Sesión 3. Métricas de evaluación. Generalización y complejidad de modelos

- ¿Cómo evaluar los modelos?
- Capacidad de generalización, complejidad de modelos y sobreajuste.
- Dilema bias-varianza.
- Regularización. Hiperparámetros.
- Selección de modelos y técnicas de validación.

¿Cómo evaluar los modelos?

- Métricas para regresión

Métrica	Interpretación
$r^2 = coeficiente\ de\ determinación = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$	Proporciona una medida de la proporción de la variación en la variable objetivo que es explicada por el modelo. Si, por ejemplo, r2 = 86,3 podríamos afirmar: "86,3% de la variabilidad en la variable respuesta es tomada en cuenta por la combinación lineal entre esta y las variables predictoras". Dependiendo del contexto, podría no obtenerse valores altos de R2. Por ejemplo, en estudios sociales normalmente se obtienen valores inferiores al 50%, por la dificultad de incluir en el modelo todas las variables que pueden incidir en las predicciones. Sin embargo, aun con valores bajos, si hay variables predictoras estadísticamente significativas, siempre serán de utilidad hacer el análisis.
$Error\ cuadrático\ medio = MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$	Para la interpretación, se suele utilizar la raíz cuadrada de este valor (RMSE) conocida como error típico o error estándar. Una ventaja es que el resultado se expresa en las unidades de medida de la variable objetivo. Es muy importante conocer la distribución de esta variable para poder determinar la calidad del modelo con base en el error. Por ejemplo, si el RMSE es de 5 ¿Podemos decir que el error es pequeño? ¿Es grande? Si la escala de la variable objetivo está entre [1, 10000] podríamos concluir que es pequeño, pero si la escala es de [1, 10] ¿También lo será? Seguramente no en este contexto. Al elevar al cuadrado el error de predicción se produce una sobreestimación cuando la diferencia entre el valor estimado y el valor real es grande.
$Error\ absoluto\ medio = MAE = \frac{1}{N}\sum_{i=1}^{N}\lvert\hat{y}_i - y_i\rvert$	Al igual que MSE, mide el error de predicción. Pero resulta más adecuada cuando en el conjunto de datos hay valores extremos.

- Métricas para clasificación

Matriz de confusión.

En un problema de clasificación binaria		Clase predicha	
		Clase = +	Clase = -
Clase actual	Clase = +	VP	FN
	Clase = -	FP	VN

FP = error tipo I
FN = error tipo II

N = 1150		Clase predicha		
		Clase = +	Clase = -	
Clase actual	Clase = +	VP = 210	FN = 70	= 280
	Clase = -	FP = 25	VN = 845	= 870

Para poder medir el rendimiento del modelo sobre los datos se definen métricas o medidas de rendimiento, las cuales dependen de la tarea.

Métrica de rendimiento o índice (medida de evaluación)

$$exactitud\ (accuracy) = \frac{VN + VP}{No.\,total\ de\ datos} = \frac{210 + 845}{1150} = 0.92$$

$$error\ = \frac{FN + FP}{No.\,total\ de\ datos} = \frac{70 + 25}{1125} = 0.08$$

Miden el rendimiento global

- Para determinar el rendimiento sobre cada una de las clases de manera independiente pueden utilizarse otras medidas:

N = 1150		Clase predicha		
		Clase = +	Clase = -	
Clase actual	Clase = +	VP = 210	FN = 70	= 280
	Clase = -	FP = 25	VN = 845	= 870

- Sensibilidad (*recall*):

$$Recall = \frac{VP}{VP + FN} = \frac{210}{280} = 0.75$$

  - Indica cuántos ejemplos de la clase positiva fueron clasificados correctamente.
  - También conocida como True Positive Rate (TPR) o Probabilidad de detección (Probability of detection).
  - Puede calcularse por cada clase.
  - Por ejemplo, para a clase negativa:

$$specificicidad = \frac{VN}{VN + FP} = \frac{845}{870} = 0.97$$

- Precisión (*precision*):

$$Precision = \frac{VP}{VP + FP} = \frac{210}{235} = 0.89$$

  - Indica, de los ejemplos clasificados como positivos, cuántos fueron clasificados correctamente.
  - Puede calcularse para cada clase.
  - Por ejemplo, para la clase negativa:

$$precisión = \frac{VN}{VN + FN} = \frac{845}{915} = 0.92$$

- F-score: *combina la precisión y el recall en un único número*

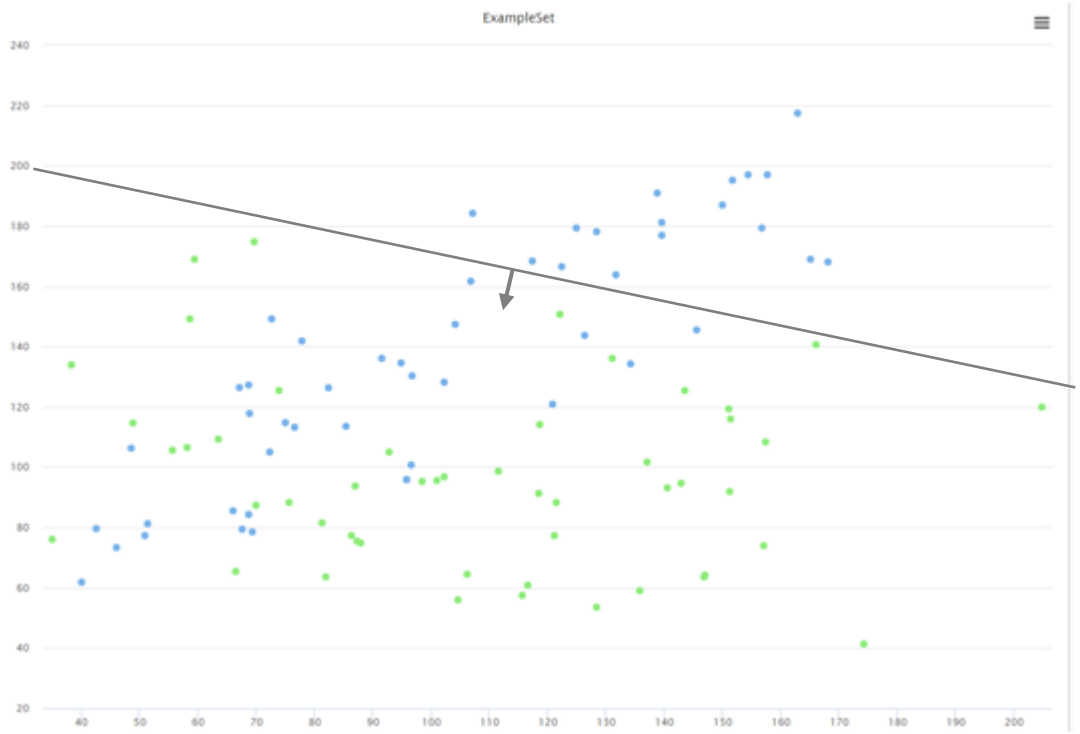
$$F_1 = \frac{2 \times recall \times precisión}{recall + precisión}$$

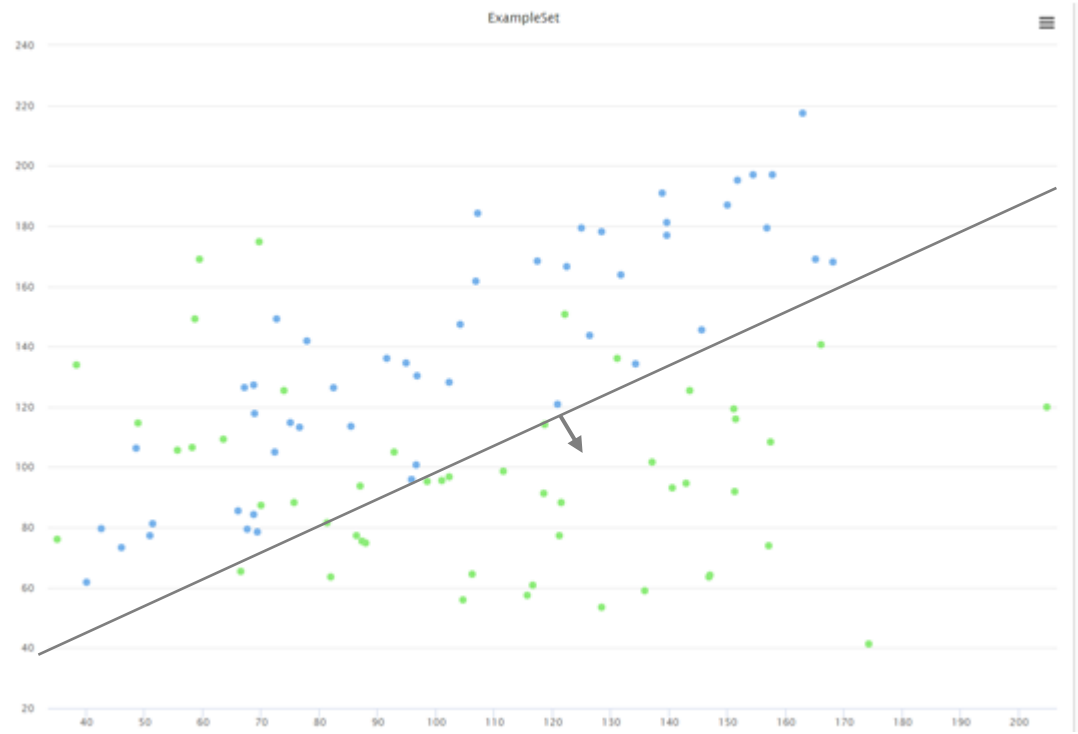
Recall	Precisión	F1-score
1	0	0
0	1	0
1	1	1
0.8	0.8	0.8
0.5	0.5	0.5
0.2	0.7	0.31
0.5	0.4	0.44

Tomar en cuenta…

- Existen tareas de aprendizaje para las cuales será necesaria una alta precisión, pero también hay tareas orientadas al *recall*.
- Importante: comprender las necesidades de la aplicación y seleccionar una métrica que se ajuste a los objetivos que se persiguen.

Ejemplo: dos posibles soluciones a un problema de clasificación binario.





Para la clase positiva (verde):

VP =50	**FN = 0**
FP = 33	**VN = 17**

$$Recall = \frac{VP}{FN + VP} = 1.00$$

$$Precision = \frac{50}{FP + VP} = \frac{50}{83} = 0.60$$

VP =32	**FN = 18**
FP = 0	**VN = 50**

$$Recall = \frac{VP}{FN + VP} = \frac{32}{50} = 0.64$$

$$Precision = \frac{32}{32} = 1.00$$

Generalización y complejidad de modelos

modelo	precio	kilometraje	motor	poder_maximo	asientos	combustible_CNG	combustible_Diesel	combustible_LPG	combustible_Petrol	propietario_First Owner	propietario_Fourth & Above Owner	propietario_Second Owner	propietario_Test Drive Car	propietario_Third Owner
2014	450000	145500	1248.0	74.0	5.0	0	1	0	0	1	0	0	0	0
2014	370000	120000	1498.0	103.52	5.0	0	1	0	0	0	0	1	0	0
2006	158000	140000	1497.0	78.0	5.0	0	0	0	1	0	0	0	0	1
2010	225000	127000	1396.0	90.0	5.0	0	1	0	0	1	0	0	0	0
2007	130000	120000	1298.0	88.2	5.0	0	0	0	1	1	0	0	0	0
2017	440000	45000	1197.0	81.86	5.0	0	0	0	1	1	0	0	0	0
2007	96000	175000	1061.0	57.5	5.0	0	0	1	0	1	0	0	0	0
2001	45000	5000	796.0	37.0	4.0	0	0	0	1	0	0	1	0	0
2011	350000	90000	1364.0	67.1	5.0	0	1	0	0	1	0	0	0	0
2013	200000	169000	1399.0	68.1	5.0	0	1	0	0	1	0	0	0	0
2014	500000	68000	1461.0	108.45	5.0	0	1	0	0	0	0	1	0	0

Algoritmo de regresión lineal

Attribute	Coefficient
modelo	41325.047
kilometraje	-1.375
motor	62.155
poder_maximo	15376.899
asientos	-83764.513
combustible_CNG	-1304207.715
combustible_Diesel	-1342347.092
combustible_LPG	-1137825.371
combustible_Petrol	-1405728.344
propietario_First Owner	-297427.476
propietario_Fourth & Above Owner	-308554.679
propietario_Second Owner	-372519.938
propietario_Test Drive Car	2037632.758
propietario_Third Owner	-350738.963
(Intercept)	-81840138.488

Evaluación del modelo

RMSE: 483646.11
MAE: 290222.03
R^2: 0.65

¿Todo bien?

¿Qué es la capacidad de generalización?

Habilidad de un modelo para predecir correctamente la salida (respuesta) para un dato nuevo

Modelo

Idealmente, quisiéramos que el modelo tenga un buen rendimiento sobre el conjunto de datos utilizado para su construcción

Datos

Muestra de la población

Sin embargo, lo que realmente es importante es que el modelo realice **predicciones válidas** para la **población** a partir de la cual se generó el conjunto de datos.

Así, el **objetivo** del aprendizaje supervisado es **construir modelos que generalicen bien**, es decir, que realicen predicciones válidas para datos nuevos.

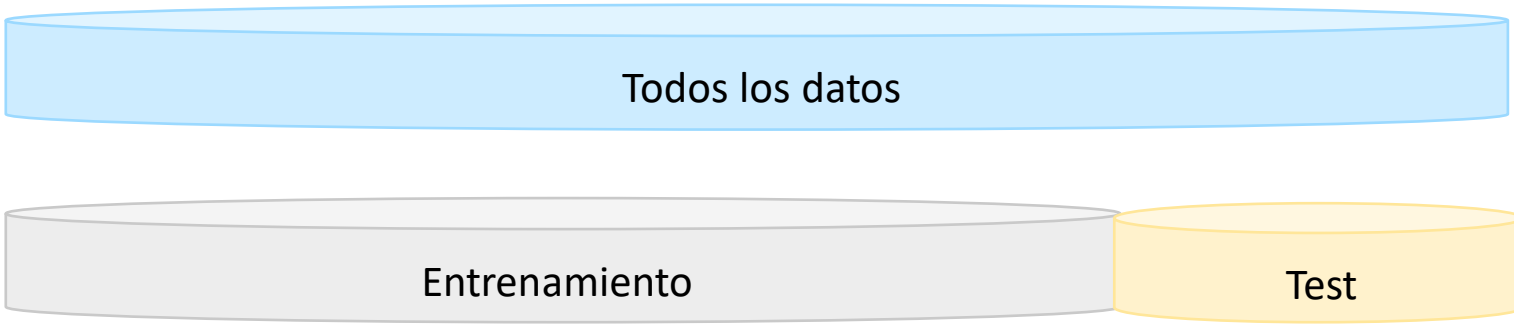
Es importante entonces conocer, antes de poner en producción un modelo, cuál es su capacidad de generalización.

¿Cómo determinar la capacidad de generalización de un modelo?

Una técnica ampliamente utilizada es dividir el conjunto de datos en dos: uno para el entrenamiento y otro conocido como test.

Esta división se debe realizar de manera aleatoria, a fin de garantizar que se mantienen las propiedades del conjunto de datos en las dos particiones. Entonces ¿Qué debemos hacer?

Una vez recolectados los datos, se realiza la división entre entrenamiento y test; por ejemplo, dejando un 20% para este último.

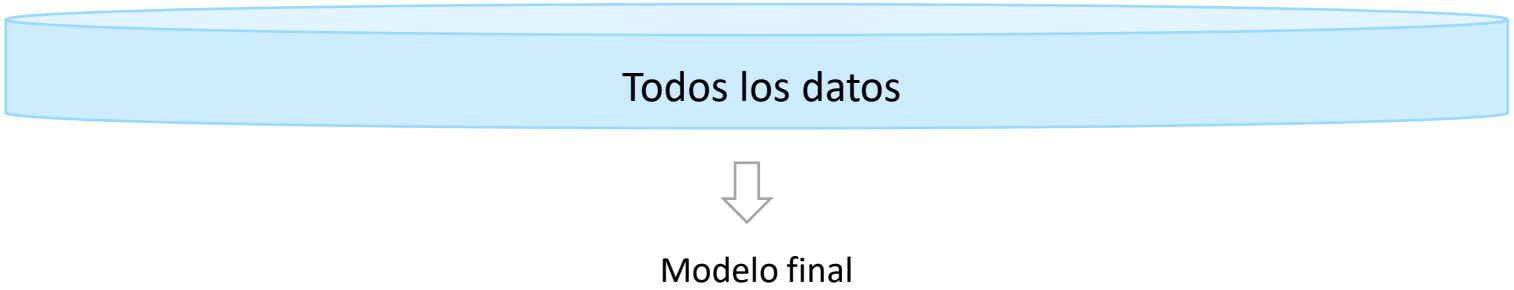


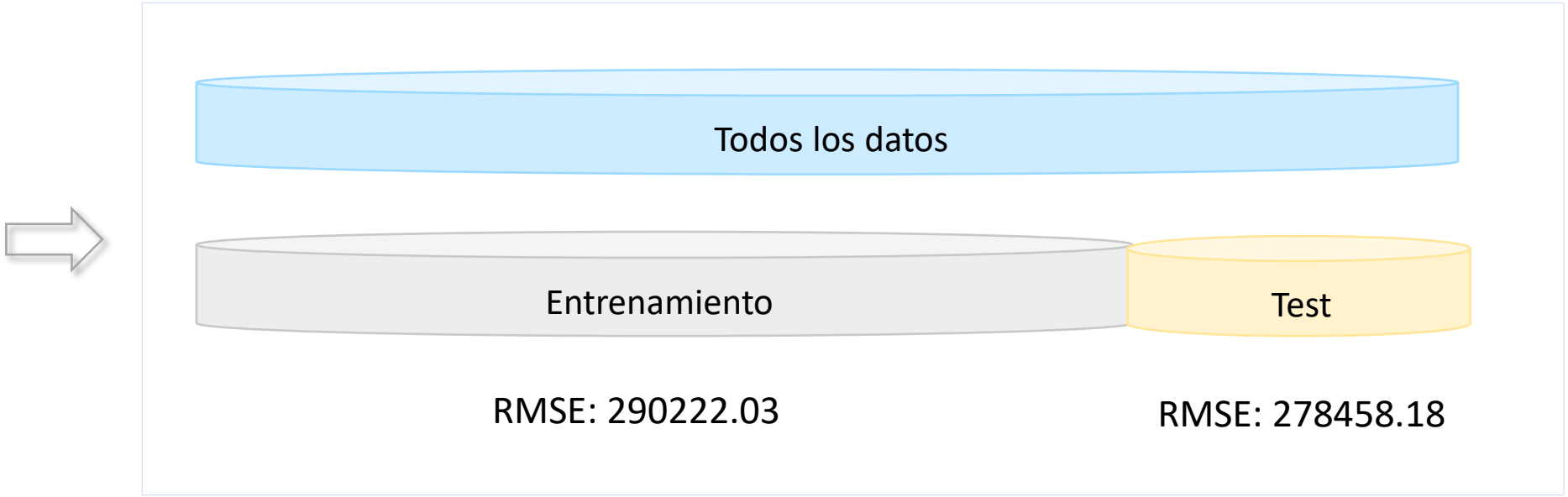
El conjunto de entrenamiento se utiliza para construir el modelo. Luego, utilizando las métricas seleccionadas se determina el rendimiento sobre este conjunto.

El modelo construido se aplica al conjunto test y se determinan los valores de las métricas de rendimiento seleccionadas sobre este conjunto. En el caso de la regresión, una de estas puede ser el error cuadrático medio. Así, tendríamos entonces dos medidas de error: una sobre entrenamiento y otra sobre test. Este último puede ser considerado entonces como un estimado del **error de generalización** sobre datos nuevos.

Es importante resaltar que el rendimiento sobre el conjunto de entrenamiento **no** es un estimado confiable de la capacidad de generalización de un modelo. Este estimado se obtiene al aplicar este sobre el conjunto de test.

Una vez obtenido este estimado, se puede construir el modelo final utilizando todos los datos.





¿Cómo interpretar los resultados?

Modelo de regresión lineal ¿Interpretable?

Algunas librerías: LIME, SHAP

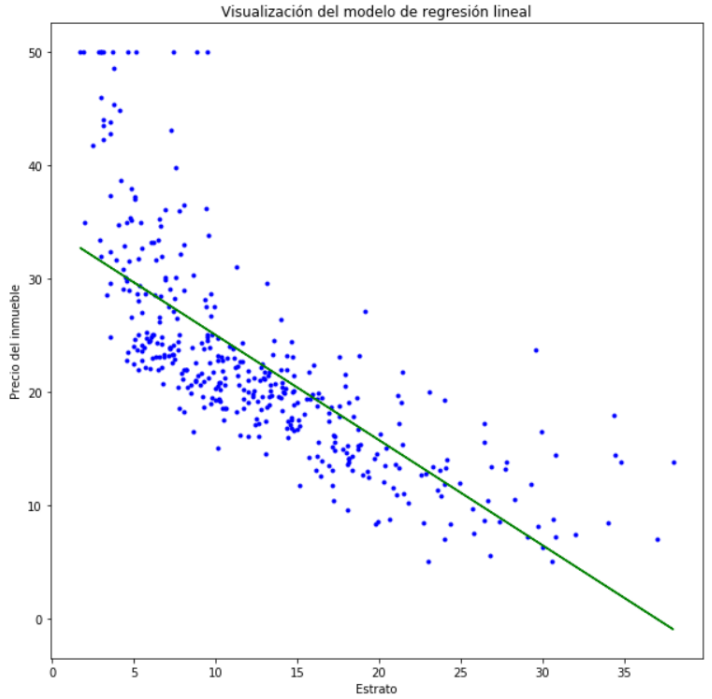
Regresión polinomial

Si la relación entre la variable objetivo y las variables independientes es no lineal ¿Qué pasaría si aplicamos nuestro algoritmo de regresión lineal?

$estrato$	$precio$
⋮	

Algoritmo de regresión lineal

$$precio_{estimado} = 34{,}24 - 0{,}93 \times estrato$$



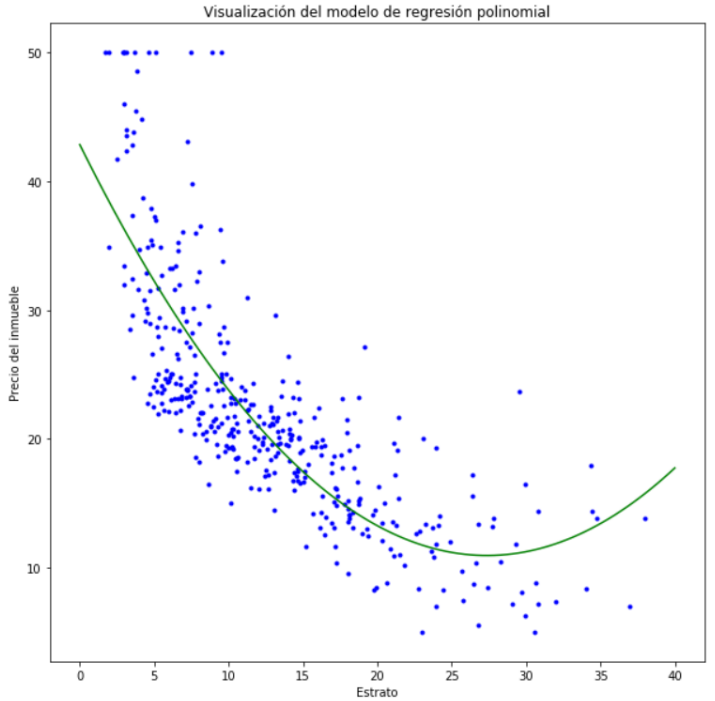
Una solución: aplicar transformaciones.

Preparación de datos

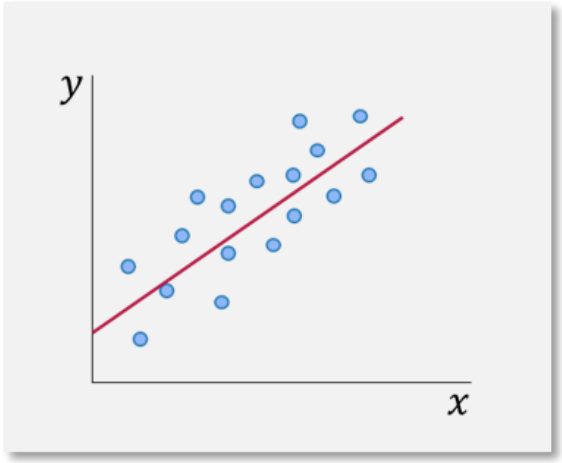
$estrato$	$estrato^2$	$precio$
⋮		

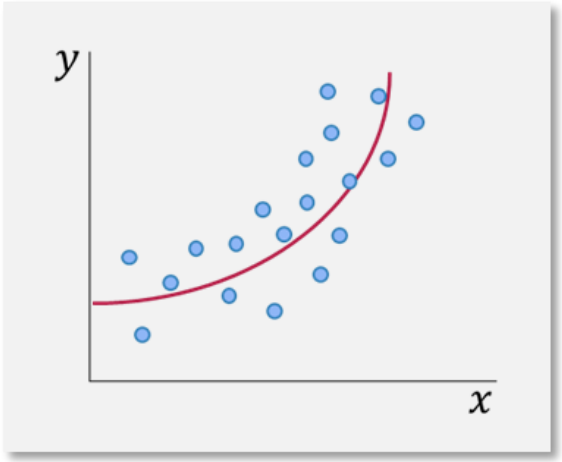
Algoritmo de regresión lineal

$$precio_{estimado} = 42{,}16 - 2{,}26 \times estrato + 0{,}04 \times estrato^2$$



- Este tipo de modelo de regresión se conoce como **Regresión polinomial**, el cual representa la relación entre la variable objetivo y las predictoras como un polinomio de orden p.
- El modelo sigue siendo lineal en sus parámetros. De esta forma, se puede utilizar un algoritmo sencillo, como es el caso de la regresión lineal, para aprender relaciones más complejas entre las variables.
- La construcción de nuevas características, que se añaden al conjunto de datos, ocurre en la fase de preparación de datos.
- Estas nuevas características, construidas a partir de la originales, se generan a través de una **transformación** polinomial.



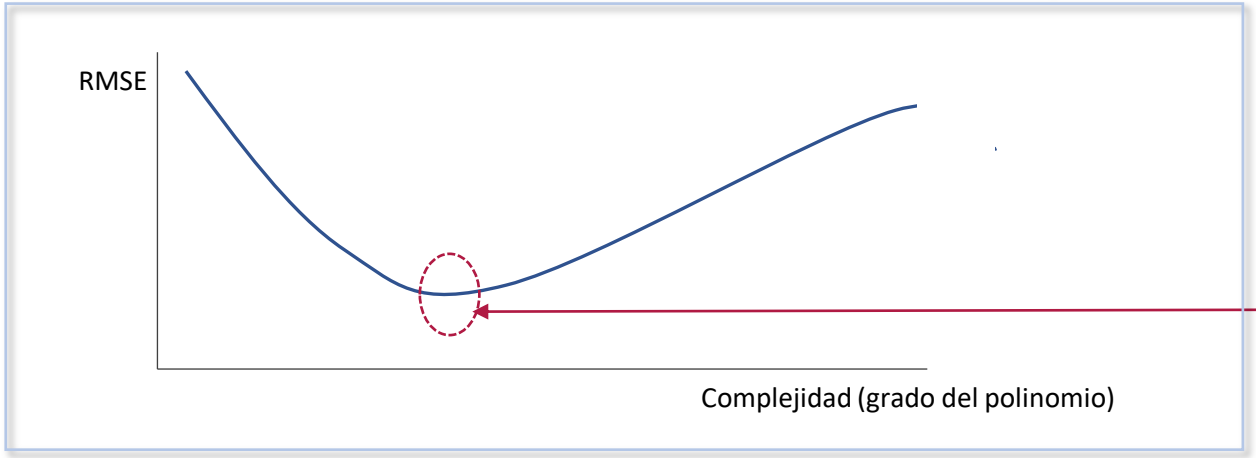


La inclusión de términos polinomiales al conjunto de datos permite que al algoritmo de regresión lineal expresar relaciones no lineales.

Complejidad de modelos y generalización

En nuestro problema de predicción de precios:

Modelo	Cantidad de variables	RMSE en test
Regresión lineal	14	278458.18
Regresión polinomial grado 2	105	210850.68
Regresión polinomial grado 3	560	218964.68
Regresión polinomial grado 4	2380	265285.78
Regresión polinomial grado 5	8568	330713.33
Regresión polinomial grado 6	27132	4615794.01

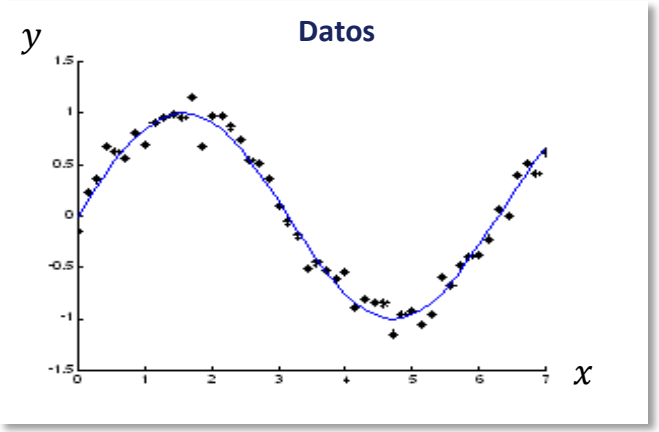


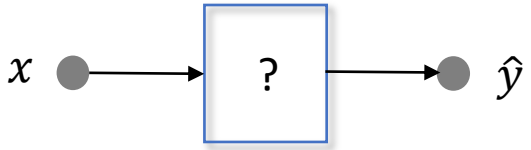
Una medida de la complejidad del modelo es el número de parámetros que lo componen y que son los que se ajustan a los datos.

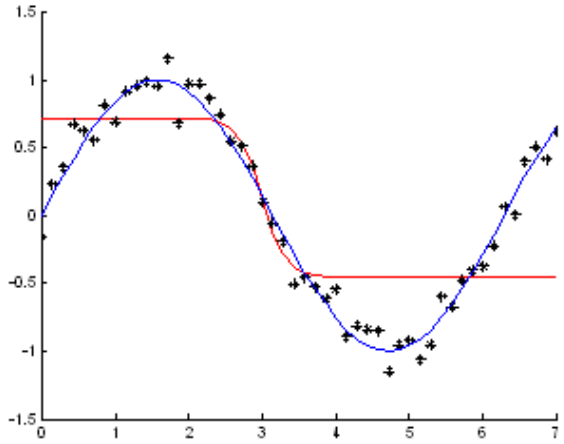
Cuando aplicamos la transformación polinomial se añaden nuevas variables al conjunto de datos en una proporción asociada al grado del polinomio. Así, a mayor grado, más coeficientes tendrá el modelo, es decir, su complejidad aumentará.

Es muy importante hallar el nivel de complejidad correcta que debería tener el modelo para nuestros datos.

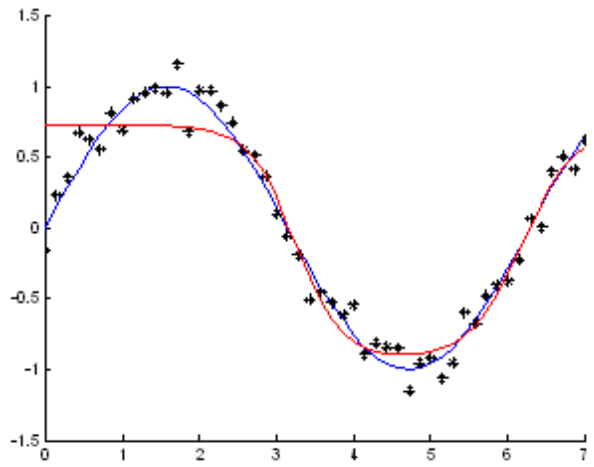
Otro ejemplo: red neuronal de una sola capa oculta.



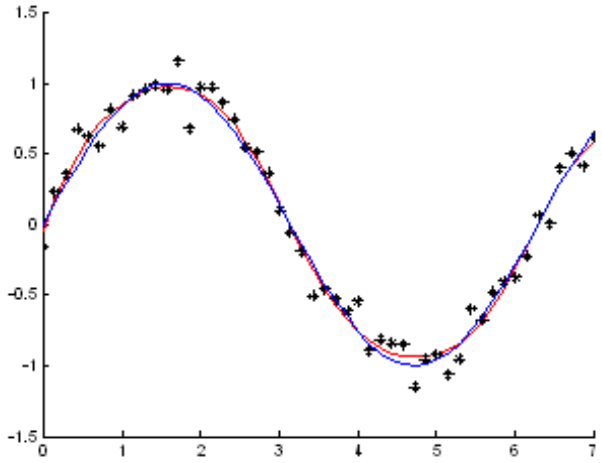




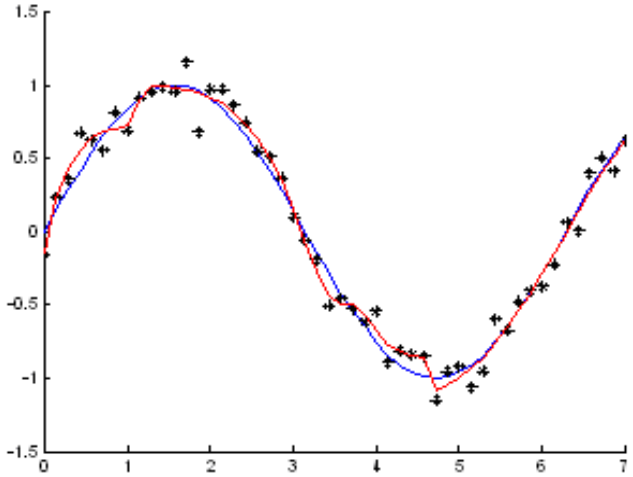
Una neurona en la capa oculta. MSE_{ent} = 0.167



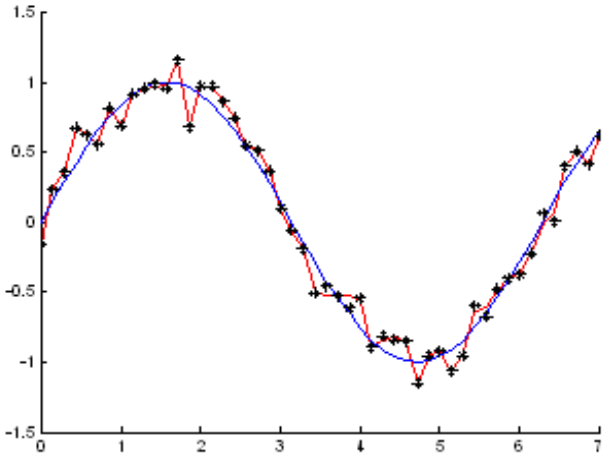
Dos neuronas en la capa oculta. MSE_{ent} = 0.040



Tres neuronas en la capa oculta. MSE_{ent} =0.009

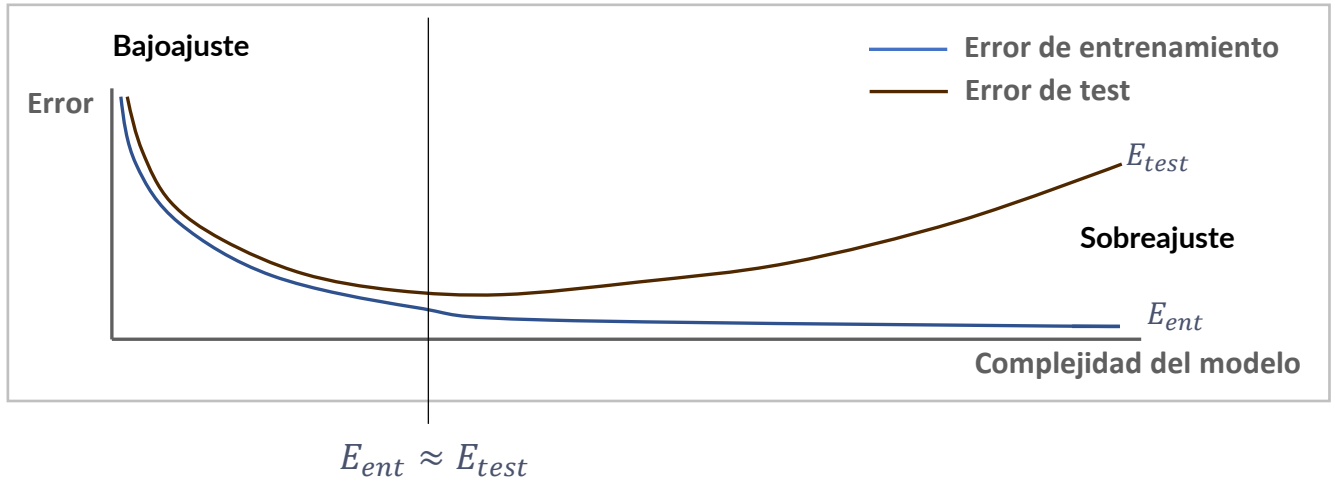


Seis neuronas en la capa oculta. MSE_{ent} = 0.007



25 neuronas en la capa oculta. MSE_{ent} = 0.0001

En general, se observa:



- A medida que aumenta la complejidad del modelo, el error de generalización desmejora.
- La mejor generalización se obtiene cuando el modelo representa los aspectos fundamentales de los datos, en vez de capturar detalles específicos de un conjunto de entrenamiento particular.

- La mayoría de los algoritmos de aprendizaje tienen parámetros que se pueden ajustar para controlar la complejidad de los modelos que generan. Estos parámetros se conocen como **hiperparámetros**.
- No confundir con los parámetros del modelo.
- Un aspecto esencial será entonces determinar los valores de los hiperparámetros que controlan la complejidad para obtener un modelo que generalice bien para nuevos datos.
- Para hacerlo se utilizan técnicas de validación o de selección de modelos.

En regresión polinomial

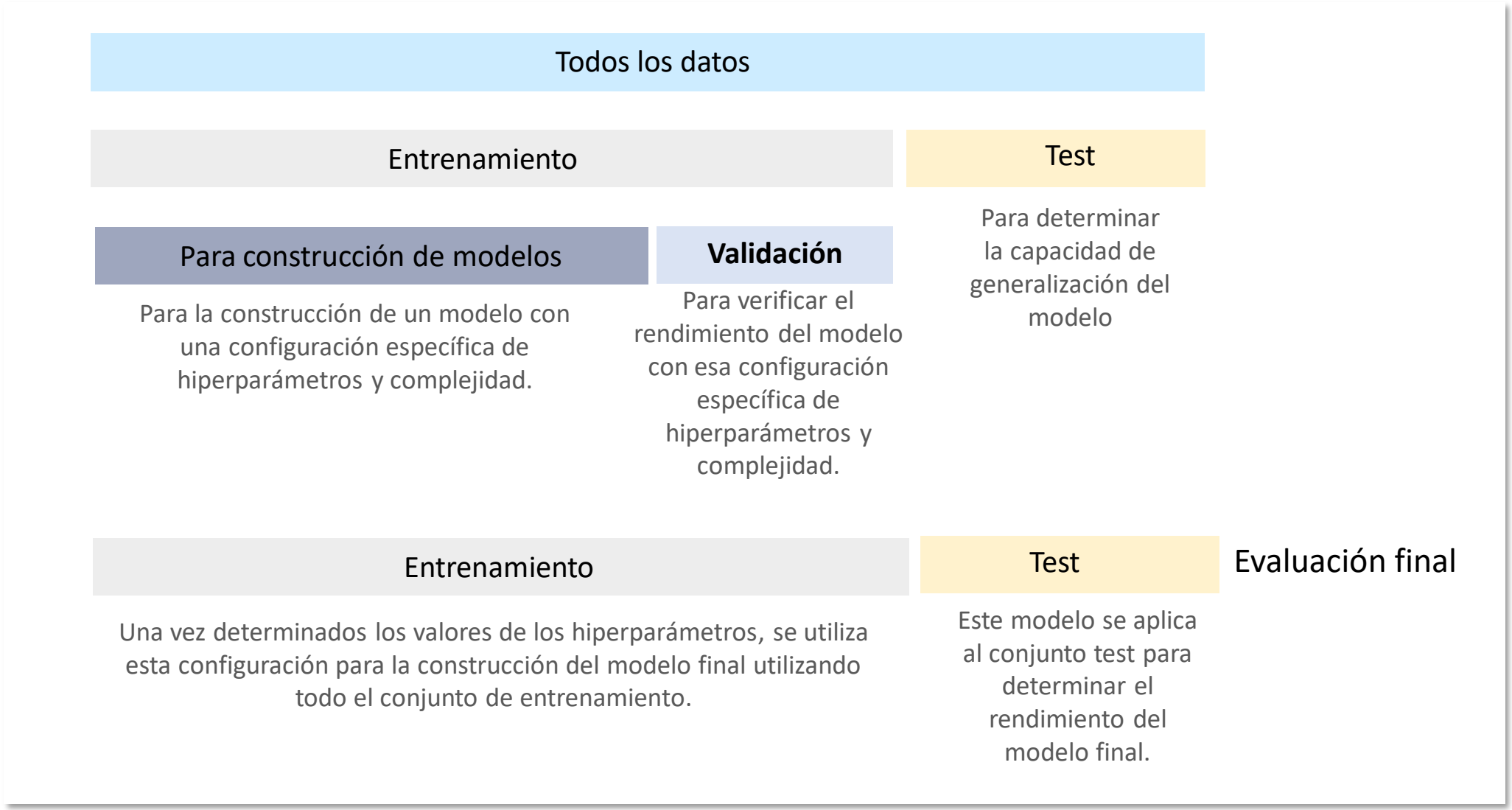
$$\hat{y} = w_0 + w_1 \times x + w_2 \times x^2 + \cdots + w_p \times x^p$$

Hiperparámetro

Parámetros del modelo

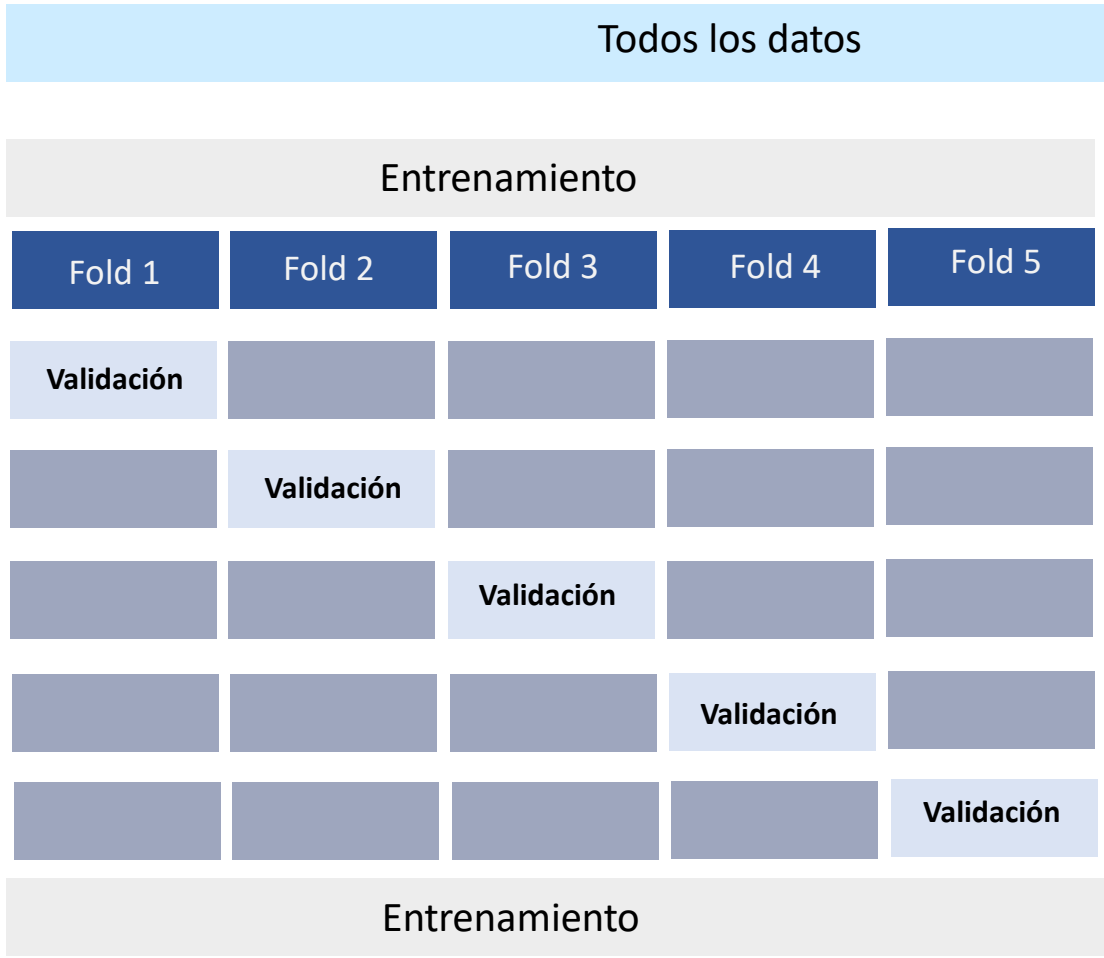
Técnicas de selección de modelos

- Validación hold-out:



Validación cruzada:

- K-fold cross validation.



Para determinar la capacidad de generalización del modelo

Se divide el conjunto de datos D en k particiones disjuntas (D_i, i=1.. k) de manera aleatoria.

Desde i = 1 hasta k

Construir un modelo con $D_{entrenamiento} - Fold_i$

Determinar el valor de la medida de evaluación me_i sobre $Fold_i$

Fin

Determinar el valor promedio: $ME_{final} = \frac{1}{k}\sum_{i=1}^{k} me_i$

Una vez determinados los valores de los hiperparámetros, se utiliza esta configuración para la construcción del modelo final utilizando todo el conjunto de entrenamiento.

Evaluación final

Este modelo se aplica al conjunto test para determinar el rendimiento del modelo final.

- Otro: Leave-One-Out cross-validation

Algunos problemas en el aprendizaje a partir de datos

Con respecto al conjunto de datos

- Insuficientes datos
- Datos no representativos del problema bajo estudio.
- Características irrelevantes.

Con respecto a la calidad y formato de los datos

- Baja calidad de datos
- Formato no adecuado para el algoritmo de aprendizaje.

Con respecto al modelado

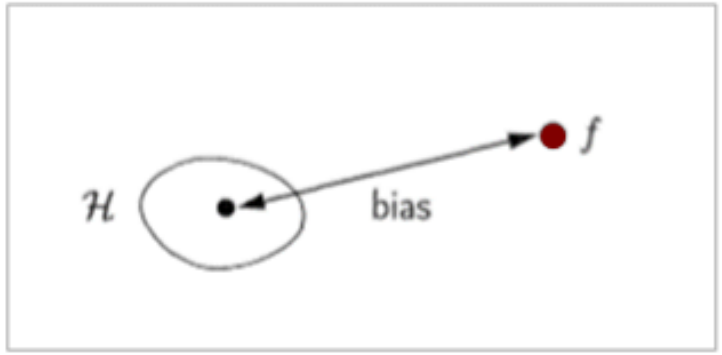
- Sobreajuste (*overfitting*) a los datos
- Bajo ajuste (*underfitting*) a los datos.

¿Cómo saber si hay sobreajuste?

Una forma: diagnóstico bias - varianza

- **El bias**. Este error se asocia a suposiciones incorrectas con respecto a la función target f, nuestra meta del aprendizaje. Por ejemplo, al utilizar un algoritmo de regresión lineal se parte de la suposición de que esta función sigue una relación lineal. Pero, ¿qué pasaría si la función f representa una relación no lineal como un polinomio? El resultado sería que la regresión lineal no aprenderá esta función, ya que su espacio de hipótesis no tiene la suficiente capacidad. Así, el bias representa la inhabilidad de un algoritmo para capturar o aprender la función target debido a que su espacio de hipótesis es muy simple o tiene una capacidad limitada.

$$bias = \left(\bar{g}(\boldsymbol{x}) - f(\boldsymbol{x})\right)^2$$

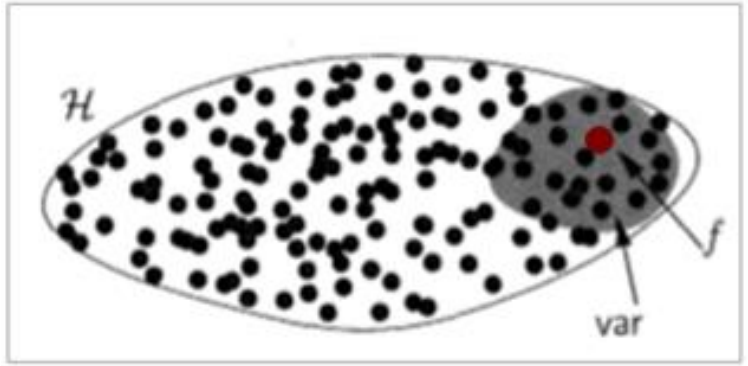


Espacio de hipótesis restringido.
Como solo hay una hipótesis, la función $\bar{g}$ como $g^{(D)}$serán las mismas para cualquier conjunto de datos. Por lo tanto, varianza = 0. El sesgo dependerá únicamente de qué tan bien esta hipótesis se aproxima al objetivo f, pero se espera un sesgo alto.

Yaser S. Abu-Mostafa, Malik Magdon-Ismail, Hsuan-Tien Lin. (2012). *Learning from Data*. A Short Course. AMLBook.com

▪ **La varianza**. Representa la situación contraria. El algoritmo utiliza un espacio de hipótesis con mucha capacidad y produce diferentes modelos sobre conjuntos de datos con pequeñas variaciones, pero generados a partir de la misma distribución fundamental. Cada modelo se ajusta muy bien a los datos, pero al aplicarlos a instancias nuevas predicen de manera diferente. Así, el algoritmo es sensible a la aleatoriedad en los datos y tiene mucha variabilidad en sus predicciones. Puede aprender muchos modelos (hipótesis) que concuerdan con f, pero no hay manera de saber cuál es la generaliza mejor.

$$varianza = \mathbb{E}_{(D)}\left[\left(g^{(D)}(x) - \bar{g}(x)\right)^2\right]$$



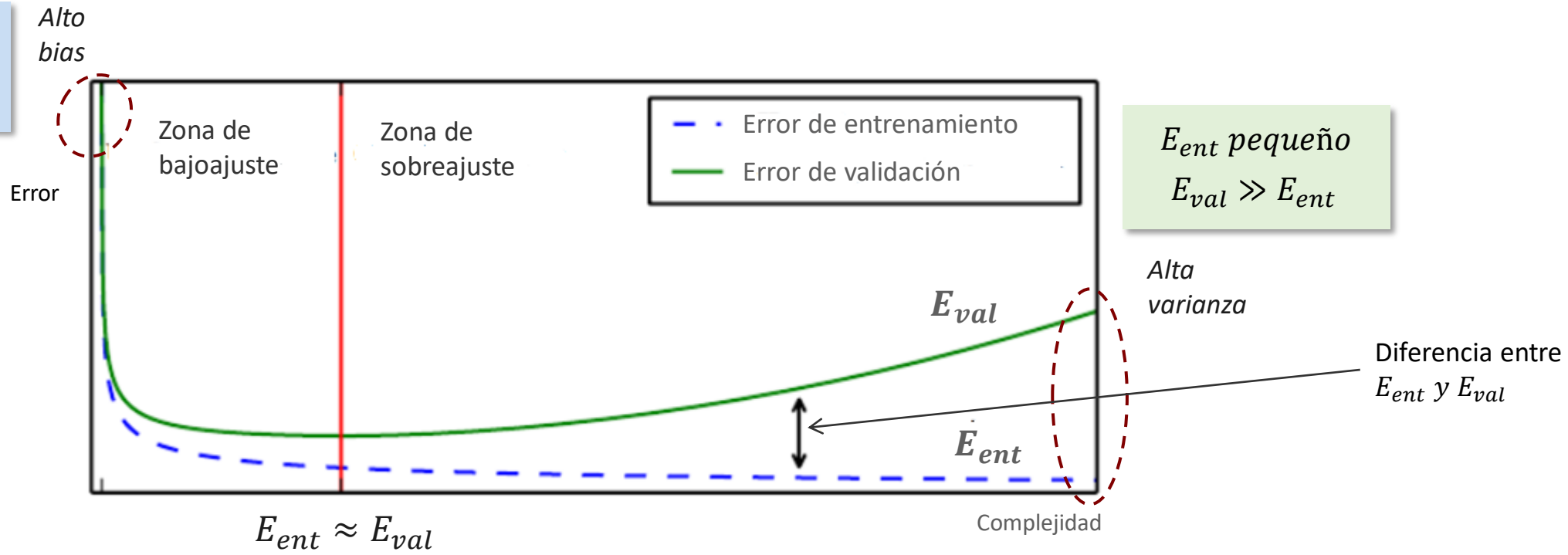
Espacio de hipótesis muy grande.
La función objetivo está en H. Diferentes conjuntos de datos conducirán a diferentes hipótesis que concuerdan con f . Por lo tanto, el $bias \approx 0$, ya que es probable que g esté cerca de f. La varianza será alta.

¿El dilema?

$$error_{generalización} = bias + varianza$$

☞ Los conceptos de bias y varianza nos proporcionan una explicación y guía que se pueden utilizar para determinar cuando hay un **sobreajuste**.

E_{ent} *grande*
$E_{ent} \approx E_{val}$

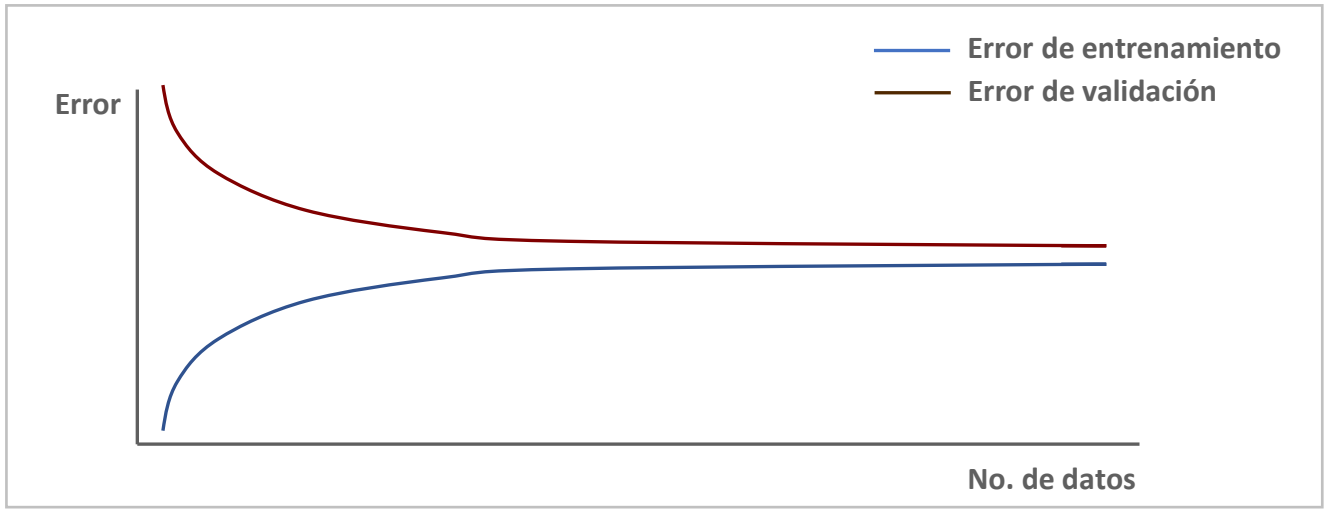


E_{ent} *pequeño*
$E_{val} \gg E_{ent}$

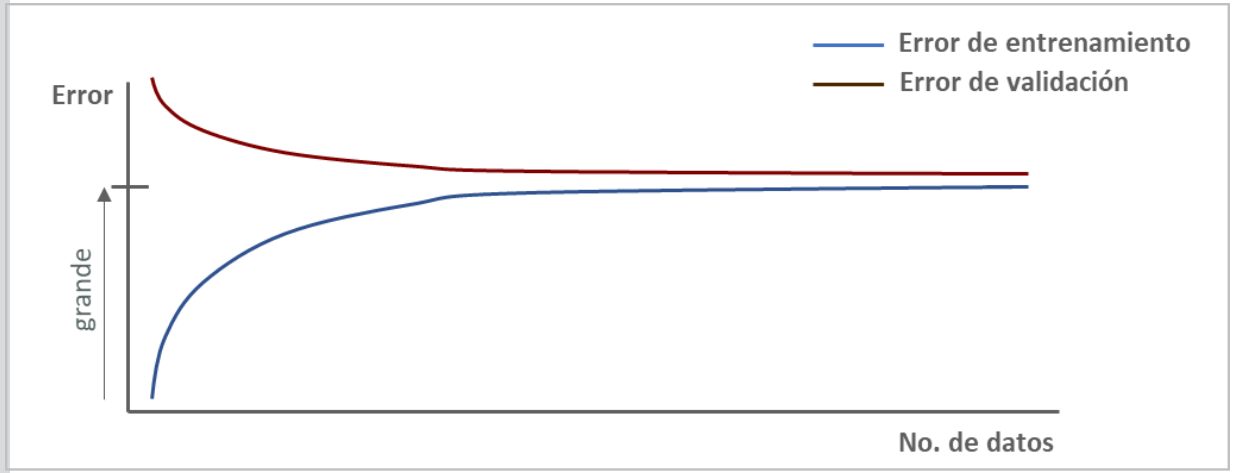
¿Y qué hacer?

Bajo ajuste	Sobre ajuste
Incluir más variables	**Más datos**
Modelos más complejos, características polinomiales,…	Modelos más simples, selección de variables,…
Regularización (cambiar el hiperparámetro de regularización)	Regularización (cambiar el hiperparámetro de regularización)

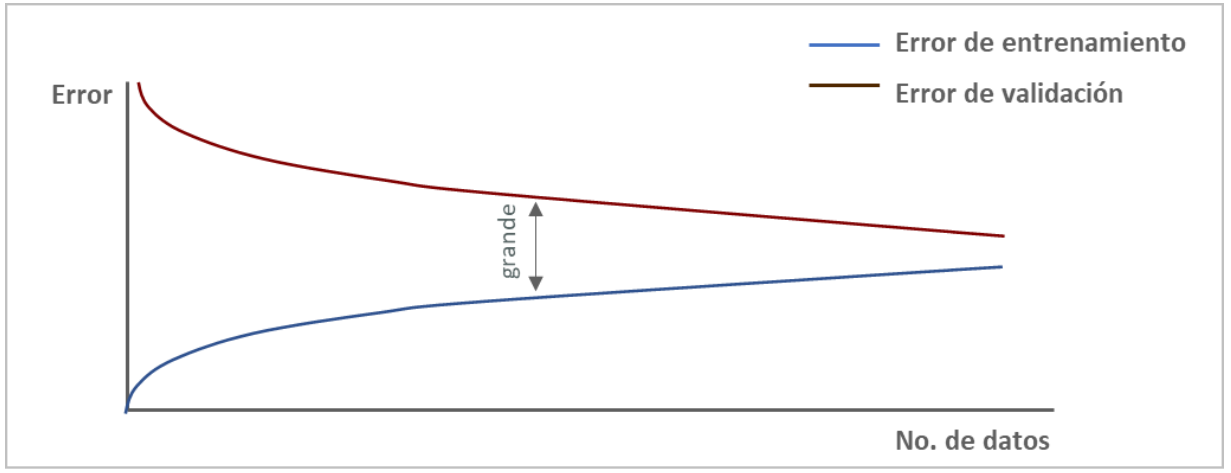
Curvas de aprendizaje



Alto bias



Alta varianza



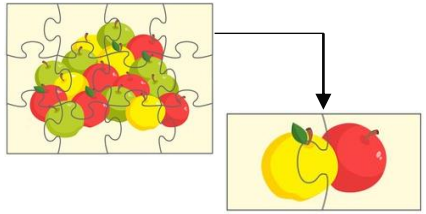
Regularización

Regularizar (o regular) significa "Determinar las reglas o normas a que debe ajustarse alguien o algo."[1]

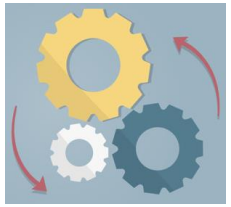
REGULARIZACIÓN

En machine learning se utiliza este mismo concepto para imponer restricciones sobre el algoritmo de aprendizaje para evitar el sobreajuste.

Estas restricciones permiten que utilicemos modelos complejos para construir soluciones simples, mejorando el rendimiento de generalización.



Hay varias maneras de implementar la regularización, una de ellas es modificando la función de costo del algoritmo para imponer restricciones sobre los valores de los parámetros.



[1]Diccionario de la lengua española (https://dle.rae.es/regular#DYICo7t)

¿Cómo podemos modificar la función de costo del algoritmo para introducir estas restricciones?

- Se añade a la función de costo un término que limita los valores que pueden tomar los parámetros del modelo
- Este nuevo término se conoce como término de regularización o de penalización.
- La idea es regular (obligar) a que los valores de los parámetros cumplan alguna condición.
- Por ejemplo, la función de costo para la regresión lineal es:

$$Función\ de\ costo = \boldsymbol{J}(\boldsymbol{W}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

- Con regularización utilizaríamos la siguiente función de costo

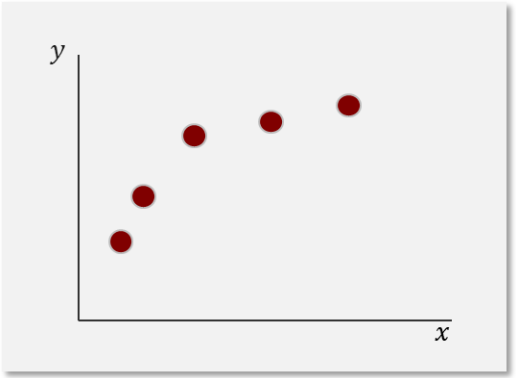
$$\boldsymbol{J}_{reg}(\boldsymbol{W}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 + penalty(\boldsymbol{W})$$

¿Cómo funciona?

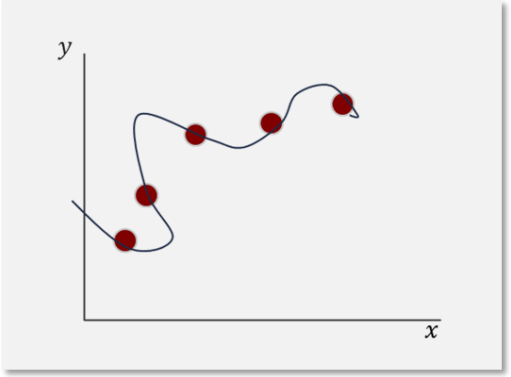
Supóngase que el modelo que se va a utilizar para ajustar los datos es:

$$\hat{y} = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4$$

Y utilicemos los datos:



Un modelo con esa complejidad produce la siguiente solución:



Modifiquemos la función de costo de esta forma:

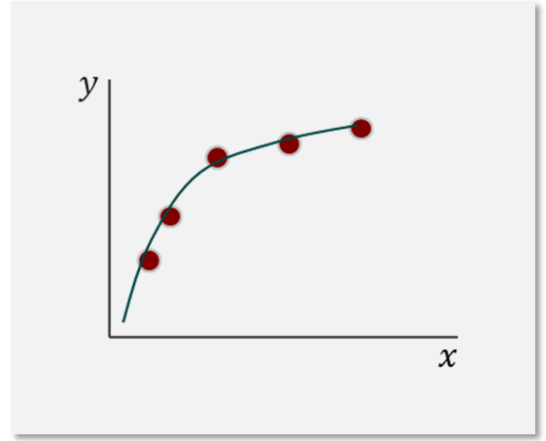
$$Función\ de\ costo) = J(W) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 + 400 \times \sum_{j=3}^{4} w_j$$

$$J(W) = \sum_{i=1}^{N}\left(\left(w_0 + w_1 x_i + w_x x_i^2 + w_3 x_i^3 + w_4 x_i^4\right) - y_i\right)^2 + 400w_3 + 400w_4$$

Al minimizar este funcional se obtiene:

$$w_3 \approx 0, w_4 \approx 0$$

$$\hat{y} = w_0 + w_1 x + w_2 x^2$$



- El algoritmo trata de minimizar esta función de costo para obtener los valores de los coeficientes.
- Al imponer restricciones sobre los valores que pueden asumir los parámetros w_3 y w_4 , se produce una solución (modelo) más simple que se ajusta suavemente a los datos.
- Así, **la regularización permite utilizar modelos complejos para generar soluciones más simples**, sin que haya sobreajuste.
- Se pueden observar que el término de penalización incluye un parámetro que controla cuánto se quiere penalizar o regularizar (en este caso utilizamos un valor de 400).
- Este parámetro es un **hiperparámetro** del algoritmo y se debe ajustar al conjunto de datos antes de construir el modelo, con técnicas de validación.
- Algunos tipos de regularización que se utilizan en la práctica:

Regularización norma L2: $término\ de\ penalización = \alpha \sum_{j=1}^{d} (w_j)^2$

Regularización norma L1: $término\ de\ penalización = \alpha \sum_{j=1}^{d} |w_j|$

α es un hiperparámetro de control de la complejidad

En regresión lineal:

- Regresión Ridge (utiliza norma L2)

$$\textbf{Función de costo} = \frac{1}{N}\sum_{i=1}^{N}(\widehat{y_i} - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}\left((w_0 + w_1 \times x_{i1} + w_2 \times x_{i2} + \cdots + w_d \times x_{id}) - y_i\right)^2 + \alpha\sum_{j=1}^{d}\left(w_j\right)^2$$

- Regresión Lasso (utiliza norma L1)

$$\textbf{Función de costo} = \frac{1}{N}\sum_{i=1}^{N}(\widehat{y_i} - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}\left((w_0 + w_1 \times x_{i1} + w_2 \times x_{i2} + \cdots + w_d \times x_{id}) - y_i\right)^2 + \alpha\sum_{j=1}^{d}\left|w_j\right|$$

¿Cuál es el efecto de la regularización en la regresión lineal?

- Funciones más simples, que generalizan mejor.
- Si L2, los valores de los coeficientes son muy pequeños.
- Si L1, algunos coeficientes tendrán el valor de cero. Puede ser utilizado para selección de variables.
- Si solución analítica puede resolver el problema de la colinealidad.
- Si solución por descenso por el gradiente, mejor rendimiento (número de iteraciones se puede reducir).

Por ejemplo, utilizando regresión Lasso en nuestro problema de predicción de precios de vehículos:

	coeficientes	variables
0	3.506291e+04	modelo
1	-1.105315e+00	kilometraje
2	5.101320e+01	motor
3	1.004054e+04	poder_maximo
4	-1.216578e+04	asientos
5	-0.000000e+00	combustible_CNG
6	1.634769e+04	combustible_Diesel
7	1.038633e+05	combustible_LPG
8	-5.318462e+04	combustible_Petrol
9	5.644763e+04	propietario_First Owner
10	4.466653e+04	propietario_Fourth & Above Owner
11	-6.308077e+03	propietario_Second Owner
12	2.512492e+06	propietario_Test Drive Car
13	-0.000000e+00	propietario_Third Owner

Próxima sesión

- La ingeniería de características.
- Preparación de imágenes.
- Máquinas de vectores de soporte.