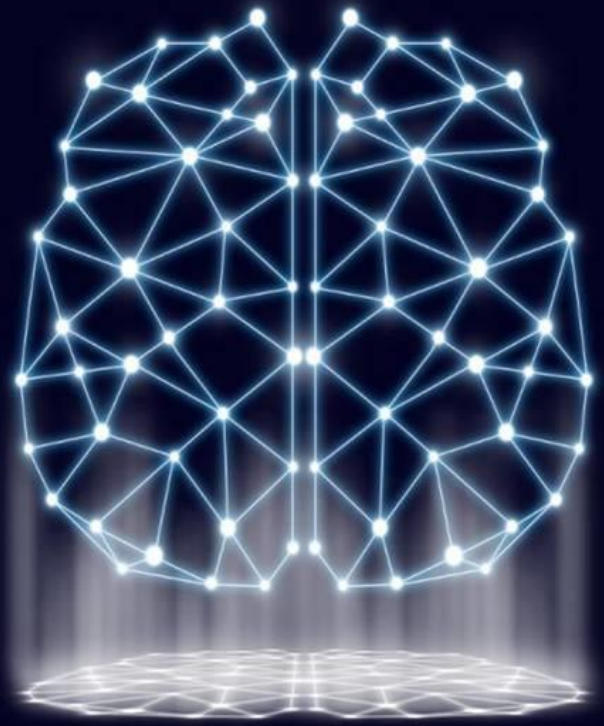
FULLSTACK



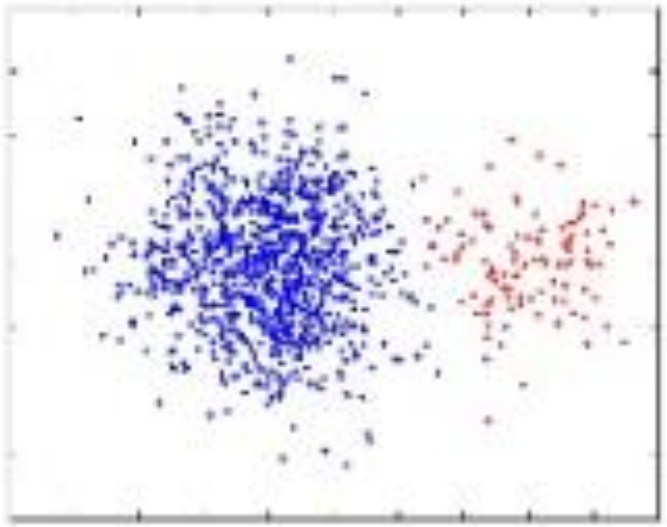
Curso de Especialización de Machine Learning con Python

TÉCNICAS DE BALANCEO DE DATOS

¿Qué es data desbalanceada?

Comúnmente hace referencia a la variable “Target”.

Def: Se da cuando la frecuencia de clases de la variable Target son muy distintas o muy desiguales.



Data desbalanceada

¿Qué consecuencias puede traer?

Al momento de entrenar un algoritmo de ML con el dataset desbalanceado, se puede originar un sesgo hacia una clase en particular de la variable target. Hacia la clase mayoritaria.

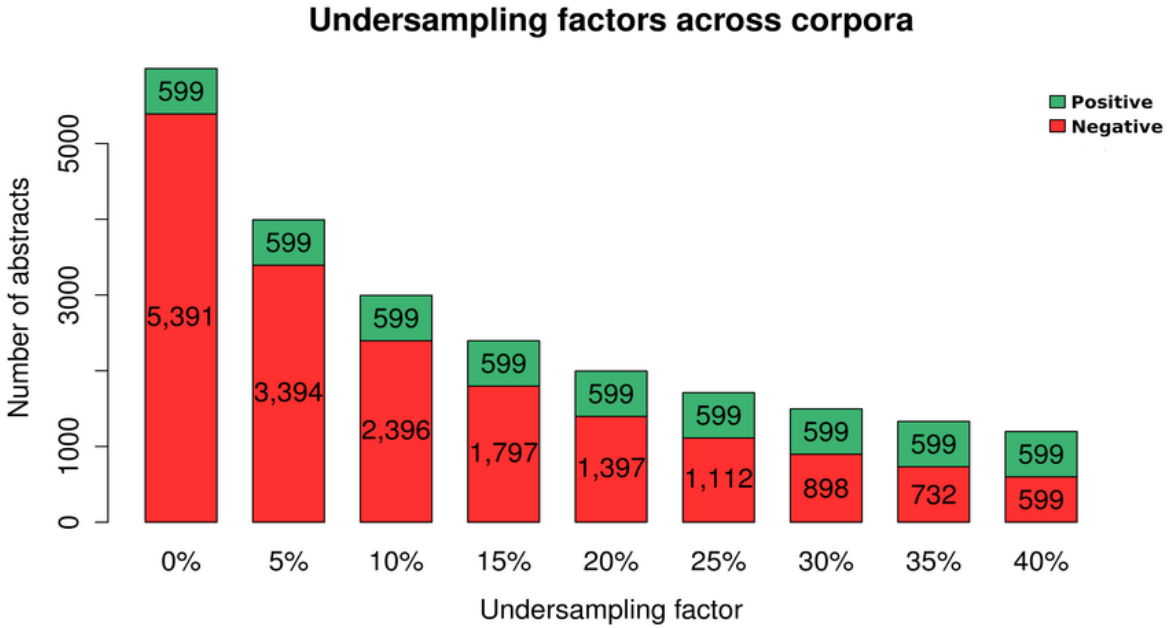
¿Cómo lo solucionamos?

Balanceando la data.

- Técnicas: Undersampling, Oversampling, SMOTE, a criterio propio, otras

UNDERSAMPLING

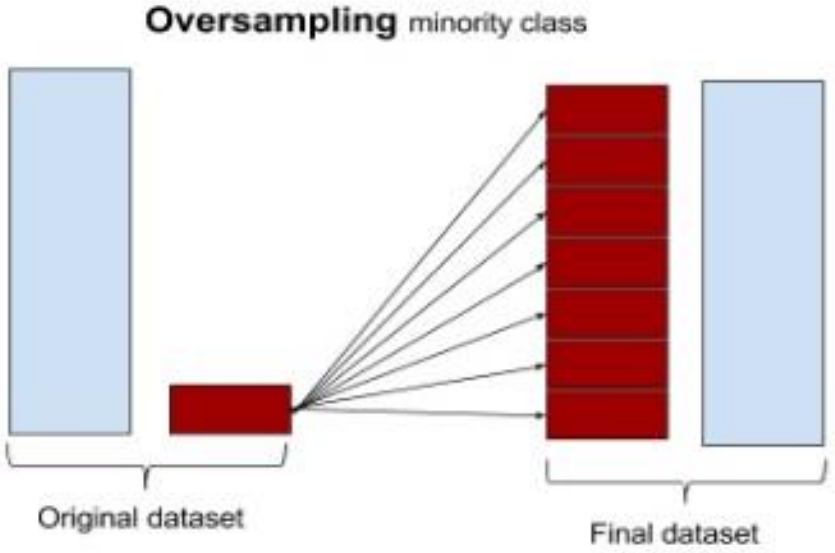
En esta técnica se busca reducir la cantidad de registros de la clase mayoritaria a la cantidad de registros de la clase minoritaria.



Mayor a menor → Undersampling

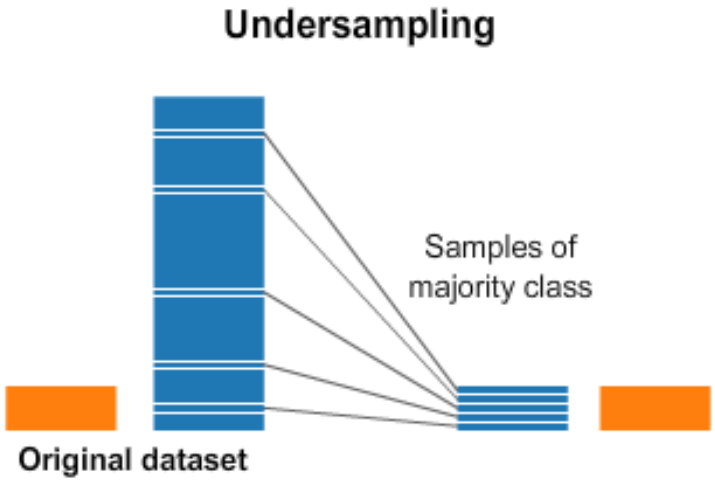
OVERSAMPLING

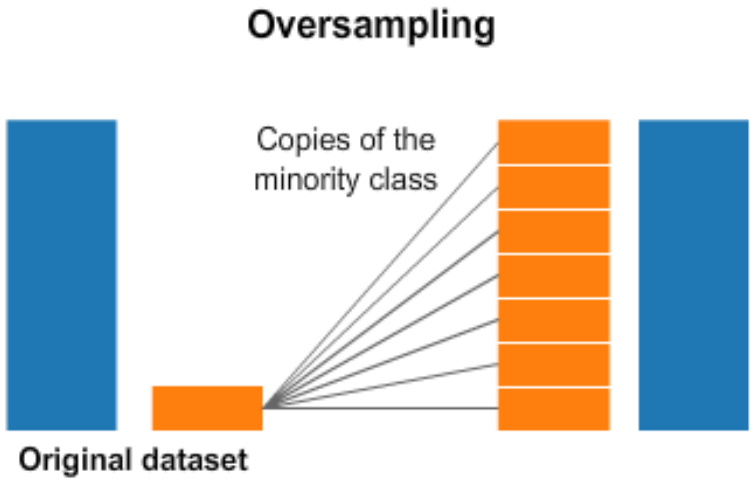
En esta técnica se busca incrementar la cantidad de registros de la clase minoritaria a la cantidad de registros de la clase mayoritaria.



Menor a mayor → Oversampling

RESUMEN


Undersampling
Samples of
majority class
Original dataset

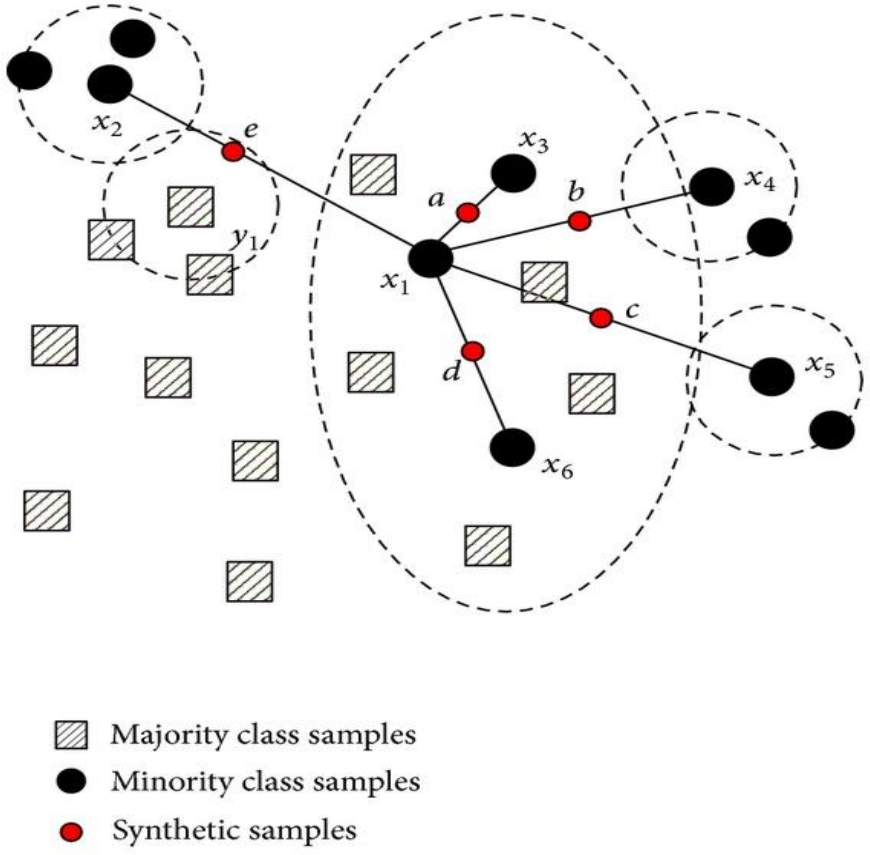

Oversampling
Copies of the
minority class
Original dataset

SMOTE

Synthetic Minority Over-sampling Technique

Es una técnica de oversampling.

El objetivo es crear puntos sintéticos a partir de la data de la clase minoritaria.
Ejemplo: k = 5



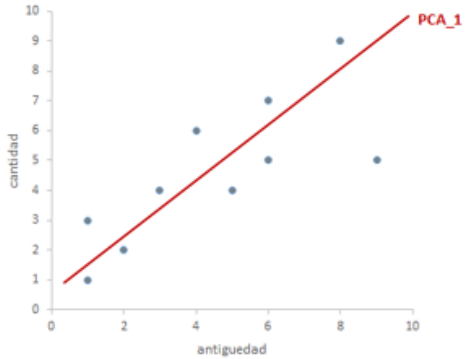
Análisis de Componentes Principales (PCA)

Técnica utilizada para describir un set de datos en términos de nuevas variables ("componentes") no correlacionadas.

Objetivo

- Reducir la dimensionalidad de un conjunto de datos
- Convertir un conjunto de observaciones de variables posiblemente correlacionadas en un conjunto de valores de variables sin correlación

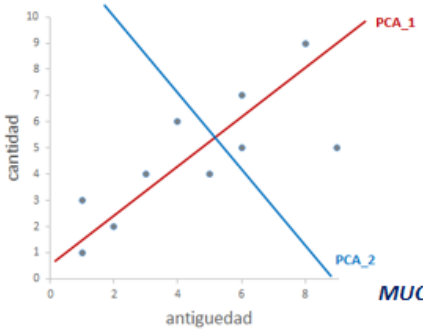
PCA 1



nombre	PCA_1
ana	-4.98
pdro	3.76
maria	-1.54
jose	-3.58
luisa	0.01
pablo	0.49
laura	1.42
luis	5.49
jesus	-3.73
lucia	2.67

En este ejemplo el PCA_1 tiene el **89%** de la varianza

PCA 1 + PCA 2



nombre	PCA_1	PCA_2
ana	-4.98	-0.61
pdro	3.76	-2.51
maria	-1.54	0.47
jose	-3.58	-0.46
luisa	0.01	-0.78
pablo	0.49	1.40
laura	1.42	-0.63
luis	5.49	1.23
jesus	-3.73	0.95
lucia	2.67	0.93
varianza	*11.95*	*1.485*

MUCHA informacion

Poca informacion

En este ejemplo el PCA_2 tiene el **11%** de la varianza (1.485/(11.95+1.48))

Módulo 5:
Construcción y Evaluación de Modelos

Tipos de Modelos

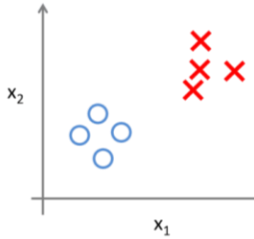
Datos Estructurados

Pre-Procesamiento

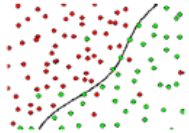
Datos No Estructurados

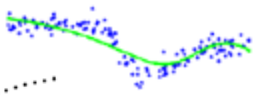
- Texto
- Voz
- Imágenes
- Video

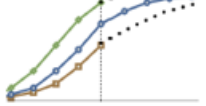
Aprendizaje Supervisado



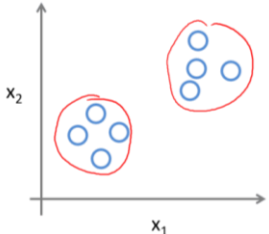
- ❑ Clasificación
- ❑ Regresión
- ❑ Pronóstico



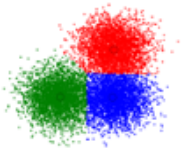


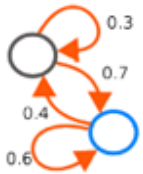


Aprendizaje No Supervisado



- ❑ Segmentación
- ❑ Asociación
- ❑ Secuenciación





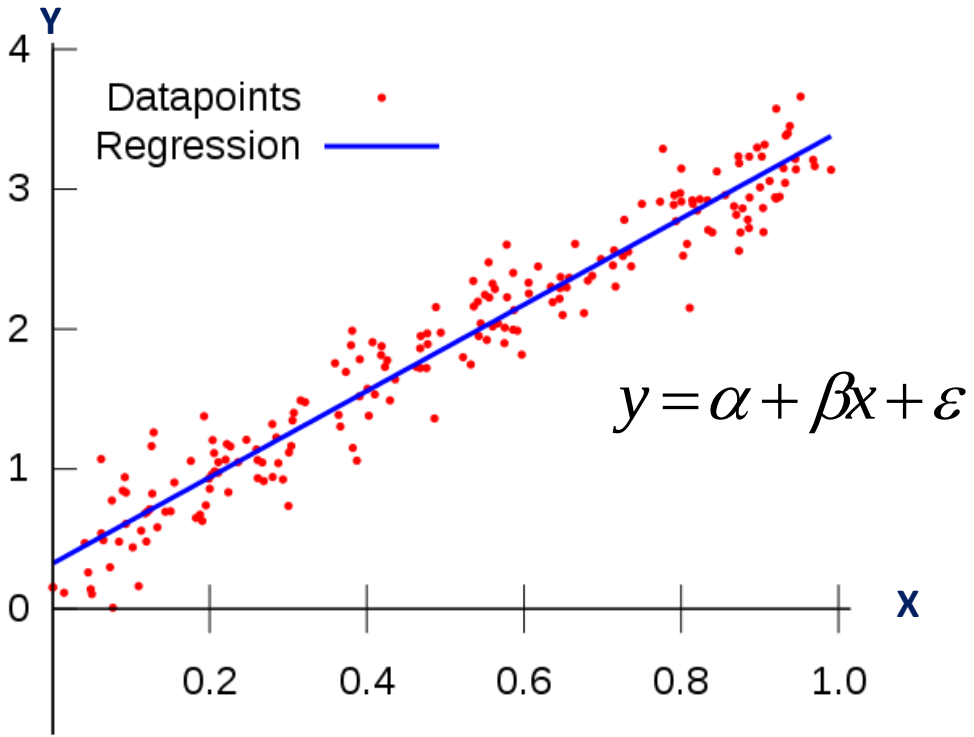
Regresión Lineal

Modelo matemático para estimar los valores de una **variable continua** (*dependiente*) en función de otra(s) variable(s) (*independientes*).

Modelo: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + \varepsilon$

Y = Variable dependiente
(Predicha o Explicada)
X = Vector de Variables Independientes
(Predictora o Explicativa)
β = Vector de Coeficientes

Se llama **regresión lineal <u>simple</u>** cuando solo existe una variable independiente, o **<u>múltiple</u>** si existen varias variables independientes.



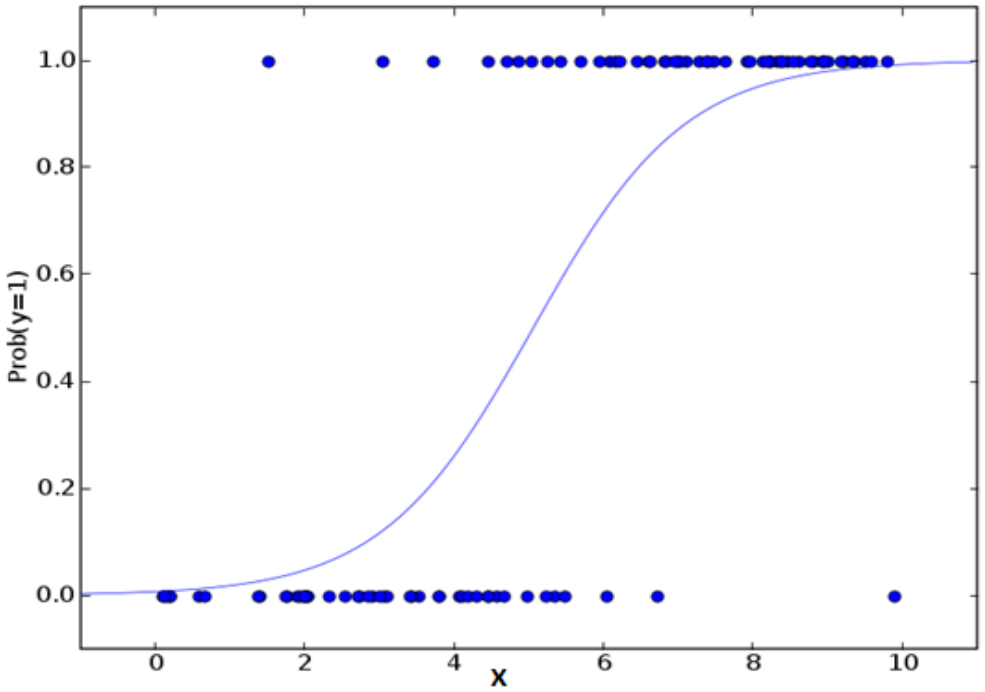
Regresión Logística

Modelo matemático para estimar el valor de una **variable categórica o discreta** (*dependiente*) en función de otra(s) variable(s) (*independientes*). Predice la probabilidad de ocurrencia de un evento ajustando los datos a una **función logit**.

p = Probabilidad de ocurrencia de un evento
1 – p = Probabilidad de no ocurrencia de un evento

Para poder predecir la variable binaria, se transforma la regresión lineal en una regresión logística, convirtiendo **"y"** en "**ln(p/(1-p)"** y luego se aplica una regresión lineal sobre esta transformación.

$$\ln\left(\frac{p}{1-p}\right) = z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + u$$



$$\Pr ob(Y_i = 1) = \frac{1}{1+e^{-(\alpha+\beta_k X_{ki})}} = \frac{e^{\alpha+\beta_k X_{ki}}}{1+e^{\alpha+\beta_k X_{ki}}}$$

Regresión Logística

Datos

y	x
0	25
1	54
1	67
0	6
1	77
0	49
1	81

$$y = \beta_0 + \beta_1 x_1 + u$$

Paso 1

Se tranforma **y** en el logaritmo de la probabilidad de **y**

A la transformacion **P/(1-P)** tambien le dicen **Odds Ratio,** donde P=probabilidad de **y**

$$\ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 x_1 + u$$

Paso 2

Se calcula la regresion lineal para predeir el logaritmo de los **Odds Ratio**

Paso 3

Transformar el resultado de la regresion lineal en la probaiblidad final

$$\frac{e^{\beta_0 + \beta_1 x_1 + u}}{1 + e^{\beta_0 + \beta_1 x_1 + u}}$$

Donde e es una constante = 2.718281828

$$\ln\left(\frac{p}{1-p}\right) = z$$

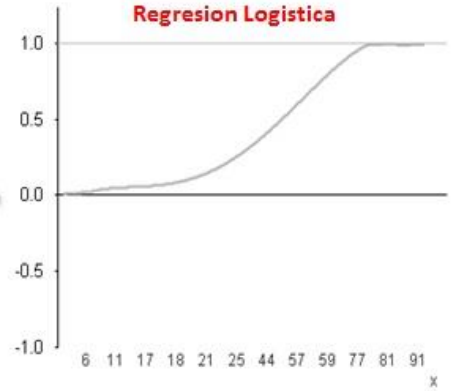
$$e^{\ln\left(\frac{p}{1-p}\right)} = e^z$$

$$\frac{p}{1-p} = e^z$$

$$p = e^z - pe^z$$

$$p(1 + e^z) = e^z$$

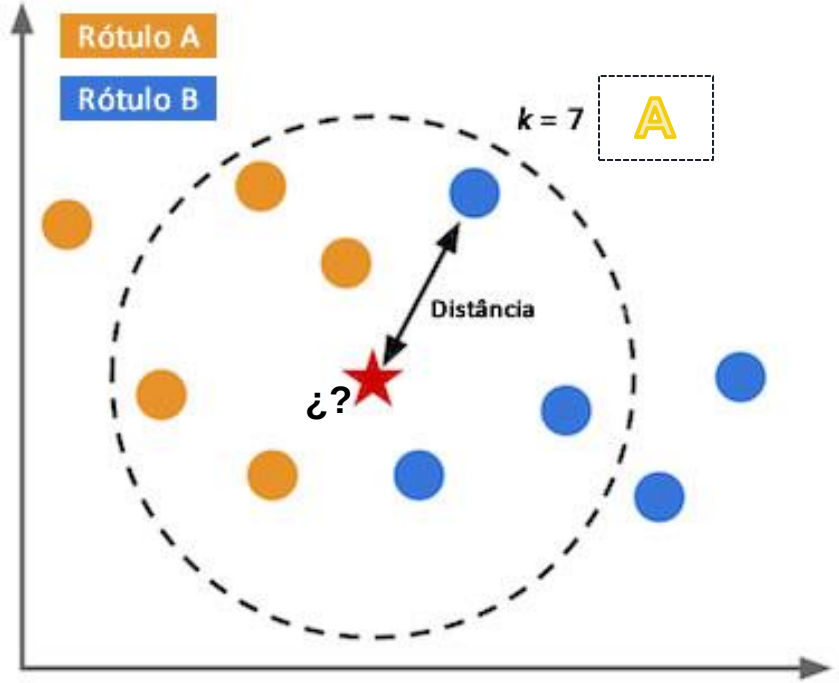
$$\text{(prob final) } p = \frac{e^z}{1+e^z}$$



Fuente: http://apuntes-r.blogspot.pe/2015/06/regresion-logistica.html

Knn (K vecinos más cercanos)

Es un simple algoritmo que almacena todos los casos disponibles en el “entrenamiento” y clasifica los nuevos casos por el voto mayoritario de sus k vecinos más cercanos (según una función de distancia). Se puede usar para problemas de **clasificación** y **regresión**.



Funciones de Distancia:

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Vars Continuas

Hamming Distance

$$D_H = \sum_{i=1}^{k}|x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

Vars Categóricas

Naive Bayes

Los métodos de Naive Bayes son un conjunto de algoritmos de aprendizaje supervisado basados en la aplicación del teorema de Bayes con la suposición de independencia "ingenua" (Naives) entre cada par de características.

Según el supuesto "Naives" de independencia condicional

$$p(F_i|C, F_j) = p(F_i|C)$$

Entonces la formula anterior queda así

$$\begin{aligned} p(C)\, p(F_1, \ldots, F_n|C) &= p(C)\, p(F_1|C)\, p(F_2|C)\, p(F_3|C)\, \cdots \\ &= p(C) \prod_{i=1}^{n} p(F_i|C). \end{aligned}$$

Modelo de probabilidad para un clasificador

$$p(C|F_1, \ldots, F_n)$$

Según teorema de Bayes $p(C|F_1, \ldots, F_n) = \dfrac{p(C)\, p(F_1, \ldots, F_n|C)}{p(F_1, \ldots, F_n)}.$

Numerador

El objetivo es hallar $p(C)\, p(F_1, \ldots, F_n|C)$ (numerador)

Z

Es igual a

$$\begin{aligned} &= p(C)\, p(F_1, \ldots, F_n|C) \\ &= p(C)\, p(F_1|C)\, p(F_2, \ldots, F_n|C, F_1) \\ &= p(C)\, p(F_1|C)\, p(F_2|C, F_1)\, p(F_3, \ldots, F_n|C, F_1, F_2) \\ &= p(C)\, p(F_1|C)\, p(F_2|C, F_1)\, p(F_3|C, F_1, F_2)\, p(F_4, \ldots, F_n|C, F_1, F_2, F_3) \end{aligned}$$

Y por lo tanto

$$p(C|F_1, \ldots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^{n} p(F_i|C)$$

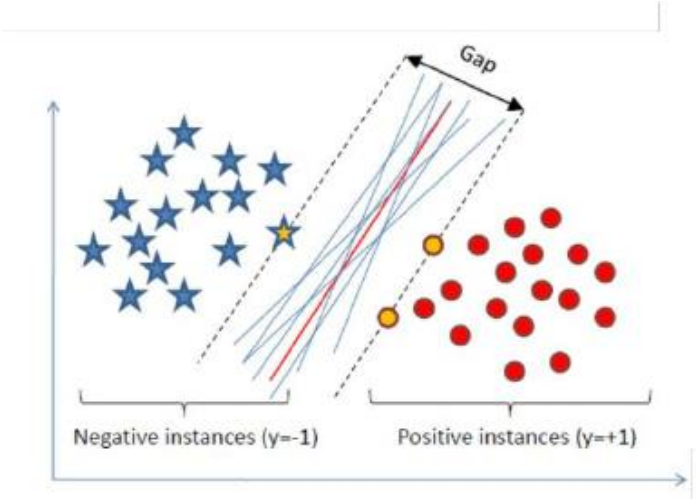
Algoritmos:

- Gaussian Naive Bayes
- Multinomial Naive Bayes
- Bernoulli Naive Bayes

Support Vector Machine (SVM)

Es un algoritmo de aprendizaje supervisado cuyo objetico es encontrar un hiperplano canónico que maximice el margen del conjunto de datos de entrenamiento, esto nos garantiza una buena capacidad de generalización

Maquina de Aprendizaje Lineal



Problema con SVM

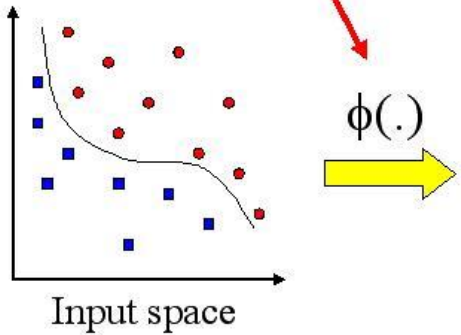
En los dataset de aplicación comunes existen:

a) Más de dos variables predictoras
b) Curvas no lineales de separación
c) Casos donde los conjuntos de datos no pueden ser completamente separados
d) Clasificaciones en más de dos categorías.

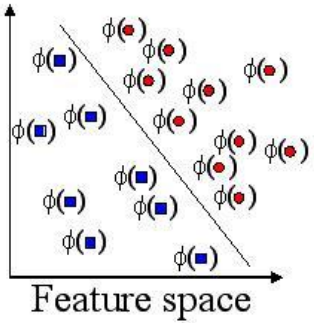
Para solucionarlo utilizamos los KERNEL.

Kernel

Kernel Function



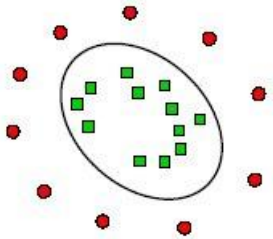
$\phi(.)$



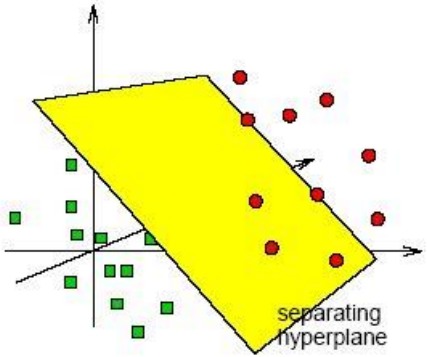
Non -linearly separable

Linearly separable

Separation may be easier in higher dimensions



feature map

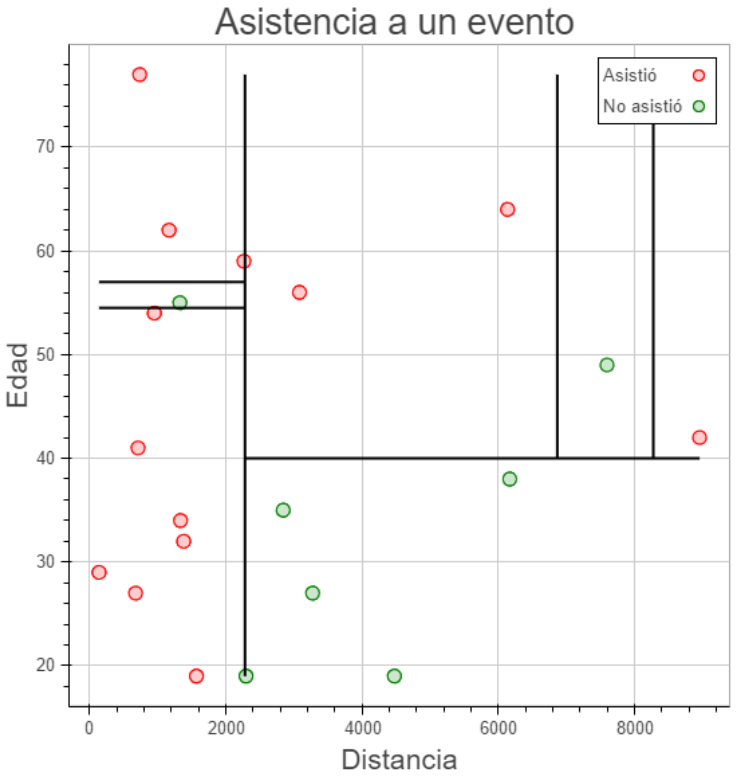


complex in low dimensions

simple in higher dimensions

Árbol de Decisión

Técnica de Aprendizaje Supervisado no-paramétrica que puede ser usado tanto para predecir una variable categórica (clasificación) y continua (regresión). Los árboles responden preguntas secuenciales que nos envían a cierta ruta del árbol dada la respuesta. El modelo se comporta usando condiciones de “si esto se cumple entonces" que finalmente produce un resultado específico.

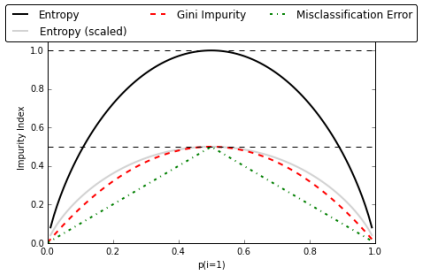


Funciones para división:

Índice de Gini $1 - \sum_j p_j^2$

Entropía $-\sum_j p_j log_2(p_j)$

Error de Clasificación $1 - max(p_j)$



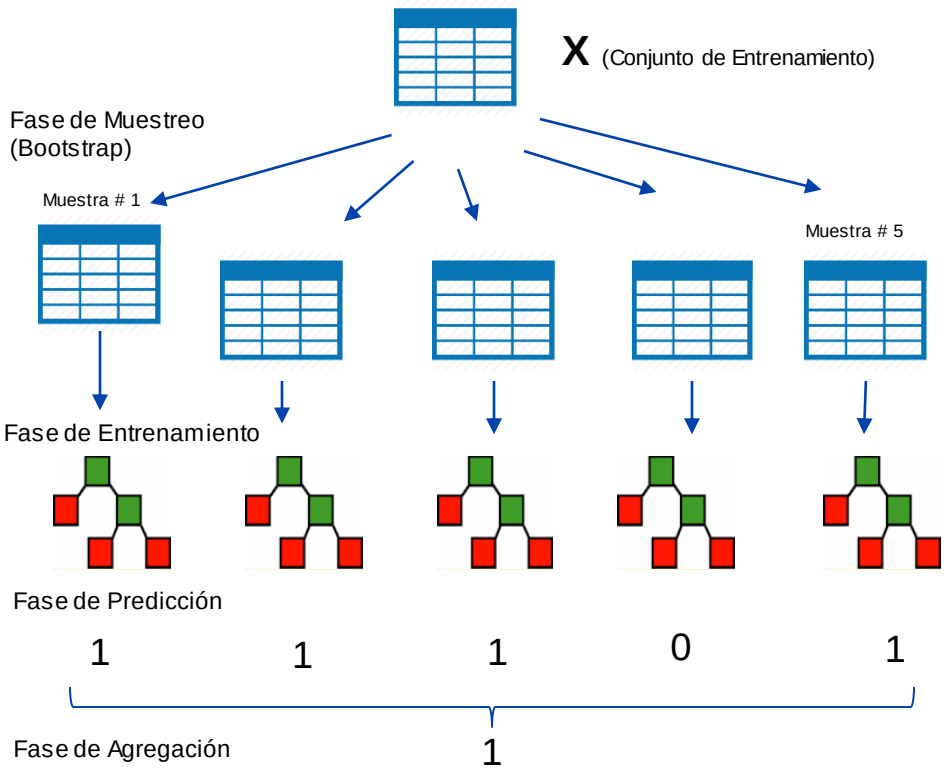

Y SI COMBINAMOS VARIOS ÁRBOLES?...

Métodos Ensamblados (Bagging)

El aprendizaje ensamblado (o "conjunto") es el proceso de combinar varios modelos predictivos para producir un modelo combinado que es más preciso que cualquier modelo individual.

- Regresión: tomar el promedio de las predicciones.
- Clasificación: vote y use la predicción más común, o tome el promedio de las probabilidades.

"Si tienes todas las opiniones de un comité de expertos, considéralas todas para tomar una decisión"



Métricas de Evaluación de Clasificación

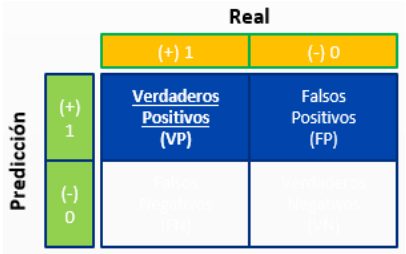
Matriz de Confusión

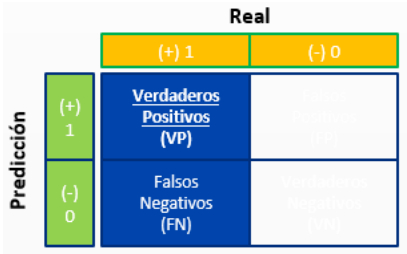
Predicción \ Real	(+) 1	(-) 0
(+) 1	**Verdaderos Positivos (VP)**	Falsos Positivos (FP)
(-) 0	Falsos Negativos (FN)	**Verdaderos Negativos (VN)**

$$tasa\ de\ aciertos = \frac{VP + VN}{total}$$ ***(accuracy)***

$$tasa\ de\ error = \frac{FN + FP}{total}$$

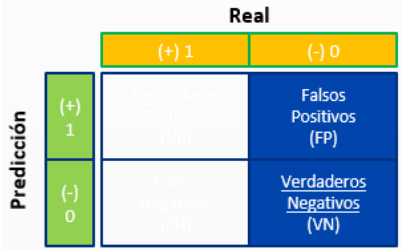
$$precisión = \frac{VP}{VP + FP}$$





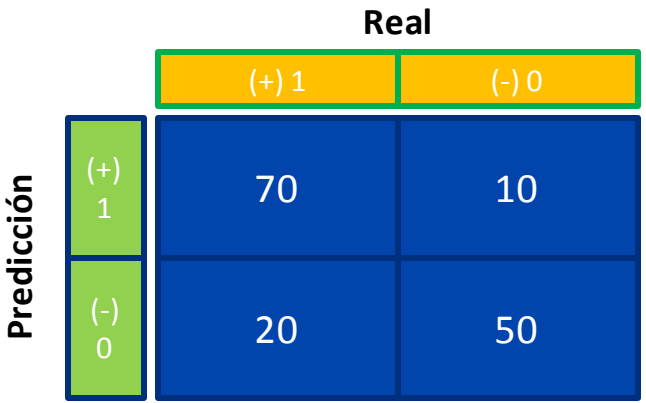
$$sensibilidad = \frac{VP}{VP + FN}$$

$$especificidad = \frac{VN}{VN + FP}$$



Métricas de Evaluación de Clasificación

Matriz de Confusión



Calcular:

$$tasa\ de\ aciertos = \frac{70 + 50}{150} = 0.800$$

$$precisión = \frac{70}{70 + 10} = 0.875$$

$$sensibilidad = \frac{70}{70 + 20} = 0.778$$

$$especificidad = \frac{50}{50 + 10} = 0.833$$

Métricas de Evaluación de Clasificación

Matriz de Confusión

Predicción \ Real	(+) 1	(-) 0
(+) 1	30	25
(-) 0	20	925

Target altamente desbalanceado.
Clase (+) representa el 5% de toda la base

Calcular:

$$especificidad = \frac{925}{925 + 25} = 0.974$$

$$sensibilidad = \frac{30}{30 + 20} = 0.600$$

$$precisión = \frac{30}{30 + 25} = 0.545$$

$$\boldsymbol{tasa\ de\ aciertos = \frac{30 + 925}{1000} = 0.955}$$

Modelo excelente?

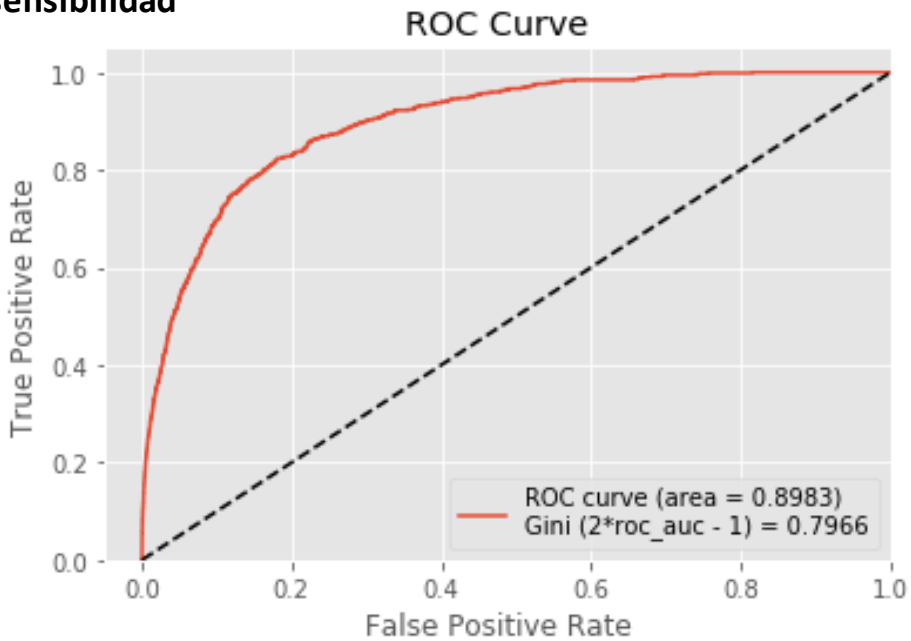
Métricas de Evaluación de Clasificación

ROC - AUC

La Curva ROC proporciona un índice de la capacidad de un modelo para discriminar entre estados alternativos de la clase target.

Es útil para comparar modelos y seleccionar umbrales de decisión (puntos de corte entre (+) y (-))

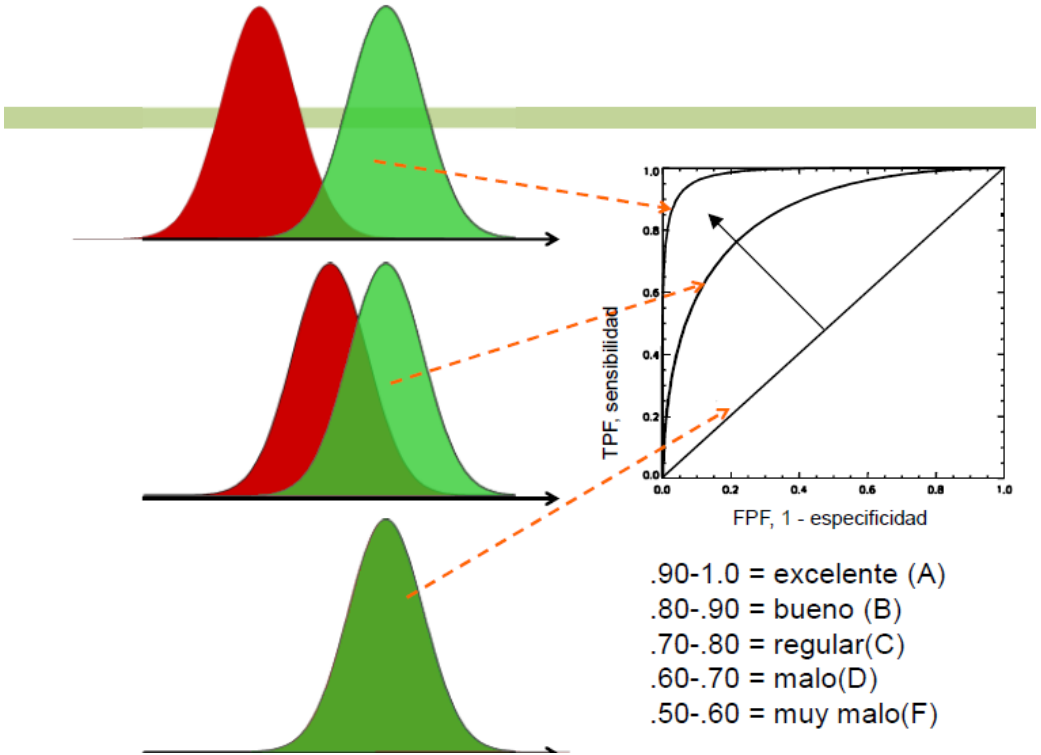
sensibilidad



1 - especificidad

Métricas de Evaluación de Clasificación

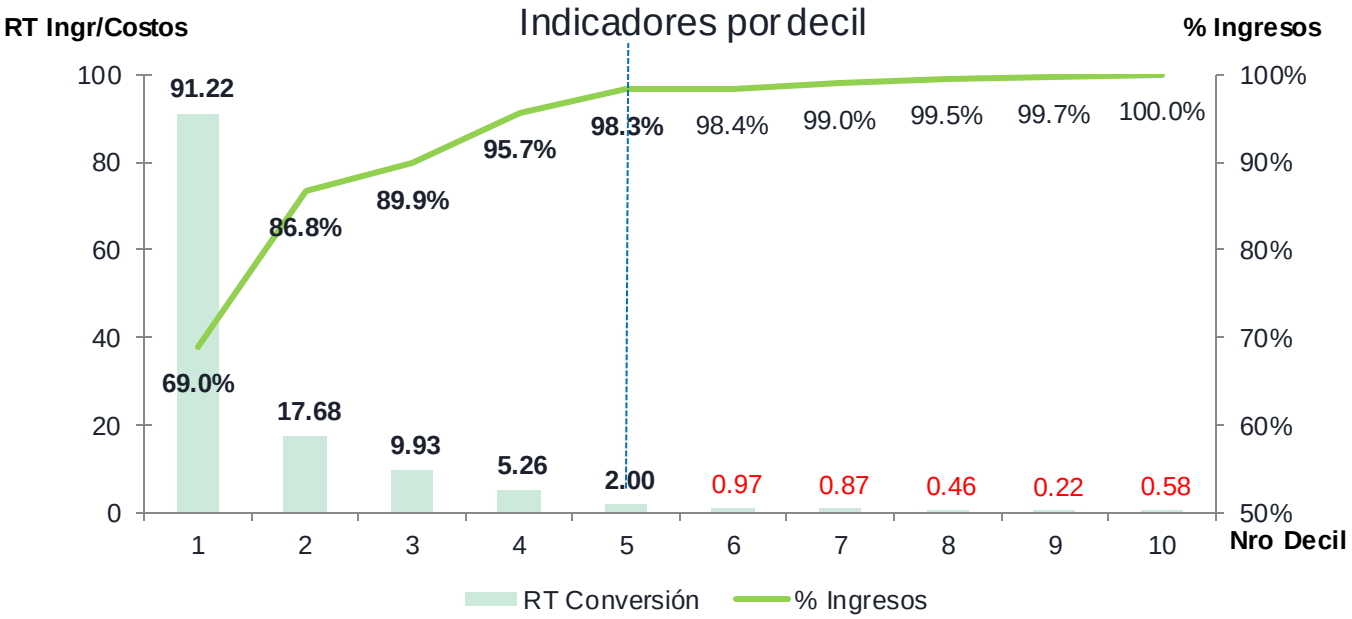
ROC - AUC



.90-1.0 = excelente (A)
.80-.90 = bueno (B)
.70-.80 = regular(C)
.60-.70 = malo(D)
.50-.60 = muy malo(F)

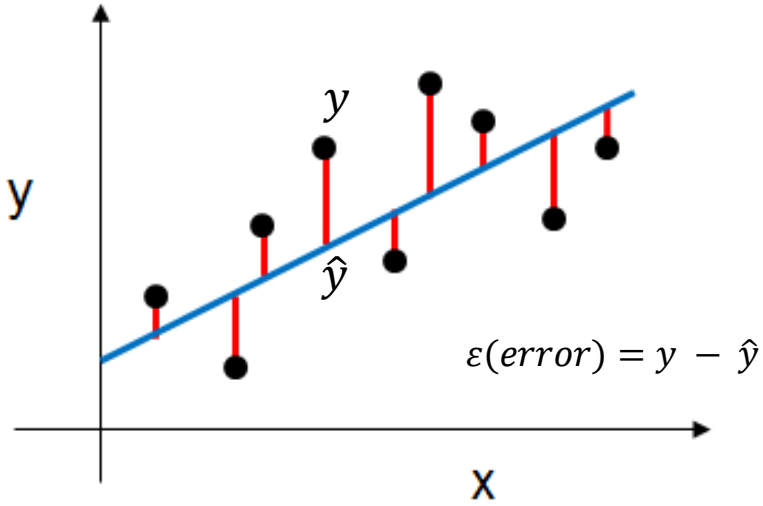
Métricas de Evaluación de Clasificación

Resultados Modelo de Propensión de Compra



Se dividió la base por deciles según los resultados del modelo. Los primeros 5 grupos generaron el 98% de los ingresos totales de la campaña feb-17.

Métricas de Evaluación de Regresión



Error Absoluto Medio:

$$MAE = \frac{\sum_{t=1}^{n} |\hat{y}_t - y_t|}{n}$$

Error Cuadrático Medio:

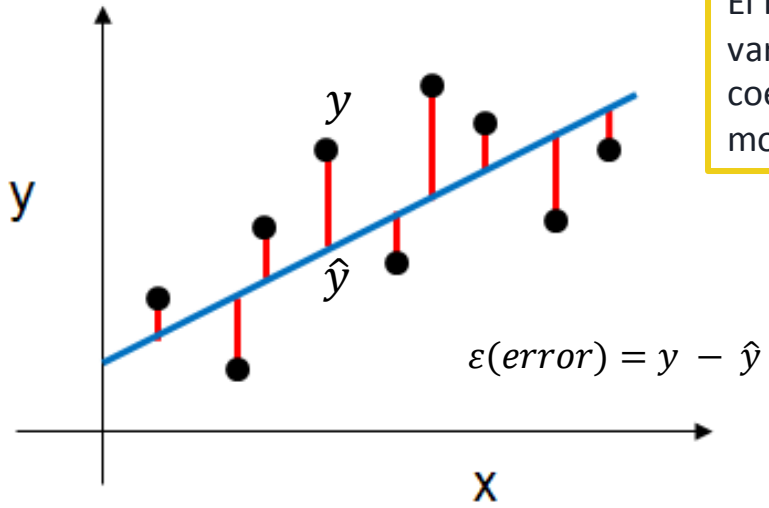
$$MSE(y, \hat{y}) = \frac{1}{n} \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2$$

Raiz Cuadrada del Error Cuadrático Medio:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} (\hat{y}_t - y_t)^2}{n}}$$

Métricas de Evaluación de Regresión

Coeficiente de Determinación, R2



El R Cuadrado se define como la proporción de la varianza total de la variable explicada por la regresión. El R Cuadrado, también llamado coeficiente de determinación, refleja la bondad del ajuste de un modelo a la variable que pretender explicar.

El cociente R2 = SCM/SCT = 1 − SCR/SCT es la proporción de variacion de las respuestas explicadas por la regresión; se conoce como c**oeficiente de determinación.**

Ej: si R 2 = 0.85, la variable x explica un 85% de la variación de la variable y