

Modelos de aprendizaje supervisado

Modelos y sus objetivos

Un modelo es una reconstrucción simplificada de un proceso.

En un modelo de datos, siempre tenemos al menos

- una variable resultante Y también llamada variable **dependiente**
- una o más variables predictoras (X_1, X_2, \dots, X_p) también llamadas **explicativas o predictores**
- una relación entre ellas dada por f

$$Y = f(X) + \epsilon \quad \text{donde } \epsilon \text{ es un término de error}$$

Encontrar f puede tener dos objetivos:

Predecir un valor de la variable dependiente \hat{Y} dados valores conocidos de las variables predictoras.

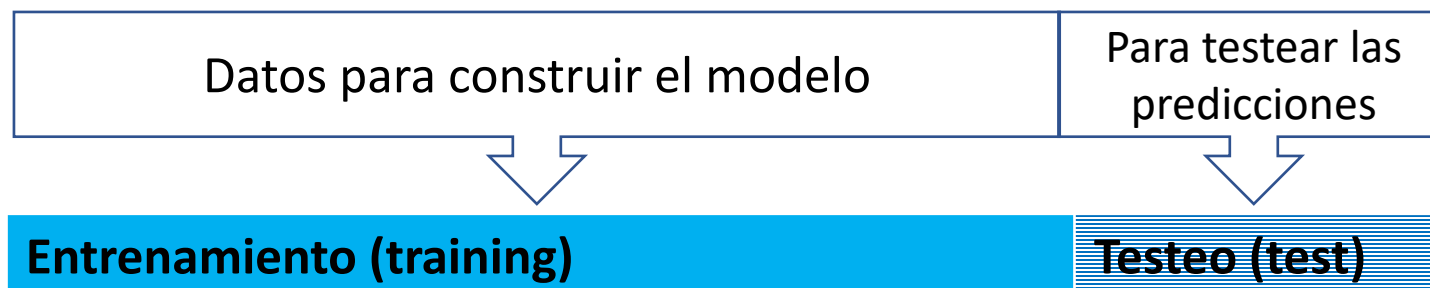
$$\hat{Y} = \hat{f}(X)$$

El modelo cumple su cometido si las predicciones son acertadas minimizando la diferencia entre Y e \hat{Y} . No importa el efecto de cada variable en la respuesta por lo que \hat{f} es tratada como una caja negra (no es necesario conocer su forma funcional) .

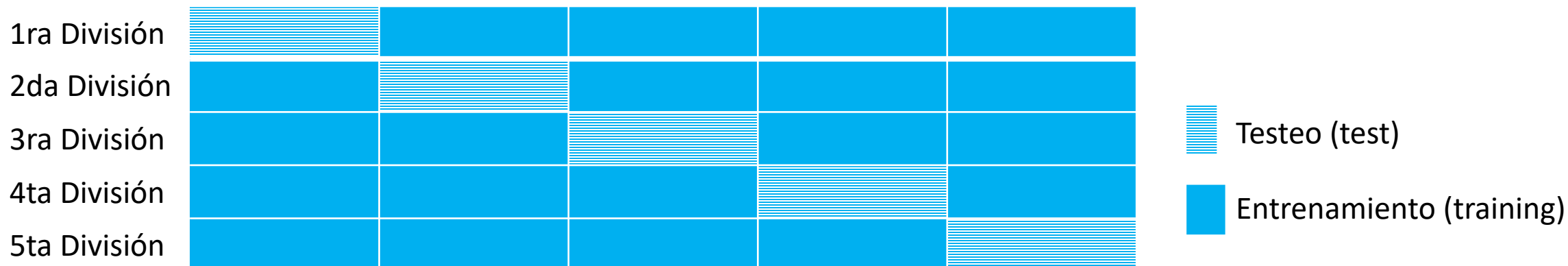
Explicar la relación entre la variable dependiente y todas las demás (las explicativas), cuando la relación es significativa. Esto permite entender el porqué de los resultados, es decir porqué ocurren las relaciones. En este caso si es necesario conocer la forma funcional de \hat{f}

Para validar los modelos

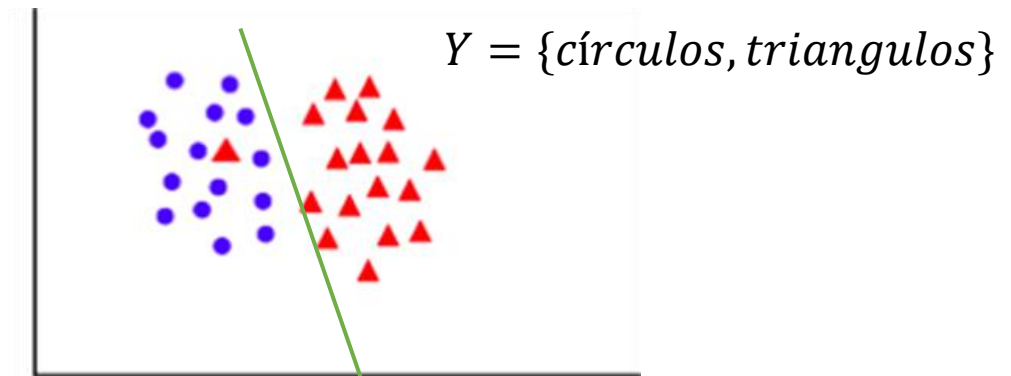
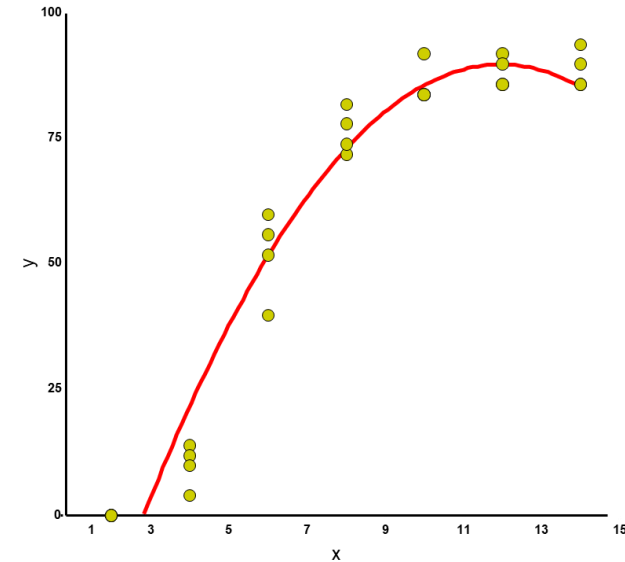
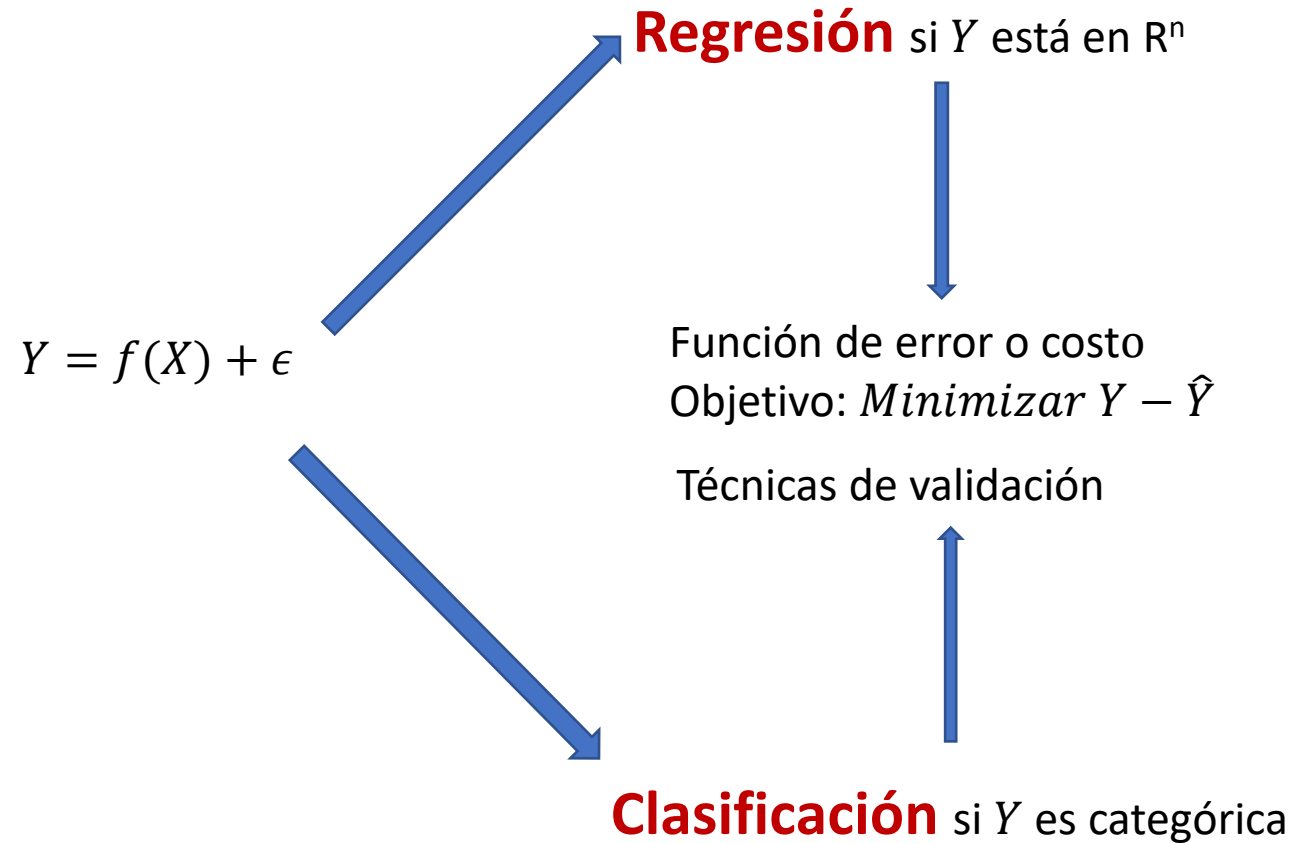
Separar el conjunto de datos en dos muestras aleatorias calcular la regla de predicción sobre



La validación cruzada (Cross-validation) es un método más estable y completo. Los datos se dividen repetidamente y se entrenan varios modelos. La versión más utilizada de validación cruzada es la validación cruzada de k veces, donde k es un número especificado por el usuario, generalmente 5 o 10.



Modelos de regresión vs modelos de clasificación



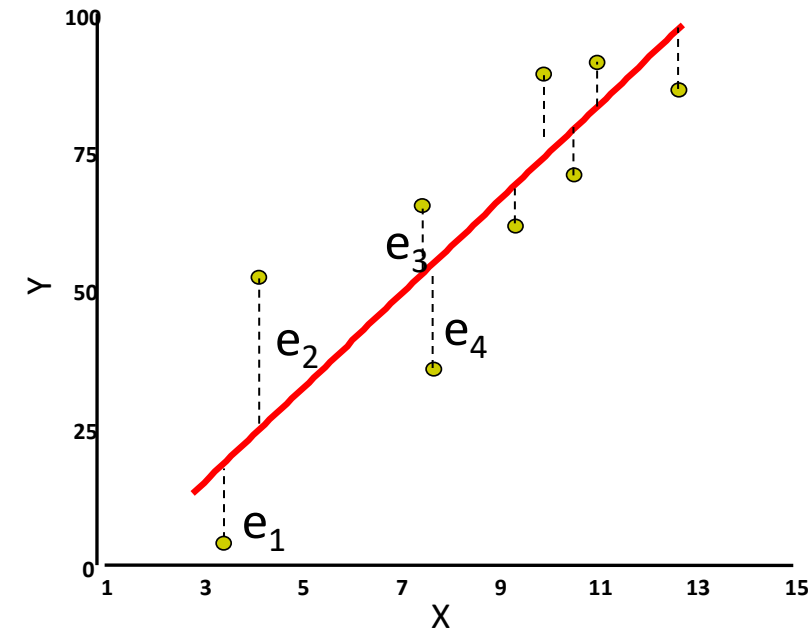
Modelos de regresión Lineal

El modelo de **regresión lineal simple** es el modelo más simple y fácil de comprender

La recta representa la relación del valor promedio de una variable (y) sobre (x).

$$Y = f(X) + \epsilon \Rightarrow Y(x, \mathbf{w}) = w_0 + w_1 X + \epsilon$$

$e_i = y_i - \hat{y}_i$ para cada una de los n ejemplos



Estimación de los coeficientes w_0, w_1 por mínimos cuadrados

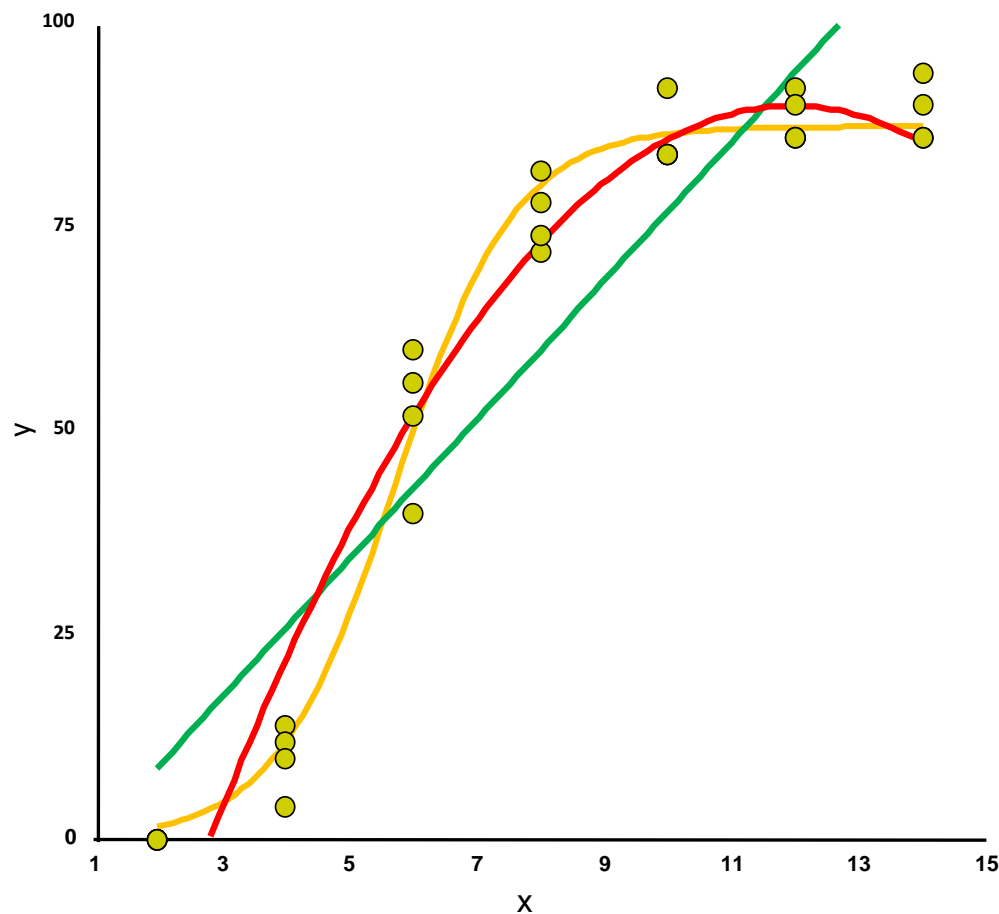
$$\min_{w_0, w_1} \sum_{i=1}^n (e_i)^2 = \min_{w_0, w_1} \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \min_{w_0, w_1} \sum_{i=1}^n (y_i - w_0 - w_1 x_i)^2$$

$$w_0 = \bar{Y} - w_1 \bar{X}$$

Donde \bar{Y} y \bar{X} son los promedios

$$w_1 = \frac{S_{xy}}{S_{xx}} = \frac{Cov(x, y)}{Var(x)}$$

¿Cuál modelo ajusta mejor?



- Lineal simple
- Polinomio cuadrático
- Otro modelo no lineal

Modelos de regresión polinomial

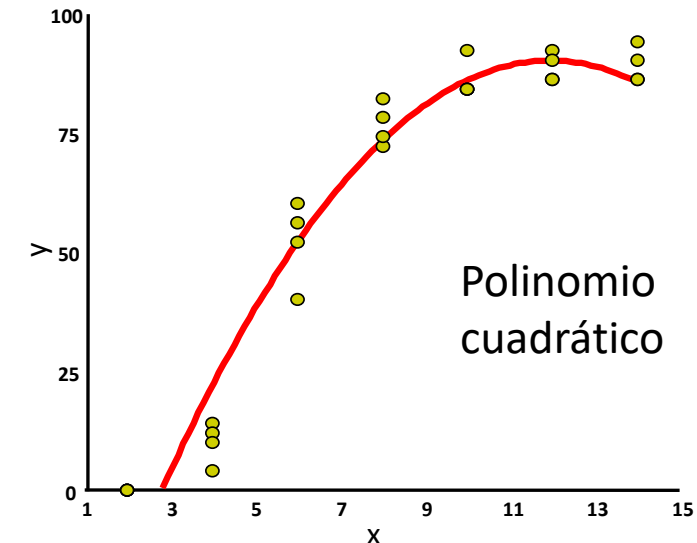
Se puede definir un polinomio de grado M para ajustar el conjunto de puntos

Polinomio de grado 2

$$Y(x, \mathbf{w}) = w_0 + w_1X + w_2X^2 + \epsilon$$

Polinomio de grado M

$$Y(x, \mathbf{w}) = w_0 + w_1X + w_2X^2 + w_3X^3 + \dots + w_M X^M + \epsilon$$

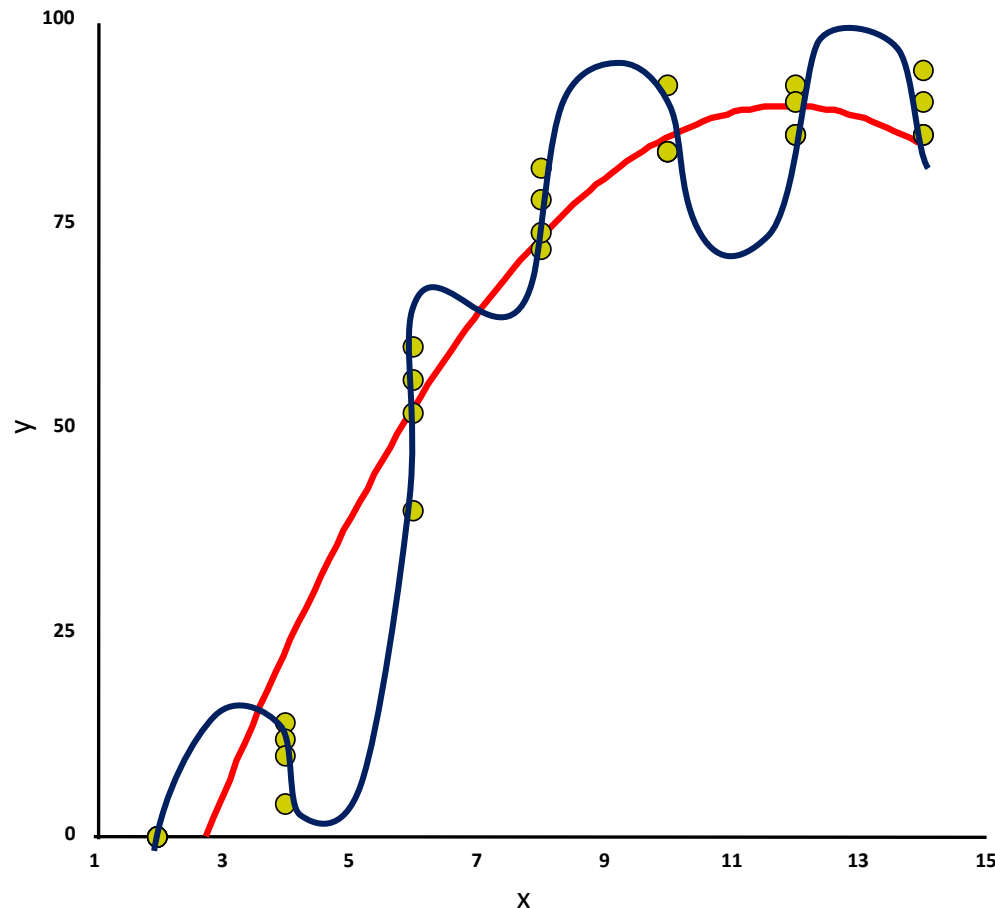


Estimación del vector de coeficientes $\mathbf{w} = (w_0, w_1, \dots, w_p)$ por **mínimos cuadrados**

$$\min_{w_0, w_1} \sum_{i=1}^n (e_i)^2 = \min_{w_0, w_1} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad \longrightarrow \quad \mathbf{w} = [X'X]^{-1}X'y \quad \text{donde } X_{n,p} \text{ es la matriz que contiene los valores de las variables independientes}$$

Podemos agregar más grados al polinomio

M es un hiperparámetro (variable de control externa que se puede variar en el proceso de estimación) ¿qué valor puede tomar M?



- Polinomio cuadrático
- Polinomio de grado M



Problemas de sobre ajuste (overfitting):
El modelo estima bien los datos pero predice mal.
Para estimar muchos parámetros se necesitan muchos ejemplos.

Modelos de regresión múltiple

El modelo se extiende **a más variables predictoras** (numéricas o categóricas)

Utilizaremos un **plano o hiperplano** para describir la dependencia del valor promedio de una variable Y de p variables

$$\mathbf{X} = (X_1, X_2, \dots, X_p)$$

$$Y(x, \mathbf{w}) = w_0 + w_1 X_1 + w_2 X_2 + \dots + w_p X_p + \epsilon \quad e_i = y_i - \hat{y}_i \text{ para cada una de las } n \text{ observaciones}$$

Si llamamos p al número de coeficientes que acompañan a las variables predictoras:

$p = 1$ se obtiene el número de ecuaciones normales de la recta

$p = 2$ se obtiene el sistema de ecuaciones del plano

$p \geq 3$ se obtiene el sistema de ecuaciones del hiperplano

Estimación del vector de coeficientes $\mathbf{w} = (w_0, w_1, \dots, w_p)$ por **mínimos cuadrados**

$$\min_{w_0, w_1} \sum_{i=1}^n (e_i)^2 = \min_{w_0, w_1} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad \longrightarrow \quad \mathbf{w} = [X'X]^{-1} X'y$$

donde $X_{n,p}$ es la matriz que contiene los valores de las variables independientes

Medidas de ajuste de modelos de regresión

Error cuadrático medio (Mean Squared Error)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n e_i^2$$

Error absoluto medio (Mean Absolute Error)

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |e_i|$$

Error absoluto mediano (Median absolute error)

$$\text{MedAE} = \text{Mediana} \{|e_1|, |e_2|, \dots, |e_n|\}$$

No está afectado por datos atípicos

Modelos de clasificación. Conceptos generales

Se conoce la clase o etiqueta de pertenencia de cada ejemplo

Probabilidad a priori $P(Y = g) = \pi_g$ para $g = 1, 2, \dots, G$

*Se definen una o más características $\mathbf{X} = (X_1, X_2, \dots, X_p)$
y la función de probabilidad condicional a la clase*

$$P(\mathbf{X}/Y=g) = f_g(\mathbf{X}) \text{ para } g = 1, 2, \dots, G$$

Si es conocida la forma funcional $f_g(\mathbf{X})$ con el teorema de Bayes se puede calcular la probabilidad de pertenecer a una clase o etiqueta

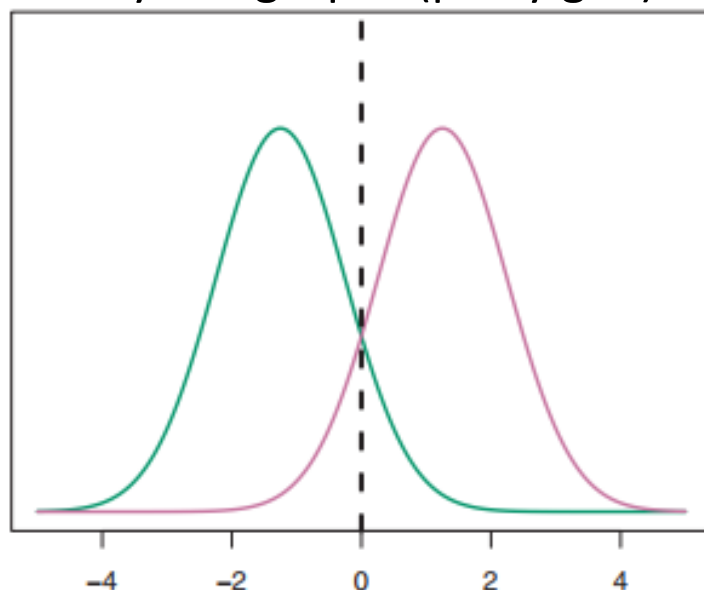
$$P(Y = g/\mathbf{X}) = \frac{\pi_g f_g(\mathbf{X})}{f(\mathbf{X})} = \frac{\pi_g f_g(\mathbf{x})}{\sum_{g=1}^G \pi_g f_g(\mathbf{x})}$$

Una nueva observación se asigna al máximo valor de $\pi_g f_g(\mathbf{X})$

Modelos de clasificación. Discriminante lineal y cuadrático

Si la función de probabilidad condicional a la clase $P(X/Y=g) = f_g(X)$ tiene distribución normal multivariada para cada clase $g = 1, 2, \dots, G$

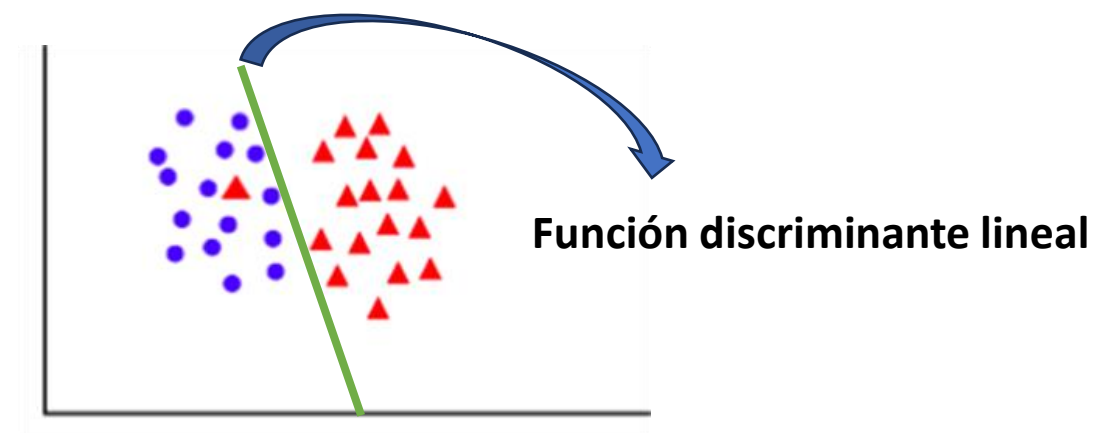
Ejemplo para una variable predictora
y dos grupos ($p=1$ y $g=2$)



Si la estructura de variabilidad de cada clase es:

- *parecida se usa LDA (Linear Discriminant Analysis)*
- *es distinta se usa el QDA (Quadratic Discriminant Analysis)*

Tienen buenos resultados si hay separabilidad lineal

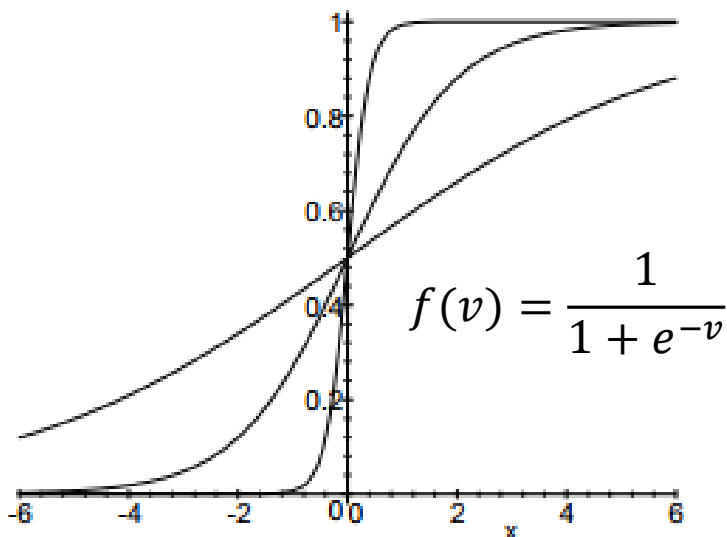


Modelos de clasificación. Discriminante logístico

Una función f que transforme el resultado de la regresión en un número entre 0 y 1

$$P(Y = g/\mathbf{X}) = f(w_0 + w_1X_1 + w_2X_2 + \dots + w_p X_p + \epsilon) = f(\mathbf{w}^t \mathbf{X})$$

Función logística



Para $g = 0, 1$

$$P(Y = 0/\mathbf{X}) = f(v) = \frac{1}{1 + e^{-(\mathbf{w}^t \mathbf{X})}}$$

$$P(Y = 1/\mathbf{X}) = 1 - f(v) = \frac{e^{-(\mathbf{w}^t \mathbf{X})}}{1 + e^{-(\mathbf{w}^t \mathbf{X})}}$$

Odds ratio $\ln \left(\frac{P(Y = 1/\mathbf{X})}{P(Y = 0/\mathbf{X})} \right) = \mathbf{w}^t \mathbf{X}$

Cuando hay más de dos etiquetas $g = 1, 2, \dots, G$, se hace lo mismo tomando una etiqueta como base o referencia (este es el **modelo multinomial**)

Ver en https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

El punto 1.1.11. Logistic regression

Modelos de clasificación. Modelo de clasificación logística múltiple

Cuando hay más de dos etiquetas $g = 1, 2, \dots, j, \dots, G$, se hace lo mismo tomando una etiqueta como base o referencia (este es el **modelo multinomial**)

Función softmax

La función softmax calcula la probabilidad de cada etiqueta sobre todas las etiquetas posibles. Dada las características de un objeto se asigna a la etiqueta con mayor probabilidad. Esta función assume valores entre 0 y 1, y la suma de los valores es igual a uno. Se utiliza también en distintos niveles de capas de redes neuronales.

$$P(Y = j/X) = \frac{e^{-(w_j^t X_j)}}{\sum_{g=1}^G e^{-(w^t X)} + 1}$$

Ver en https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

El punto 1.1.11. Logistic regression

Estimación por máxima verosimilitud

Para un conjunto de n ejemplos $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ donde cada $y_i = 0, 1$ para $i = 1, 2, \dots, n$

Los parámetros w se estiman iterativamente maximizando la siguiente función de verosimilitud (que se construye a partir de un modelo bipuntual).

$$l(\mathbf{w}) = \prod_{i=1}^n f(y_i) = \prod_{i=1}^n p(\mathbf{X}_i)^{y_i} (1 - p(\mathbf{X}_i))^{1-y_i} \quad \text{siendo} \quad \mathbf{w}^t = (w_0, w_1, w_2, \dots, w_p)$$

Tomando logaritmo y reemplazando las probabilidades por sus correspondientes expresiones

$$L(\mathbf{w}) = \ln l(\mathbf{w}) = \sum_{i=1}^n \left\{ y_i \ln \frac{e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}}{1 + e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}} + (1 - y_i) \ln \frac{1}{1 + e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}} \right\}$$

La función de costo se define como $C(\mathbf{w}) = -\ln l(\mathbf{w})$ el objetivo es entonces minimizar

$$\min_{\mathbf{w}} C(\mathbf{w}) = \min_{\mathbf{w}} \sum_{i=1}^n \left\{ -y_i \ln \frac{e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}}{1 + e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}} - (1 - y_i) \ln \frac{1}{1 + e^{e^{-(\mathbf{w}^t \mathbf{X}_i)}}} \right\}$$

El problema de optimización no tiene solución en forma cerrada, los métodos propuestos por la librería `scikit learn` son: "lbfgs", "liblinear", "newton-cg", "newton-cholesky", "sag" and "saga".

Ver detalles en https://scikit-learn.org/stable/modules/linear_model.html#solvers

Matriz de confusión y medidas de desempeño





16

**Tasa de
clasificación
correcta
(Accuracy**



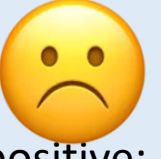
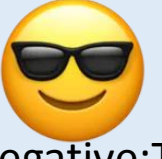
(TP+TN)/T

**Tasa de error de
clasificación
(1- Accuracy)**

(FP+FN)/T

Condición actual	Predicción		
	Etiqueta 1 Positivo o éxito (PP)	Etiqueta 2 Negativo o fracaso (PN)	
Etiqueta 1 Positivo o éxito (P)	 True positive: TP	 False negative: FN	TP + FN=P
Etiqueta 2 Negativo o fracaso (N)	 False positive: FP	 True negative:TN	FP + TN= N
	TP + FP=PP	FN + TN= PN	P+N= T=PP+PV T(Total)

Más medidas de desempeño

Condición actual	Predicción		
	Etiqueta 1 Positivo o éxito (PP)	Etiqueta 2 Negativo o fracaso (PN)	
Etiqueta 1 Positivo o éxito (P)	 True positive: TP	 False negative: FN	TP + FN = P
Etiqueta 2 Negativo o fracaso (N)	 False positive: FP	 True negative: TN	FP + TN = N
	TP + FP = PP	FN + TN = PN	P + N = T = PP + PV T (Total)

Precisión

positive predictive value
(PPV)

$$TP/PP$$

Valor predictivo negativo

negative predictive value
(NPV)

$$TN/PN$$

Sensibilidad

sensitivity, recall, hit rate,
or true positive rate (TPR)

$$TP/P$$

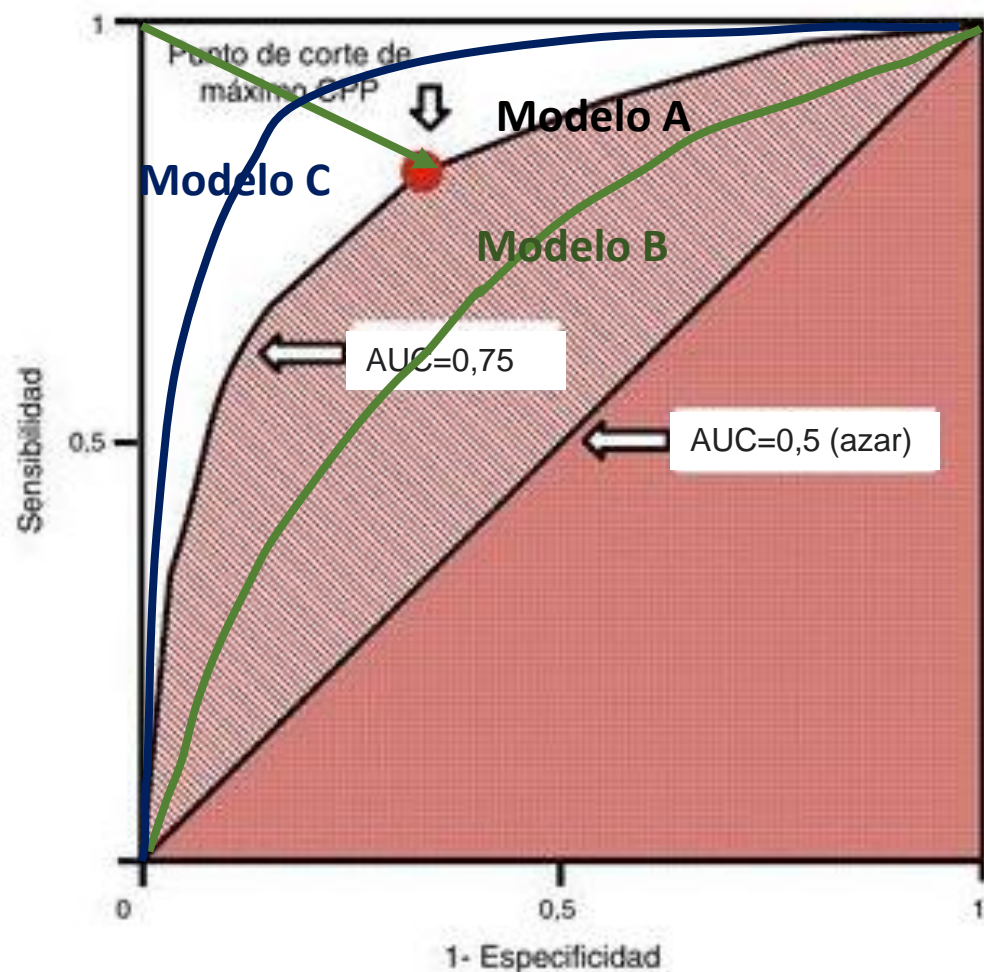
Especificidad

specificity, selectivity or true
negative rate (TNR)

$$TN/N$$

Curvas ROC (Receiver Operating Characteristic)

Curvas de rendimiento diagnóstico, son una representación gráfica de la sensibilidad frente a la especificidad para un clasificador binario según se varía el umbral de discriminación (punto de corte).



Sirve para :

- ✓ Conocer el rendimiento global de una prueba
- ✓ Elegir el umbral de discriminación o punto de corte apropiado
- ✓ Comparar dos pruebas o dos puntos de corte. La elección se realiza mediante la comparación del **área bajo la curva** (AUC) para cada prueba

Para interpretar AUC :

- [0.5]** es igual que tomar un resultado al azar
- [0.5, 0.6)** malo.
- [0.6, 0.75)** regular.
- [0.75, 0.9)** bueno.
- [0.9, 0.97)** muy bueno.
- [0.97, 1)** excelente.

Dilema sesgo (bias) - varianza de la predicción

El valor esperado del error cuadrático medio en la muestra de testeo para un vector de características \mathbf{x}_0 esto es, el error cuadrático medio promedio si tuviéramos muchas muestras de entrenamiento diferentes

$$E(y_o - \hat{f}(x_o))^2$$

Resolviendo algebraicamente se puede descomponer en:

$$E(y_o - \hat{f}(x_o))^2 = Var(\hat{f}(x_o)) + (bias(\hat{f}(x_o)))^2 + var(e^2)$$

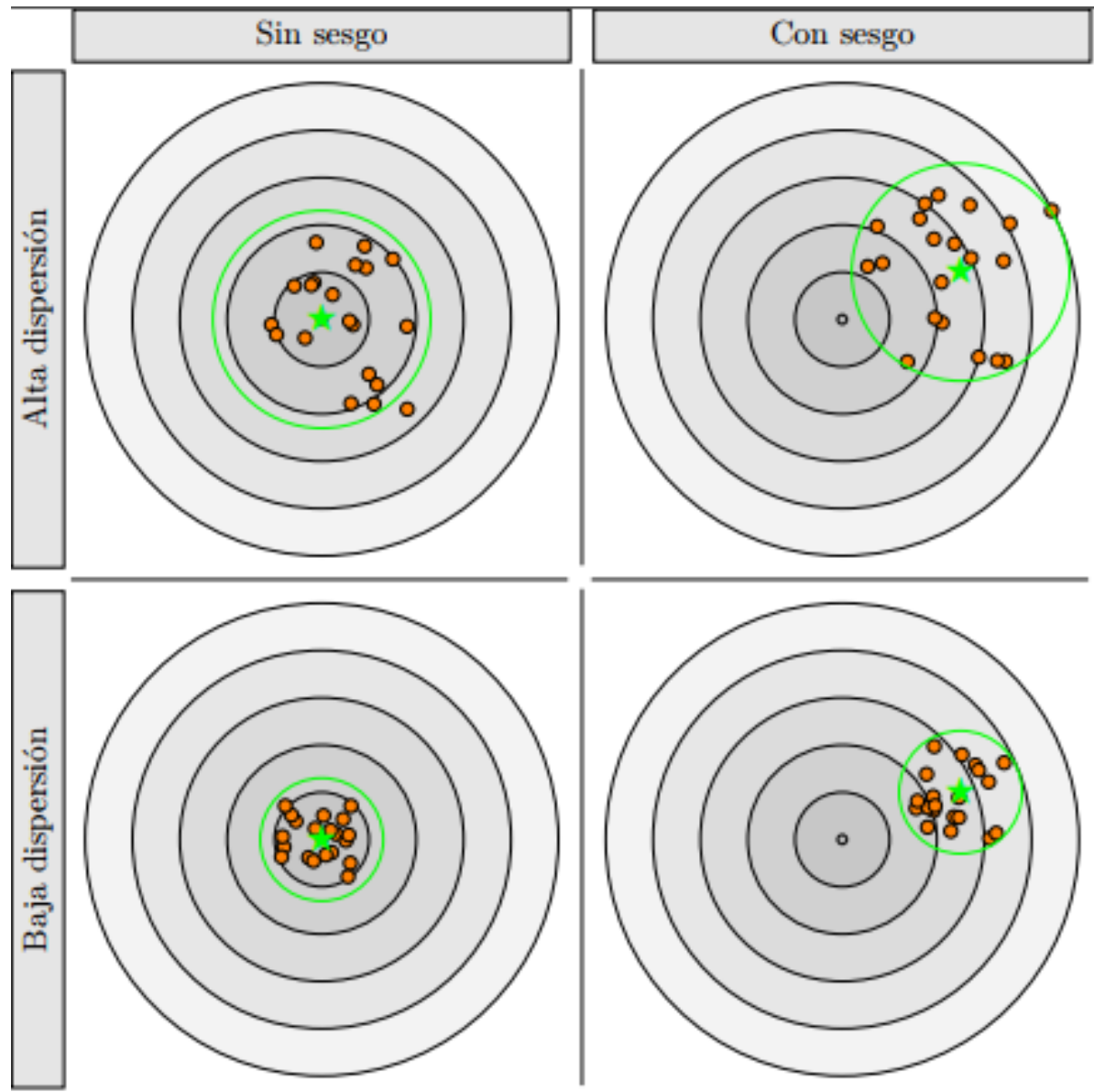
Error Total = Varianza + Bias² + Error Irreducible

Cuanto cambia la predicción cuando se usan distintos conjuntos de entrenamiento.

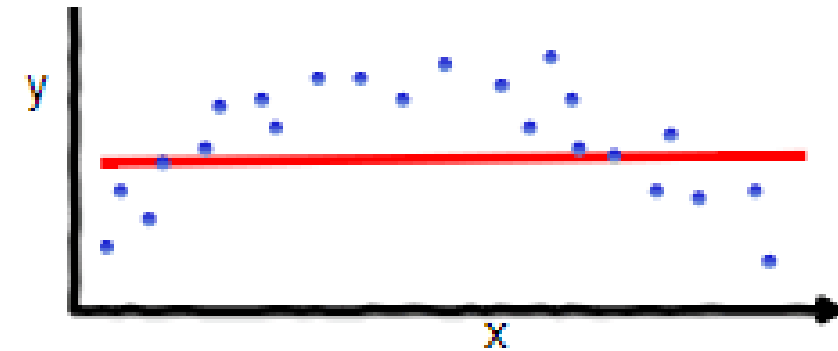
Diferencia entre la predicción esperada y el verdadero valor

Problemas de especificación del modelo

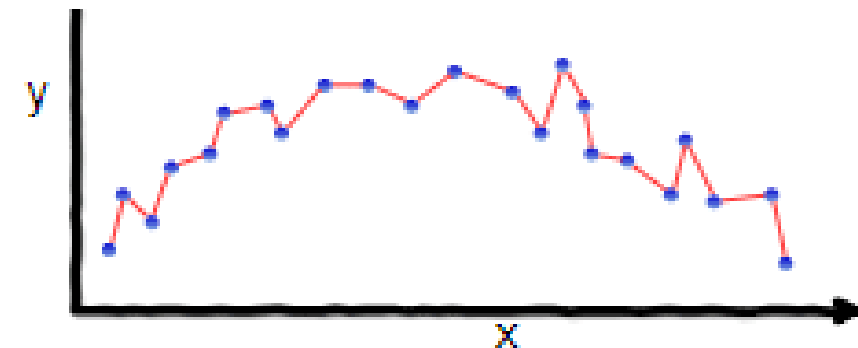
Dilema sesgo (bias) - varianza de los modelos



Alto sesgo: el modelo es muy simple y no se ha ajustado a los datos de entrenamiento (*underfitting*)



Alta dispersión (varianza): el modelo se ajusta muy bien a los datos de entrenamiento (*overfitting*)



Dilema sesgo (bias) - varianza de los modelos

