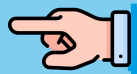
Aprendizaje Supervisado - Clasificación

Clase sincrónica

{desafío} latam_

Implementar modelos de aprendizaje automático por medio de técnicas estadísticas, adecuando los diferentes algoritmos debidamente a la situación y requerimientos necesarios

- Unidad 1: Introducción al Machine Learning
- Unidad 2: Aprendizaje Supervisado y No Supervisado
  (Parte I: No supervisado)
  (Parte II: Clasificación)
  (Parte III: Clasificación)
  (Parte IV: Regresión)
  (Parte V: Series de tiempo)
- Unidad 3: Aplicando lo aprendido
  (Parte I: Preprocesamiento de datos)
  (Parte II: Modelamiento)



Te encuentras aquí

{desafío} latam_

¿Qué aprenderás en esta sesión?

Desarrollar modelos de aprendizaje supervisado orientado a la clasificación, por medio del ajuste de hiperparámetros la utilización de diferentes algoritmos y su evaluación según las métricas de desempeño adecuadas.

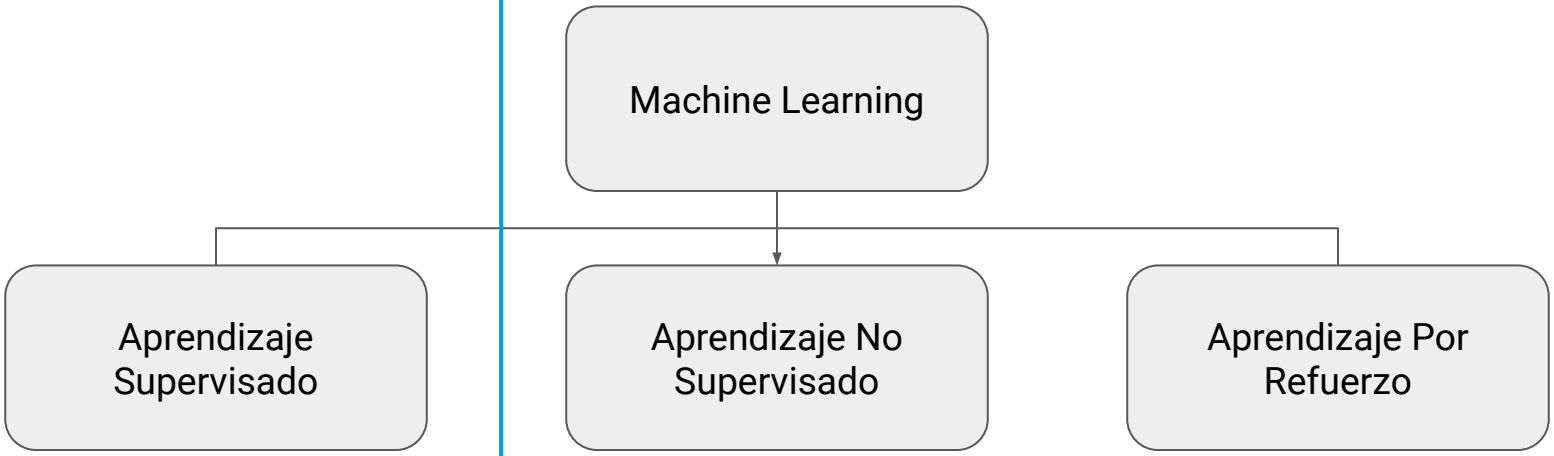
{desafío} latam_

¿Qué son los algoritmos de clustering? ¿Cuáles conoces?

¿Como se puede validar la calidad de los clusters generados?
¿Qué es el método del codo?

/* Aprendizaje Supervisado */

Aprendizaje Supervisado



El aprendizaje supervisado se refiere a la capacitación de modelos utilizando datos de entrenamiento etiquetados.

- **Clasificación**
- **Regresión**

/* Clasificación */

Clasificación

¿Qué es?

A partir de una serie de variables numéricas se llega a una categórica (puede codificarse en 1/0)

Variables Predictoras:
Se utilizan para predecir la variable objetivo

radio	textura	perimetro	area	maligno
17.930	24.48	115.20	998.9	0
11.540	14.44	74.65	402.9	1
8.878	15.49	56.74	241.0	1
11.460	18.16	73.59	403.1	1
14.250	21.72	93.63	633.0	0
18.010	20.56	118.40	1007.0	0
17.950	20.01	114.20	982.0	0

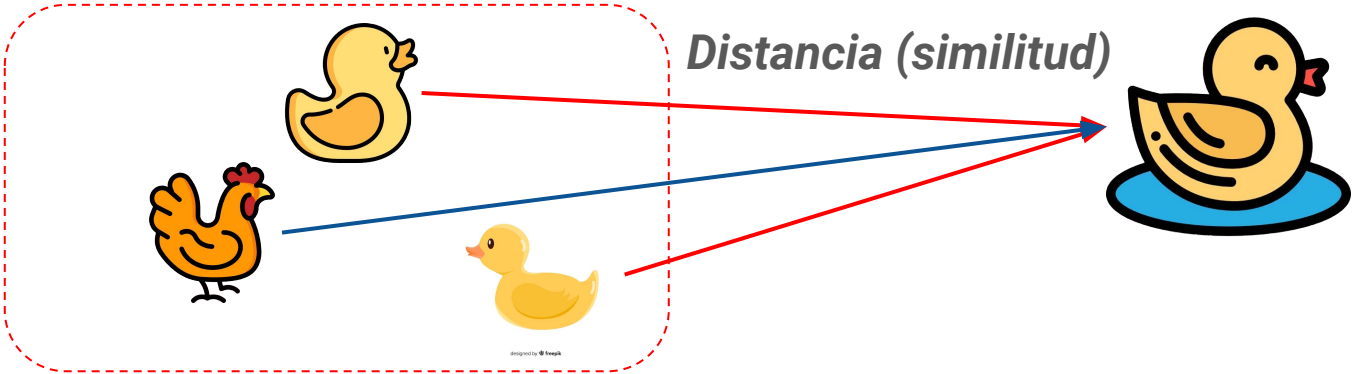
Variable Objetivo:
Categórica (Maligno o No Maligno)

/* K Nearest Neighbors - KNN*/

KNN

K Nearest Neighbors

"Dime con quién andas y te diré quien eres"



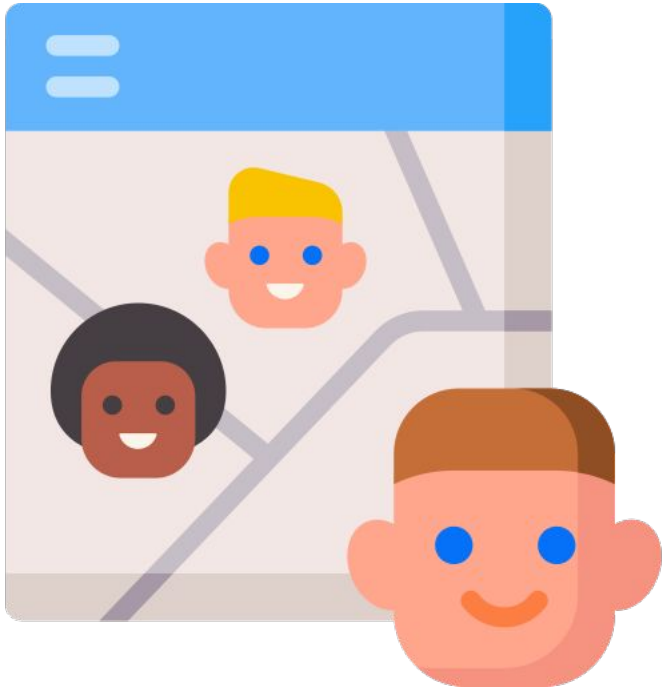
Fuente: flaticon.es

KNN

¿Cómo se aplica?

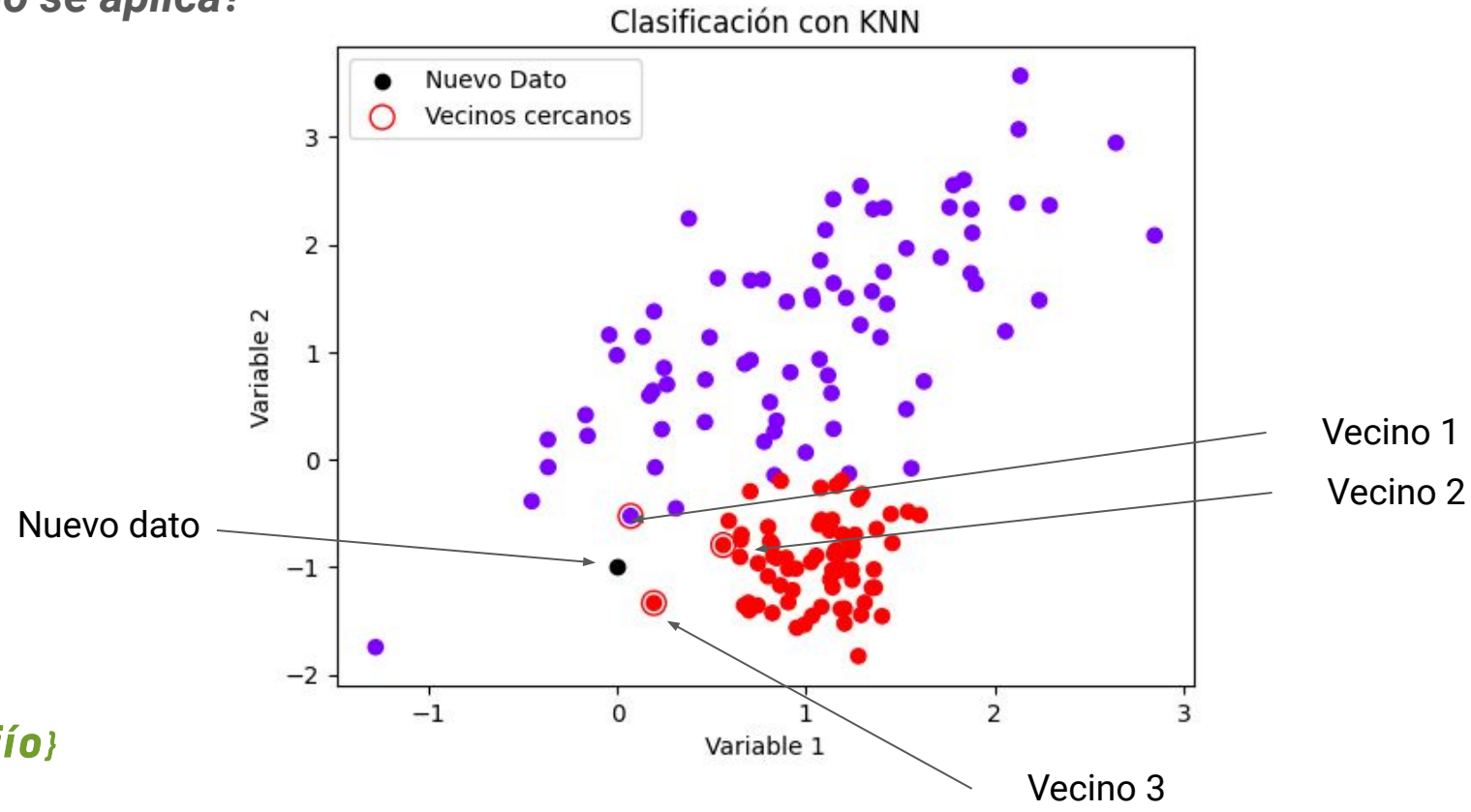
Votación de la mayoría dentro de los k vecinos más cercanos

1. Se utiliza el set de entrenamiento como ejemplos de prueba.
2. Se necesita una medida de distancia entre los elementos
3. Se necesita conocer los k valores vecinos para comparar.



KNN

¿Cómo se aplica?



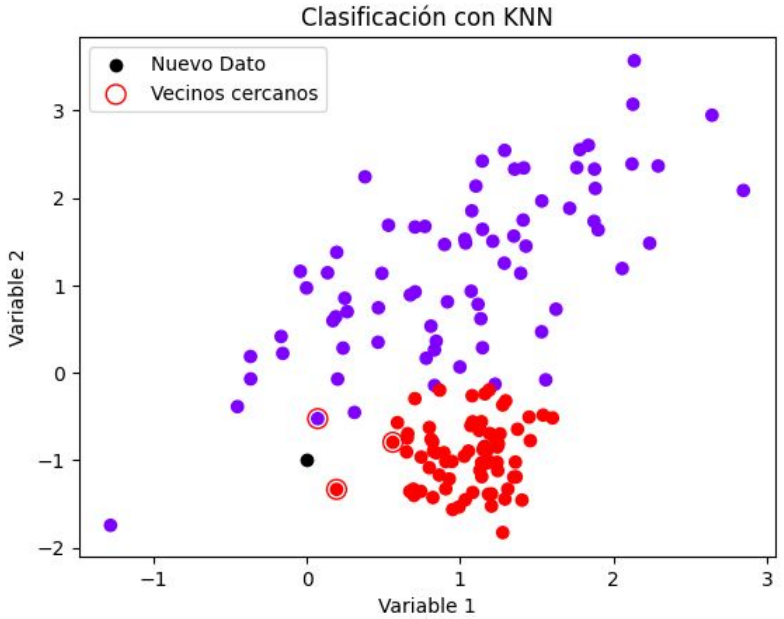
{desafío} latam_

KNN

¿Cómo se aplica?

1. Al predecir un nuevo registro se calcula la distancia con el set de entrenamiento.
2. Se identifican los k registros más cercanos.
3. Se etiqueta con la clase que se la mayoría de los k registros cercanos.

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$$



{desafío} latam_

KNN
Ventajas y desventajas

Ventajas

- Fácil de entender e implementar.
- No hace suposiciones sobre la distribución de los datos.
- Es adecuado para problemas de clasificación multi clase.
- Puede funcionar bien en conjuntos de datos pequeños o con pocos atributos.

Desventajas

- Es computacionalmente costoso (conjuntos grandes).
- Sensible a la escala de las características, por lo que es importante realizar una normalización adecuada.
- La elección del valor de k puede ser crucial.

/* Métricas de desempeño */

Métricas de desempeño

¿Cómo evaluar un modelo?

Las métricas de desempeño son medidas utilizadas para evaluar el rendimiento de un modelo de clasificación. Un primer paso es analizar los aciertos y fallos que obtiene, considerando las diferentes clases presentes en el conjunto.

Estos datos se registran en la **matriz de confusión**.

Matriz de Confusión

		Valor Predicho	
Valor Real		Class = Yes	Class = No
	Class = Yes	a (TP)	b (FN)
	Class = No	c (FP)	d (TN)

- TP = Verdaderos positivos (True positives)
- FN = Falsos Negativos (False Negatives)
- FP = Falsos Positivos (False Positives)
- TN = Verdaderos Negativos (True Negatives)

Matriz de confusión

Métricas asociadas - Accuracy

A partir de la matriz, podemos definir algunas métricas de desempeño.

		Valor Predicho	
		Class = Yes	Class = No
Valor Real	Class = Yes	a (TP)	b (FN)
	Class = No	c (FP)	d (TN)

Accuracy: Es la exactitud global del modelo corresponde a la proporción de datos que fueron correctamente clasificados, independiente de la categoría, por eso se considera la métrica de exactitud global.

$$accuracy = \frac{a+d}{a+b+c+d}$$

Matriz de confusión

Métricas asociadas - Precision

A partir de la matriz, podemos definir algunas métricas de desempeño.

		Valor Predicho	
		Class = Yes	Class = No
Valor Real	Class = Yes	a (TP)	b (FN)
	Class = No	c (FP)	d (TN)

Precision: es la proporción de ejemplos clasificados correctamente como positivos (verdaderos positivos) en relación con todos los ejemplos clasificados como positivos (verdaderos positivos y falsos positivos). Se puede entender como la capacidad del modelo para identificar correctamente los positivos.

$$precision = \frac{a}{a+c}$$

Matriz de confusión

Métricas asociadas - Recall

A partir de la matriz, podemos definir algunas métricas de desempeño.

		Valor Predicho	
		Class = Yes	Class = No
Valor Real	Class = Yes	a (TP)	b (FN)
	Class = No	c (FP)	d (TN)

Recall (Sensibilidad): es la proporción de ejemplos clasificados correctamente como positivos (verdaderos positivos) en relación con todos los ejemplos reales positivos (verdaderos positivos y falsos negativos). Se puede entender como la capacidad del modelo para detectar correctamente los positivos.

$$recall = \frac{a}{a+b}$$

Matriz de confusión

Métricas asociadas - f1

A partir de la matriz, podemos definir algunas métricas de desempeño.

		Valor Predicho	
		Class = Yes	Class = No
Valor Real	Class = Yes	a (TP)	b (FN)
	Class = No	c (FP)	d (TN)

Valor F1 (F1 Score): es una métrica que combina la precisión y la sensibilidad en una sola medida. Es útil cuando se busca un equilibrio entre la precisión y la sensibilidad, ya que tiene en cuenta tanto los falsos positivos como los falsos negativos.

$$f1 = \frac{2rp}{r+p}$$

¡Manos a la obra! KNN y Métricas de desempeño

{desafío} latam_

¡Manos a la obra!

Métricas de desempeño

Veremos cómo aplicar la clasificación KNN con Python y evaluar su desempeño, para lo que puedes abrir un archivo de Jupyter Notebook y replicar los pasos que te irá presentando tu profesor. En esta presentación abordaremos:

1. El algoritmo KNN en acción
2. Métricas de desempeño

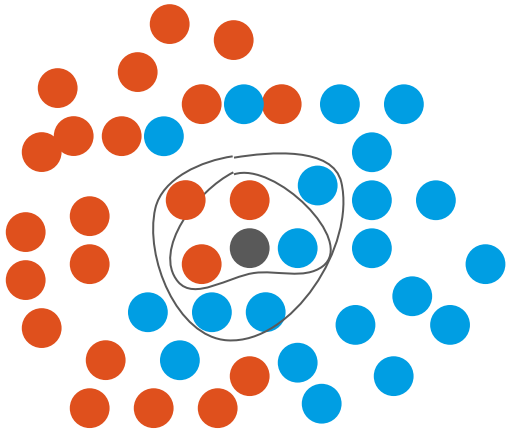
{desafío} latam_

¿Qué ocurre con las métricas en un dataset desbalanceado? ¿Qué métrica es más relevante?

KNN

Hiperparametros (k vecinos)

k = 3 (rojo)



k = 7 (azul)

¿Cómo escoger un valor óptimo?

/* Validación cruzada - Cross Validation*/

Cross Validation

Funcionamiento

Se divide el conjunto de datos disponibles en “k” subconjuntos de entrenamiento y prueba, y se realizan varios experimentos en los que se testea con cada uno de los subconjuntos, en cada iteración.

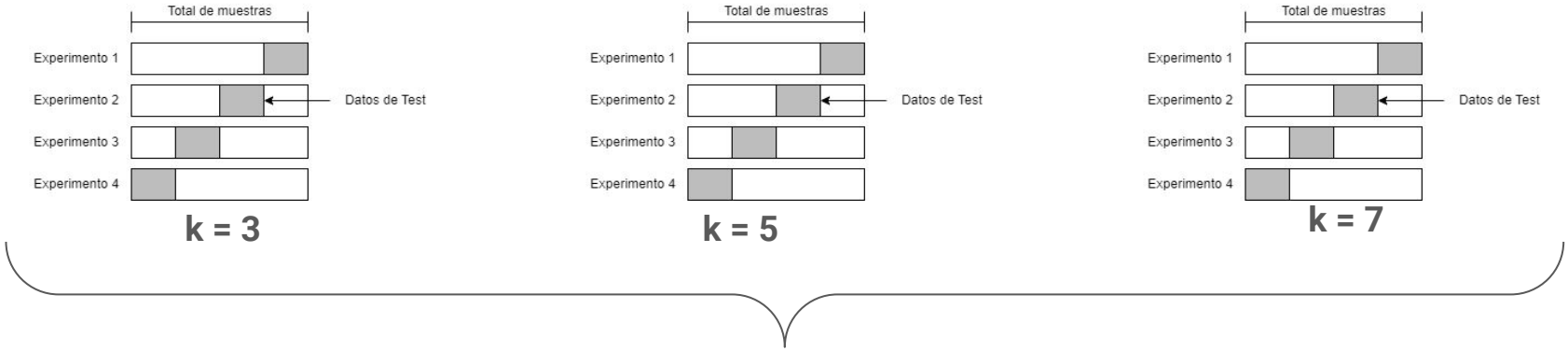
Cross Validation

Funcionamiento

1. Se entrena el algoritmo de esta forma para cada combinación de hiperparametros.
2. Se elige la combinación de hiperparametros que mejor resultado tiene.

Cross Validation

Funcionamiento



Se calculan métricas de desempeño con cada combinación y se elige la que mejores métricas tiene.

¡Manos a la obra!
Cross Validation y valor de k

¡Manos a la obra!

Cross Validation y valor de K

Veremos cómo realizar validación cruzada y escoger, a partir de ello, un valor adecuado para K en Python, para lo que puedes abrir un archivo de Jupyter Notebook y replicar los pasos que te irá presentando tu profesor. En esta presentación abordaremos:

1. Validación cruzada con KNN
2. Validación cruzada y valor de K.

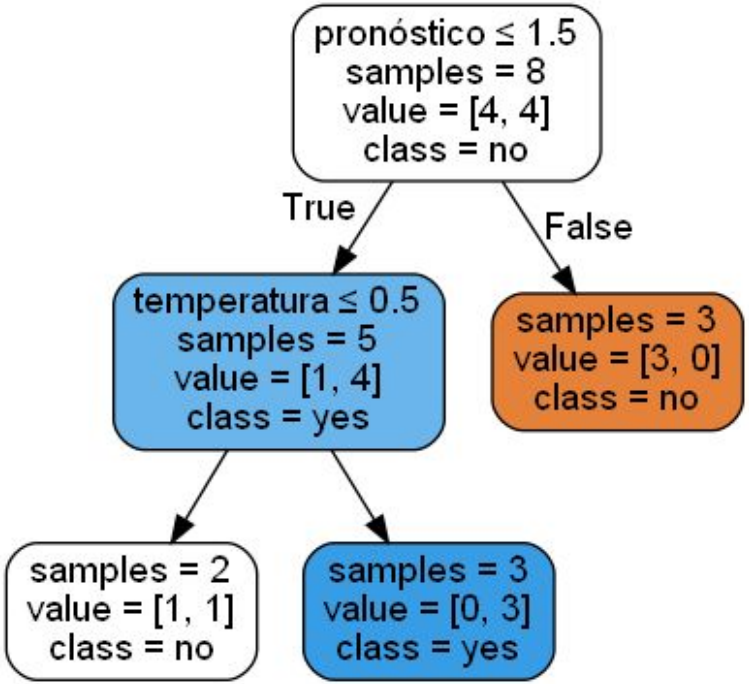
{desafío} latam_

/* Árboles de Decisión*/

Árboles de decisión

¿Qué son?

Método que busca particionar el espacio de atributos en una serie de rectángulos y posteriormente se implementa un modelo simple (o estadístico) de representación [Definición de Hastie et al. 2009]



Árboles de decisión

Hiperparámetros

Los hiperparámetros en un árbol de decisión buscan controlar la tendencia de crecer de manera irrestricta:

- ¿Hasta qué punto puedo dejar crecer un árbol?
- ¿Cuántos datos son suficientes en cada nodo para particionar o declararlo terminal?
- ¿Cuántos atributos son suficientes para que mi árbol pueda capturar de buena manera el fenómeno?

Árboles de decisión

Hiperparámetros y características

Máximo de Profundidad

¿Hasta qué niveles puede crecer un árbol?

Cantidad de atributos

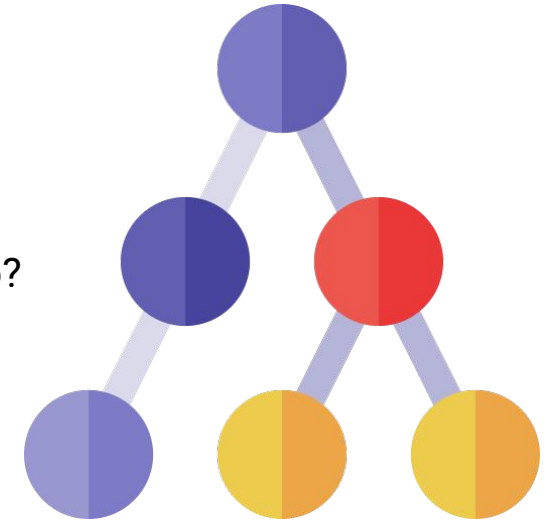
¿Cuántos atributos debemos considerar en un árbol?

Mínimo de muestras en un nodo particionable

¿Con cuántas observaciones podemos seguir subdividiendo?

Mínimo de muestras en un nodo terminal

¿Con cuántas observaciones dejamos de subdividir?



Desafío - Predicción de fuga de clientes

{desafío}
latam_

Desafío

"Predicción de fuga de clientes"

- Descarga el archivo "Desafío".
- Tiempo de desarrollo asincrónico: desde 4 horas.
- Tipo de desafío: individual.

¡AHORA TE TOCA A TI! 💪🏽

{desafío} latam_

Ideas fuerza

Clasificación *es una técnica de aprendizaje supervisado donde la* ***variable objetivo es categórica.***

Algunos de los algoritmos que vimos en clases son KNN, árbol de decisión y naive bayes. Cada uno tiene sus ventajas y desventajas propias. Aunque existen muchos algoritmos más.

La ***Matriz de confusión*** *es importante para ver el desempeño de los modelos de clasificación y de esta se desprenden varias métricas como* ***accuracy, recall, precision y f1.***

¿Qué conceptos no te quedaron claros o quieres reforzar?

Recursos asincrónicos

¡No olvides revisarlos!

Para esta semana deberás revisar:

- Guía de estudio
- Desafío “Predicción de fuga de clientes”

{desafío} latam_

Próxima sesión...

- *Aplicarás técnicas y herramientas de regresión logística para la elaboración de modelos, ajustando sus parámetros cuando corresponda.*

{desafío}
latam_

Academia de
talentos digitales