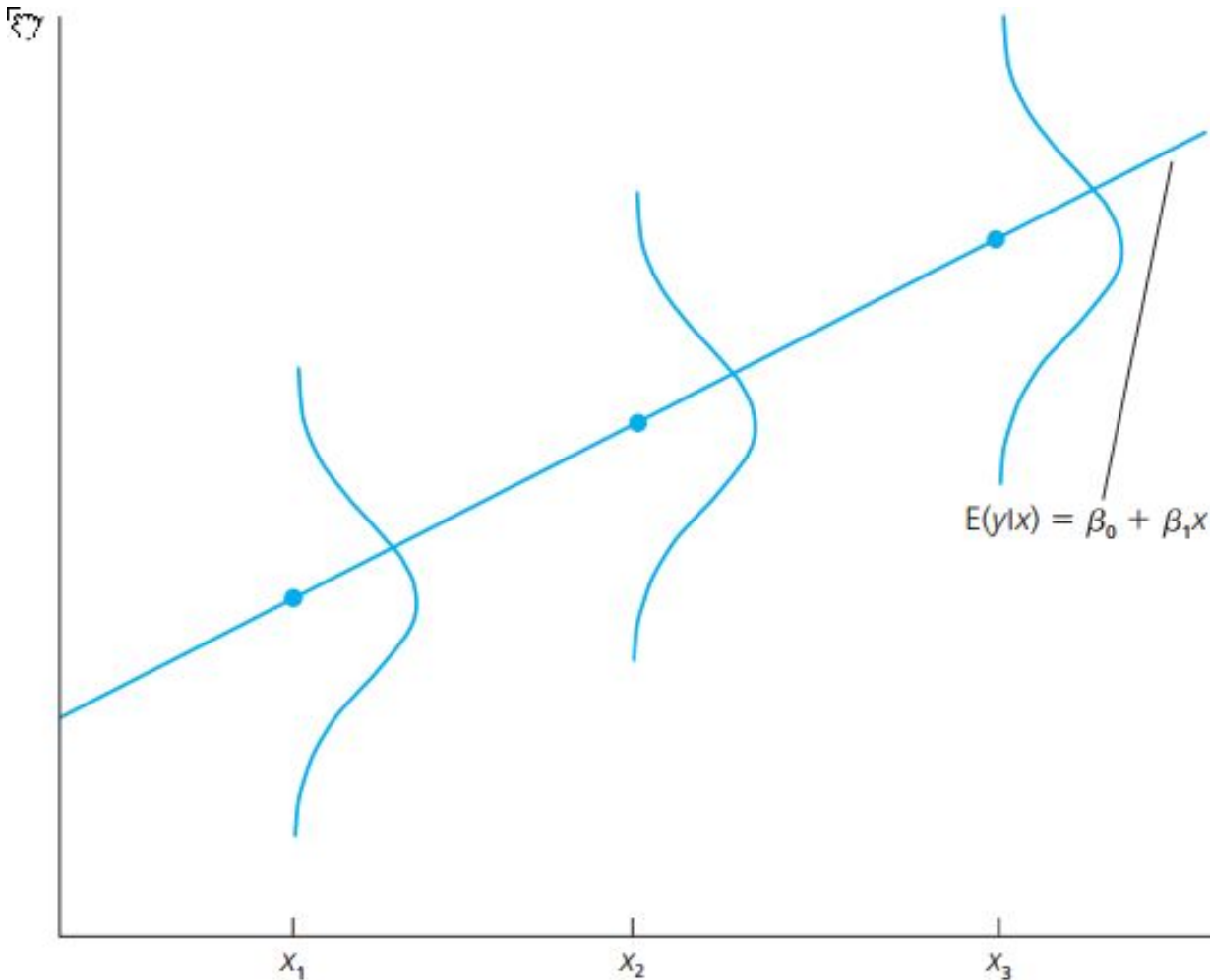
$$E(u|x) = E(u).$$

Además de que la esperanza de los errores sea cero para todo el espacio de las X, también es necesario que la varianza sea constante.

Cuando este supuesto no se cumple y la varianza de u crecer con x, decimos que hay heteroscedasticidad.

$$\text{Var}(u|x) = \sigma^2.$$





Homocedasticidad + Media Condicional igual a 0.

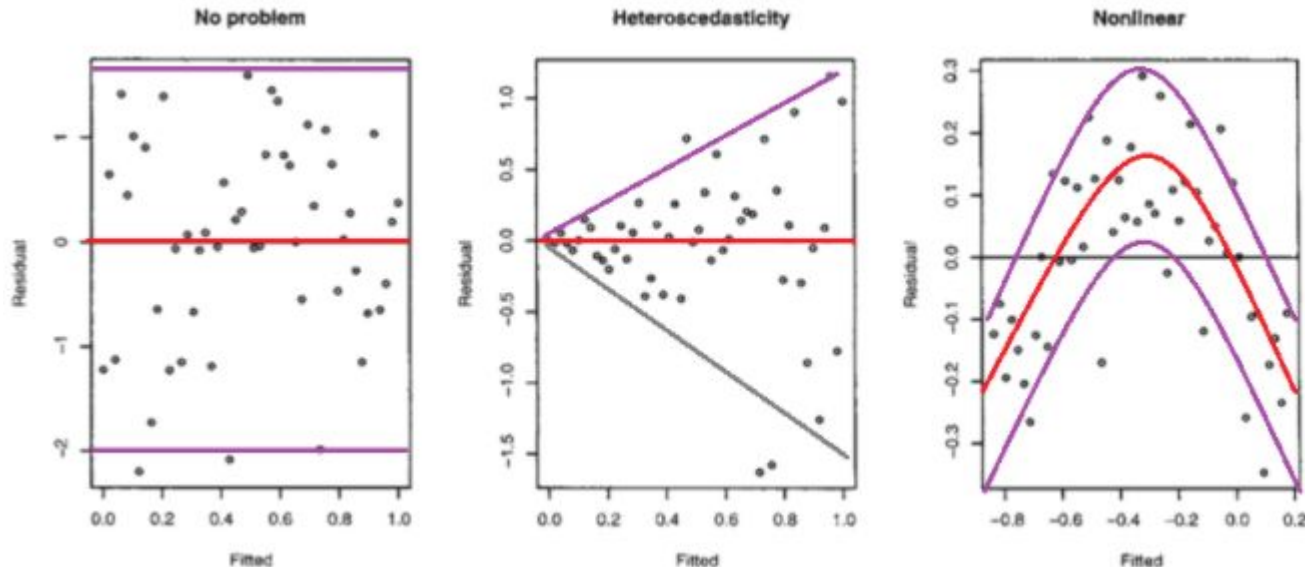
La consecuencia de estos dos supuestos, es que para cada valor de  $X$ , los errores se distribuyen con media cero y desvío estándar constante ( $\sigma$ )

$$E(y|x) = \beta_0 + \beta_1 x.$$

$$\text{Var}(y|x) = \sigma^2.$$



- Una buena forma de evaluar el cumplimiento de los supuestos sobre los residuos es graficar los mismos contra los valores predichos.
- En el primer caso, los residuos tienen la misma media y varianza para todos los valores de  $\hat{y}$ .
- En el segundo caso, la varianza de los residuos aumenta con  $\hat{y}$
- En el último caso la varianza parece constante, pero los residuos tienden a ser negativos para valores muy bajos o muy altos de  $\hat{y}$  y positivos para valores intermedios.



## “El costo de los estimadores insesgados.”

### ¿Qué significa la varianza de los estimadores?

Los estimadores de MCO que miden cómo afecta cada una de las X a la Y cambian de acuerdo a la muestra que se toma.

¿Cómo se pueden interpretar los estimadores si son demasiado distintos entre una muestra y otra?

El intervalo de confianza para  $\beta$  se hace demasiado grande.

$$\text{Var}(\hat{\beta}_j) = \frac{\sigma^2}{\text{SST}_j(1 - R_j^2)}$$

Partamos de la fórmula de la varianza de los estimadores:

$$\text{Var}(\hat{\beta}_j) = \frac{\sigma^2}{\text{SST}_j(1 - R_j^2)}$$

Varianza  
constante de los  
residuos

$$\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2$$

La varianza muestral de  $X_j$

Uno menos la  
variabilidad explicada  
de  $x_j$  contra todas las  
demás  $x$ .

### Componente 1: La varianza de los residuos.

Representa el ruido aleatorio en nuestro modelos.

Cuanto más alto el nivel de ruido en el modelo, más variables serán los estimadores.

Es una característica de la población, no se puede arreglar agrandando el tamaño de la muestra.

$$\text{Var}(\hat{\beta}_j) = \frac{\sigma^2}{\text{SST}_j(1 - R_j^2)}$$

### Componente 2: La varianza muestral de Xj.

Cuanto más amplios son los valores para la variable Xj representados en la muestra, más baja es la varianza de los estimadores.

Por ejemplo, si ajustamos una regresión lineal que calcula la velocidad al correr en función de la altura, la varianza de los estimadores será mayor si sólo tomamos en la muestra personas que miden entre 1,70 y 1,80.

$$\text{Var}(\hat{\beta}_j) = \frac{\sigma^2}{\text{SST}_j (1 - R_j^2)}$$

### Componente 3: $R^2_j$

Este componente se genera cuando dejamos afuera la clase y modelamos al regresor  $X_j$  como una regresión lineal que contiene a todos los demás regresores.

Esta regresión arroja un valor de  $R^2$  que indica en qué medida el regresor  $X_j$  se puede modelar como una combinación lineal de los demás regresores.

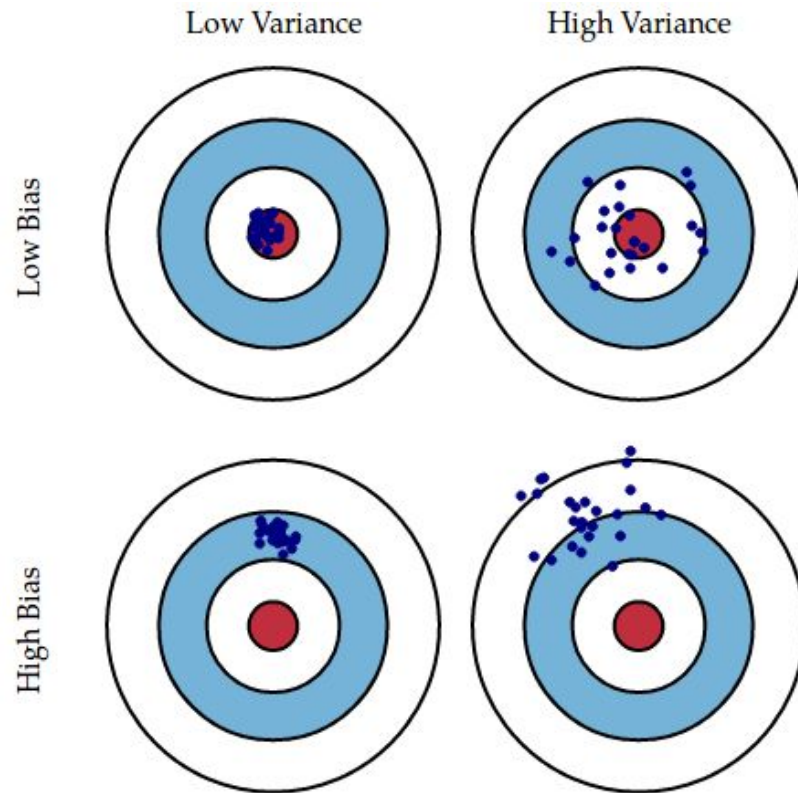
$$\text{Var}(\hat{\beta}_j) = \frac{\sigma^2}{\text{SST}_j(1 - R^2_j)'}$$

# Recordemos

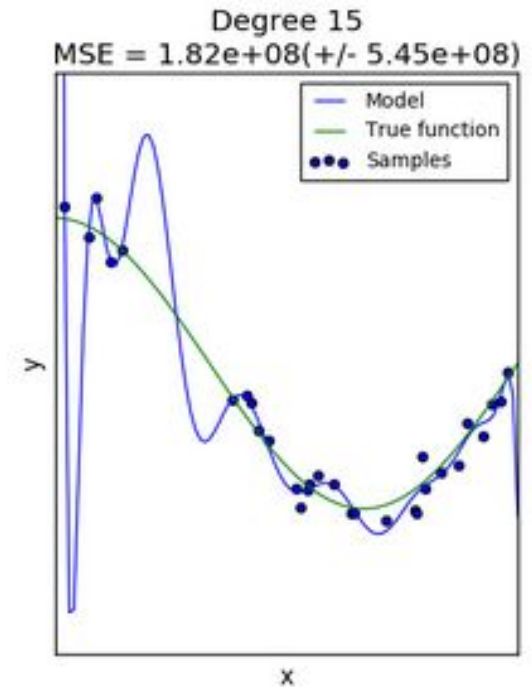
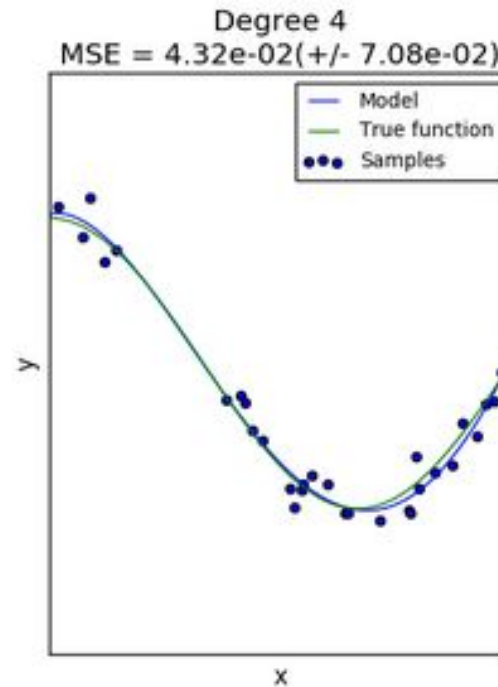
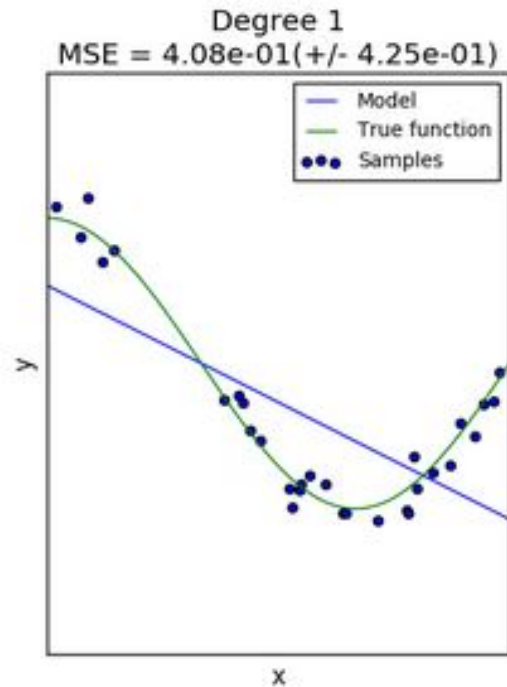


## Sesgo - Varianza

- Podemos caracterizar un modelo según el grado de ajuste a los datos
  - **Varianza alta** → ajuste exagerado ("sobreajuste" o *overfitting*)
  - **Sesgo alto** → ajuste insuficiente ("subajuste" o *underfitting*)







# Extendiendo la Regresión Lineal



- Recordemos el modelo lineal  $Y = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p$
- Nuestro objetivo es **extender la capacidad de este modelo**.

En los modelos de regresión por mínimos cuadrados, resolvemos el problema de la elección de los Betas **minimizando la suma de los residuos al cuadrado**

- A pesar de ser muy simple, el modelo de regresión lineal es
  - **fácil de interpretar**
  - **computacionalmente eficiente.**

¿Podríamos proponer una **nueva forma de ajuste** que mejore la performance del modelo?

Dos razones para buscar alternativas para extender la regresión lineal:

- **Accuracy:** ¿Recuerdan el trade-off entre sesgo y varianza? Cuando la cantidad de datos **n** se acerca a la cantidad de parámetros **p** que tenemos que ajustar, es difícil controlar la varianza de los estimadores y esto hace que caiga la performance en promedio.
- **Interpretabilidad:** Si logramos eliminar los features irrelevantes, la interpretación de los coeficientes del modelo es muy clara y directa. Vamos a ver algunas técnicas automáticas para la selección de features.

Se trata de definir qué variables deberían entrar en un modelo. En regresión lineal las técnicas se dividen en tres grupos.

- Selección de un subset: Buscar los predictores de todo el conjunto que creemos mejor se relacionan con la respuesta.
- Regularización: Ajustamos un modelo con todos los regresores, pero los coeficientes de algunos de ellos se ajustan a cero por las características de la técnica.
- Reducción de dimensiones: Proyectamos los  $p$  regresores disponibles originalmente en un espacio  $M$  de menor dimensión y utilizamos esa proyección como los nuevos regresores.

Se trata de definir qué variables deberían entrar en un modelo. En regresión lineal las técnicas se dividen en tres grupos.

- Selección de variables: Seleccionamos el subconjunto de variables que creemos mejor describe el conjunto que

**Hoy vamos a estudiar Regularización**

- Regularización: Ajustamos un modelo con todos los regresores, pero los coeficientes de algunos de ellos se ajustan a cero por las características de la técnica.
- Reducción de dimensiones: Proyectamos los  $p$  regresores disponibles originalmente en un espacio  $M$  de menor dimensión y utilizamos esa proyección como los nuevos regresores.

# Regularización





- Existe una técnica que nos ayuda a buscar el nivel de ajuste óptimo: la **REGULARIZACIÓN**
- Intuitivamente, podemos entender el concepto de regularización en términos del principio de parsimonia (Navaja de Occam).
- Este principio dice: *En igualdad de condiciones, la explicación más sencilla suele ser la más probable*
- En nuestro caso sería:
  - a misma capacidad de predicción, el modelo **más sencillo es mejor**

- En el modelo de regresión lineal la función de pérdida era la siguiente:

$$CF = \sum_i^N (\hat{y}_i - y_i)^2$$

- Cuando intentamos utilizar técnicas de regularización, se le agrega una “penalidad” a esa función de costo. La idea es hacer que a mayor complejidad del modelo, mayor sea la cantidad a minimizar.
- La forma general de la función de costo es la siguiente:

$$CF = \sum_i^N (\hat{y}_i - y_i)^2 + \alpha \theta_i$$

- Aquí theta es el vector que corresponde a los parámetros del modelo (en una regresión lineal, los betas) y alpha es un parámetro que “regula” la fuerza de la penalización: cuanto más grande es, mayor es la penalización.

Vamos a ver a continuación dos técnicas de regularización: **Regresión Ridge** y **Regresión Lasso**.

Estas técnicas proponen cambiar ligeramente el problema de optimización de mínimos cuadrados, para intentar “achicar” (*shrink*) el valor absoluto de la estimación de los Betas.

**¿Por qué esto mejoraría la estimación?** Vamos a ver de qué forma este método introduce un sesgo pero reduce la varianza.

- Recordemos la función que se minimiza en la estimación de mínimos cuadrados:

$$\text{RSS} = \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2.$$

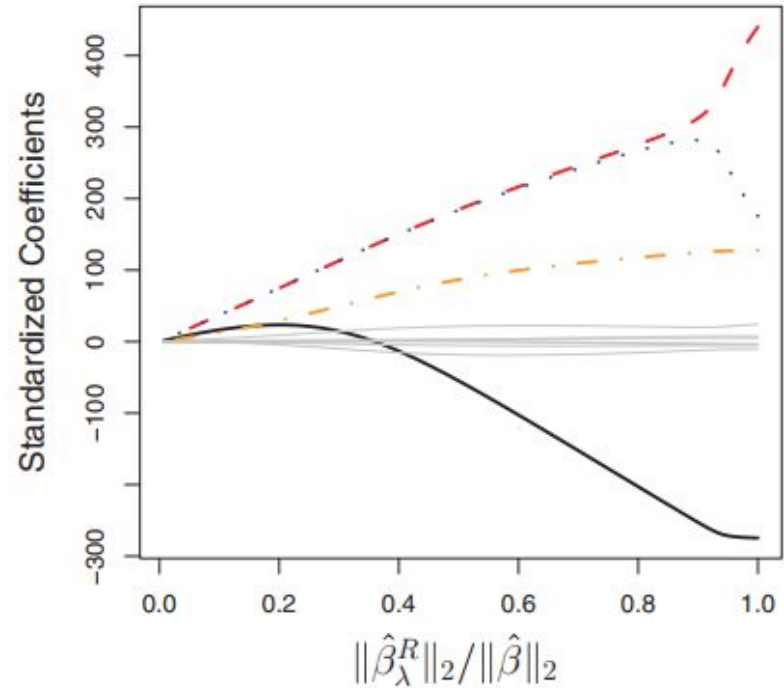
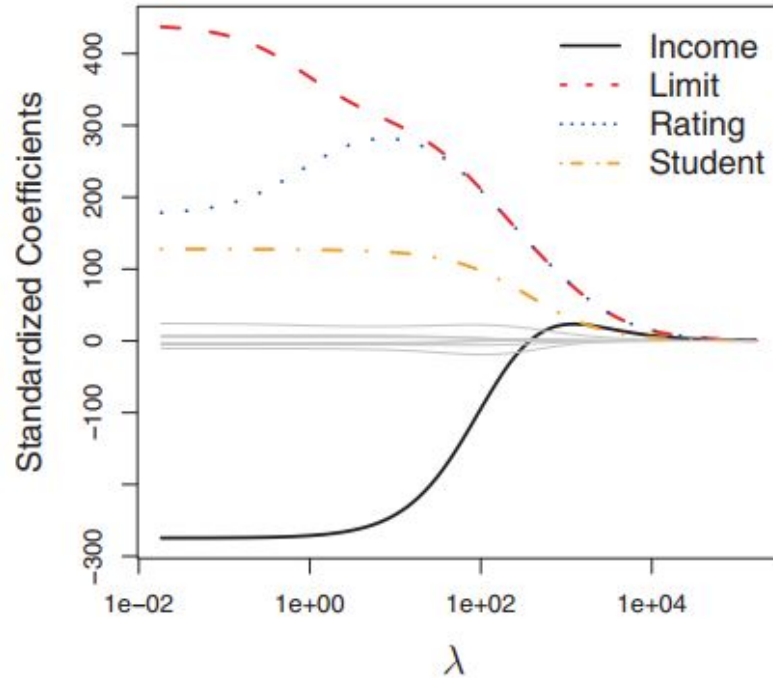
- Esta es, en cambio la función que se minimiza en Regresión Ridge:

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^p \beta_j^2.$$

La diferencia es que agregamos un término nuevo. En este término, un **hiperparámetro lambda** penaliza el valor de los coeficientes al cuadrado. Entonces, tengo que minimizar el cuadrado de los errores, intentando que ningún  $\beta_j^2$  sea demasiado grande

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

- Al igual que MCO, buscamos achicar el RSS.
- Sin embargo, existe un **término de penalización**, que es menor cuando los Betas se acercan a cero, por lo tanto tiene el efecto de achicar los mismos hacia cero (tanto si son negativos como positivos)
- El **hiperparámetro lambda**, maneja la ponderación de cada término.
- ¿Cuál es el mejor valor para lambda? ¿Cómo elegíamos el valor óptimo de un hiperparámetro?  
Como siempre, lo hacemos a través de **CROSS VALIDATION**



- En la figura de la izquierda, cada curva corresponde a los parámetros estimados de los coeficientes de la regresión Ridge, a medida que aumenta el  $\lambda$ .
- En el panel de la derecha, vemos la relación entre los coeficientes de la regresión Ridge y los de la regresión múltiple tradicional.

La métrica que calculamos para cada vector de coeficientes es lo que se denomina la “norma”, que nos da una idea del tamaño en valor absoluto de cada uno de los componentes, amplificando el efecto de los que son más grandes.

$$\|\beta\|_2 = \sqrt{\sum_{j=1}^p \beta_j^2}.$$

A medida que **aumenta  $\lambda$** , **los coeficientes de la regresión Ridge se hacen más chicos** con respecto a los de la regresión de Mínimos Cuadrados Ordinarios

- ¿Recuerdan que los coeficientes de la regresión tradicional no eran sensibles a la escala? La predicción del modelo no cambia si los valores están expresados en metros o en centímetros, en grados Fahrenheit o grados Celsius.

Las predicciones de la regresión lineal no se veían afectadas por un cambio de escala porque los coeficientes tenían la capacidad de dar cuenta de este dato.



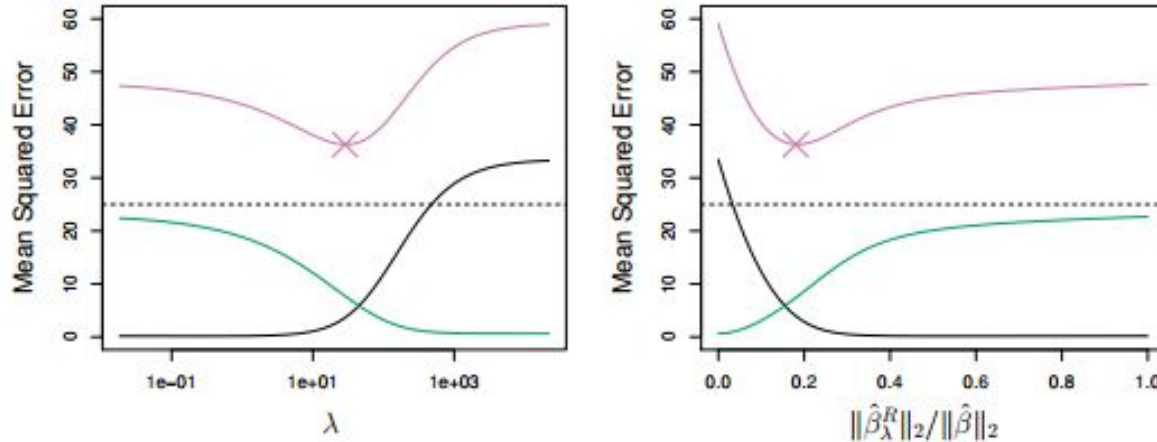
- En Regresión **Ridge**, en cambio, tanto la estimación de los coeficientes como la predicción son **sensibles a la escala**.
- Recordemos el problema de optimización que resuelve Ridge:

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^p \beta_j^2.$$

- Si una variable se encuentra en una escala que le da un valor absoluto mayor, esto **va a afectar el cálculo de la suma de cuadrados** del vector de coeficientes.
- Por esta razón **es importante estandarizar (dividir por el desvío estándar)** todos los regresores antes de ejecutar una regresión Ridge. Así ya no están en unidades físicas sino en unidades de su propio desvío estándar.

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}}$$

## El tradeoff entre Bias-Variance



Aquí se pueden ver  $n=50$  simulaciones,  $p=45$  predictores, todos con coeficientes no nulos.

El gráfico expresa el sesgo cuadrado (negro), varianza (verde) y el MSE del test (violeta), para una regresión ridge en los datos simulados, como una función  $\|\hat{\beta}_\lambda^R\|_2 / \|\hat{\beta}\|_2$

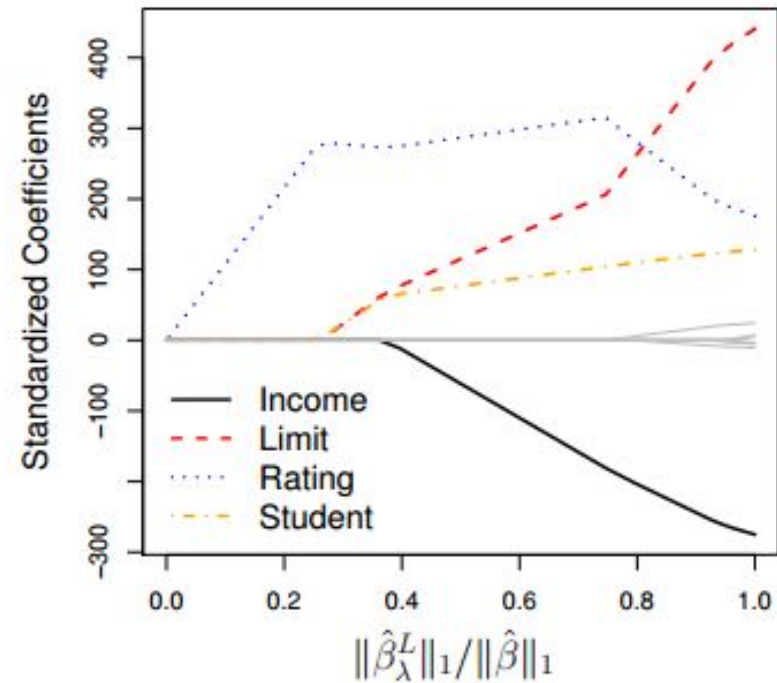
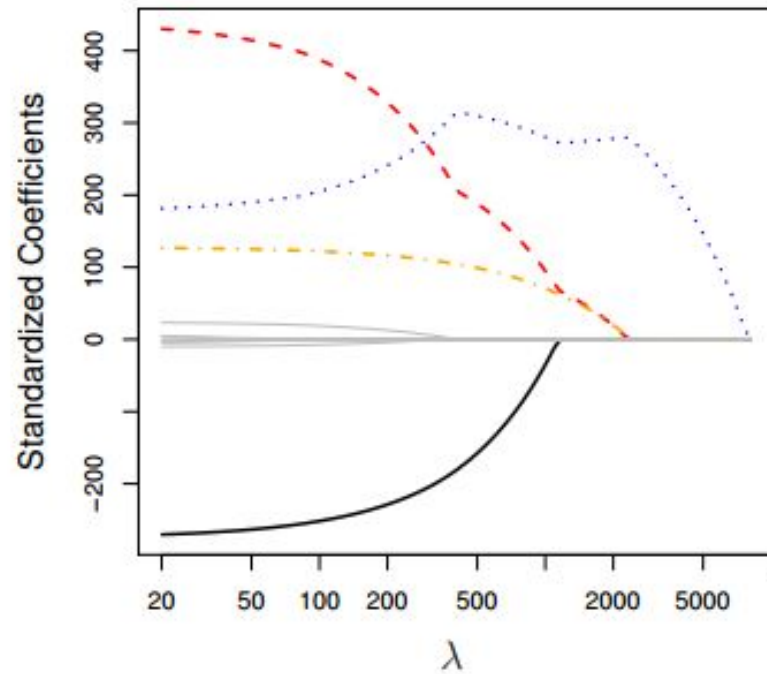
La línea punteada, indica el MSE mínimo

- La regresión Ridge tiene una clara desventaja: incluye todos los predictores  $p$  en el modelo final, a diferencia de aquellos modelos que eligen un conjunto de variables.
- La regresión Lasso es una alternativa relativamente nueva a Ridge, que corrige esta desventaja. Los coeficientes  $\hat{\beta}_\lambda^L$  minimizan el número de variables

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p |\beta_j| = \text{RSS} + \lambda \sum_{j=1}^p |\beta_j|.$$

- En lenguaje estadístico, Lasso utiliza penaliza con  $l_1$ , y no con  $l_2$ . La norma de  $l_1$  de un vector de coeficientes  $\beta$  está dada por  $\|\beta\|_1 = \sum |\beta_j|$ .

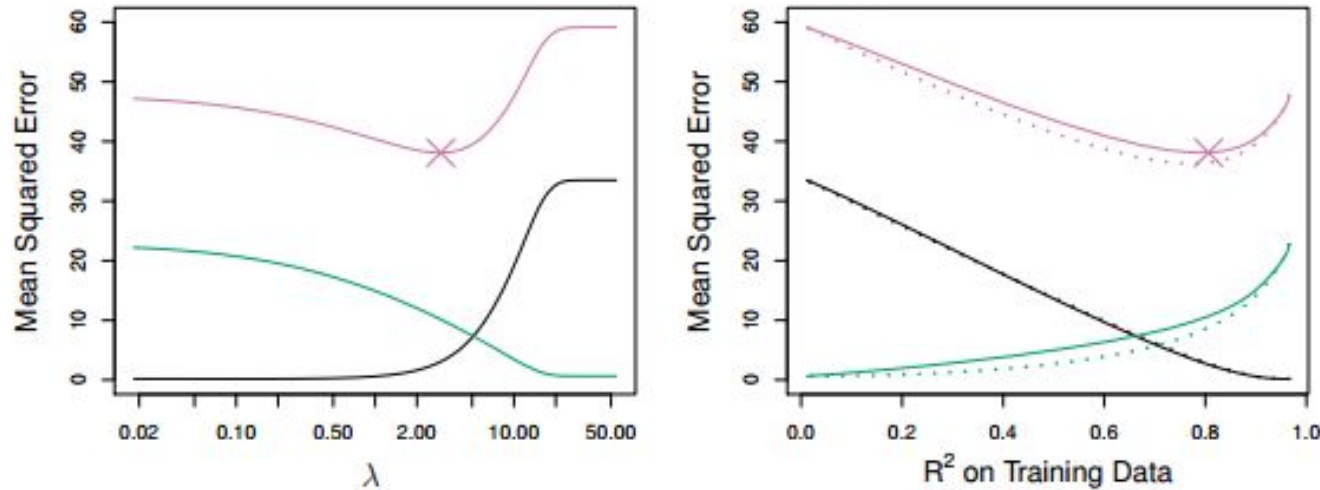
- Como en la regresión ridge, lasso “achica” los coeficiente estimados hacia el zero.
- Sin embargo, en el caso de Lasso, el  $l_1$  fuerza los coeficientes a valer exactamente cero, en el caso de que  $\lambda$  sea lo suficientemente grande.
- Por lo tanto, como en la selección de subsets, el lasso selecciona  $n$  variables
- Entonces, decimos que Lasso genera modelos dispersos, es decir, modelos con una selección de variables
- Al igual que en Ridge, la elección de un buen valor  $\lambda$  es crítico en Lasso; nuevamente, cross-validation es el método para su elección



- ¿Por qué Lasso, a diferencia de Ridge, resulta en coeficientes estimados exactamente igual a cero?
- Uno puede mostrar que la estimación de coeficientes de las regresión Lasso y Ridge resuelve estos problemas, respectivamente.

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^p |\beta_j| \leq s$$

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^p \beta_j^2 \leq s,$$



**Izquierda:** grafica el Sesgo cuadrado (negro), la varianza (verde) y el MSE del test (violeta)

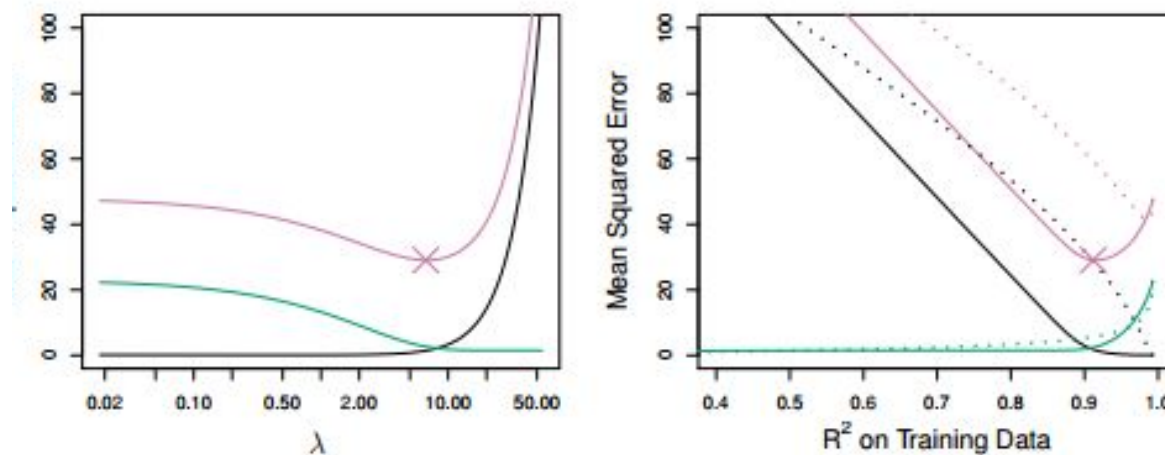
**Derecha:** Comparación del Sesgo cuadrado, la varianza y el MSE del test, entre Lasso (llena) y Ridge (punteada).

Las cruces indican el mínimo MSE

Estos datos se generaron haciendo que todos los coeficientes fueran diferentes a cero.

En este caso, los dos modelos tienden a performar prácticamente igual. Ridge tiene una menor varianza y por eso parece mejorar respecto a Lasso





**Izquierda:** gráfica el Sesgo cuadrado (negro), la varianza (verde) y el MSE del test (violeta). Los datos simulados, son similares a los anteriores, pero en este caso solo dos predictores están relacionados con la respuesta.

**Derecha:** Comparación del Sesgo cuadrado, la varianza y el MSE del test, entre Lasso (llena) y Ridge (punteada).

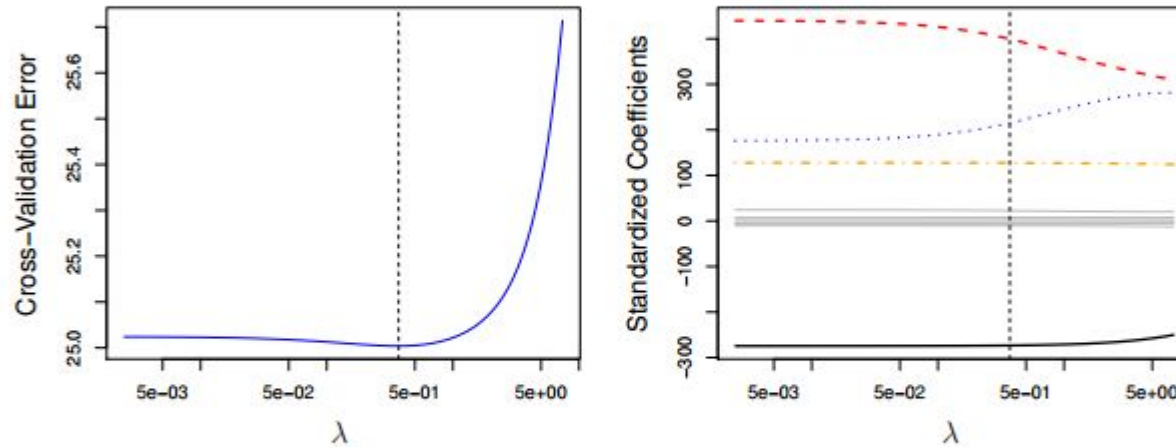
Las cruces indican el mínimo MSE

Estos datos se generaron haciendo que solamente dos coeficientes fueran diferentes a cero.

De esta forma, vemos cómo Lasso mejora claramente la performance con respecto a Ridge, tanto en lo referido a variancia como a MSE.

- Los últimos dos ejemplos demuestran que ningún método va a dominar por completo al otro.
- En general, se esperaría que Lasso performe mejor, cuando la cantidad de predictores asociados a la respuesta, es baja.
- Sin embargo, el número de predictores asociados a una respuesta, nunca es conocido a priori en un caso real.
- Una buena práctica para elegir entre uno o el otro es a través de cross-validation.

- Se necesita un método para poder ajustar el hiperparámetro  $\lambda$  o  $s$ , respectivamente
- **Cross-validation** es una manera simple de atacar este problema. Se elige un rango de valores que puede tomar el hiperparámetro, y luego se computan los errores que devuelve cross-validation, para cada caso.
- Se elige el hiperparámetro asociado al menor error computado.
- Finalmente, “re-fiteamos” el modelo con el hiperparámetro elegido.



Izquierda: El error que resulta de cross validation, al aplicar una regresión Ridge al dataset *credit*, para un rango de valores de  $\lambda$

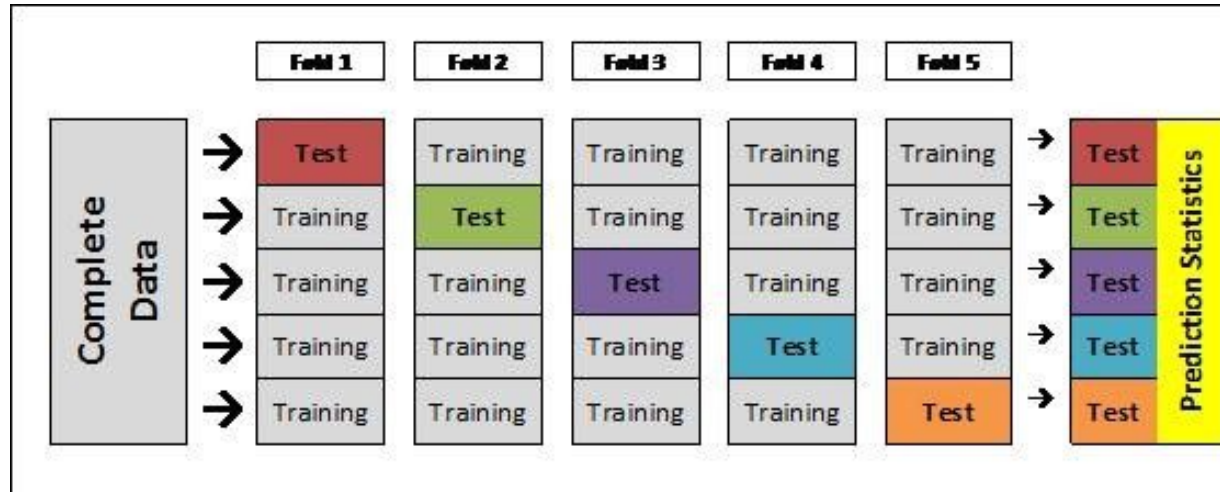
Derecha: El coeficiente estimado en función de  $\lambda$ . la línea vertical punteada, indica el  $\lambda$  elegido por cross-validation.

# Cross Validation



- Alpha es un parámetro desconocido (técnicamente es un hiperparámetro). Pero ¿cómo lo estimamos?
- Se hace a través de “cross-validation” (validación cruzada)
- 
- ¿Qué es?
  - Es un método general para evaluar un determinado modelo predictivo (sean regresiones lineales, logísticas, árboles de clasificación, etc.)
  - Es similar a la partición en test y training set... solamente, que repetida varias veces.
  - Muy útil cuando no hay suficientes datos para generar un test-set grande
- ¿Para qué sirve? (algunos usos habituales):
  - Estimar los “hiperparámetros” de un modelo (por ejemplo, alfa en el caso de las técnicas de regularización)
  - Generar estimaciones del error de generalización

- ¿Cómo funciona?
  - Empezamos dividiendo el dataset en k grupos (generalmente, 5 o 10 suele ser la medida convencional) del mismo tamaño.
  - En la primera iteración, el primer grupo generado pasa a ser un test test; el resto, pasa a ser el training set
    - Entrenamos un modelo sobre el training data
    - Hacemos las predicciones sobre el test set y calculamos el error sobre este test-set
  - Repetimos k veces, variando el test set en cada iteración.
  - Al final, promediamos los errores en cada una de las iteraciones





# Demo

## Regularización

# Práctica Guiada

Validación Cruzada con Ridge Regression

# Práctica Independiente

Análisis de datos inmobiliarios con regularización

# Conclusión

- La regularización nos ayuda a evitar el sobreajuste limitando la complejidad del modelo
- Matemáticamente lo logra penalizando la complejidad dentro de la función de costo
- Modelos con regularización suelen tener mayor poder de generalización
- Para determinar el valor de los hiper-parámetros usados para regularizar, usamos validación cruzada