EGADE Business School
Tecnológico de Monterrey

# Aplicaciones de analítica de datos a los negocios II

PROF: JUAN C. BUSTAMANTE

JUCBUSTAM@TEC.MX

# Normas para la conexión síncrona:

1. La clases tiene un back-up garantizado (Grabación disponible en la nube de Zoom).
2. Ingresar a la clase con la cámara del equipo de computo encendida.
3. La **cámara deberá permanecer encendida a lo largo de la clase**.
4. Al ingresar a la clase deben silenciar el micrófono del equipo de computo.
5. Levantar la mano es una opción cuando se quiere preguntar algo durante la sesión de clase, pero les recomiendo que mejor hagamos uso intensivo del chat del canal general para hacer preguntas**.**
6. En caso de necesitar hacer una pregunta, puede interrumpir la clase sin problema, activando el micrófono de vuestro equipo de computo, luego de la pregunta desactívelo nuevamente**.**
7. Para una buena clase online es indispensable **el debate, así que foméntelo!!!**.
8. Toda la información se gestiona en CANVAS LMS.

EGADE Business School
Tecnológico de Monterrey

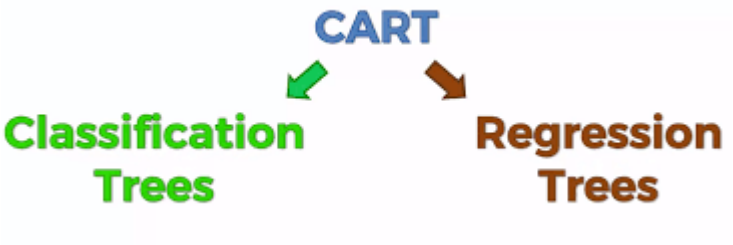
# Cronograma de trabajo:

| Sesiones | Contenidos | Actividad | | Fecha |
|---|---|---|---|---|
| 1 | Información general del curso | Utility of classification algorithms | | Martes 18/04 |
| 2 | Algoritmo de regresión logística | Ejecutar script | **Solución caso:** Retention modelling at Scholastic Travel Company (A) and (B) | Martes 25/04 |
| 3 | Algoritmo Naïve Bayes | Ejecutar script | | Martes 02/05 |
| 4 | Algoritmo k-nearest-neighbors (KNN) | Ejecutar script | | Martes 09/05 |
| 5 | Algoritmo Support vector machine | Ejecutar script | | Martes 16/05 |
| 6 | Algoritmo Decision Trees | Ejecutar script | | Martes 23/05 |
| 7 | Algoritmo Random Forest | Ejecutar script | | Martes 30/05 |
| 8 | Modelo RFM | Ejecutar script | **Solución caso**: CD Now | Martes 06/06 |
| 9 | Modelo valor de vida del cliente (I) | Ejecutar script | | Martes 13/06 |
| 10 | Modelo valor de vida del cliente (II) | Ejecutar script | | Martes 20/06 |
| 11 | Análisis de series de tiempo | Ejecutar script | | Martes 27/06 |
| 12 | **Proyecto final** | Presentación en equipos | | Martes 04/07 |
| | **Evaluación final** | | | |

# 1

# Classification and Regression Trees (CART)

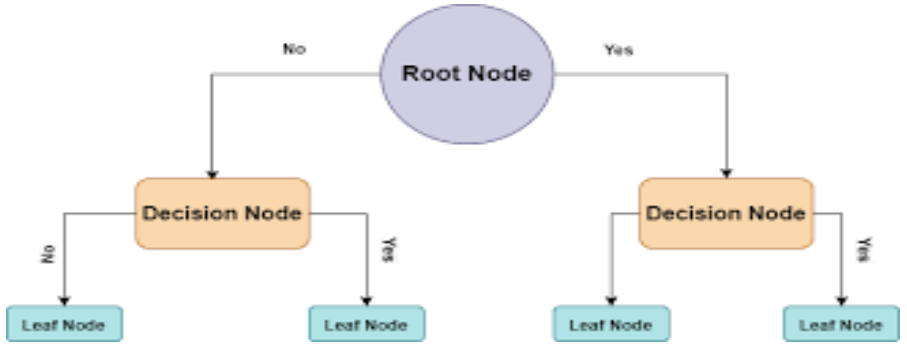
# Classification and Decision Trees
# CART




# Classification and Decision Trees CART

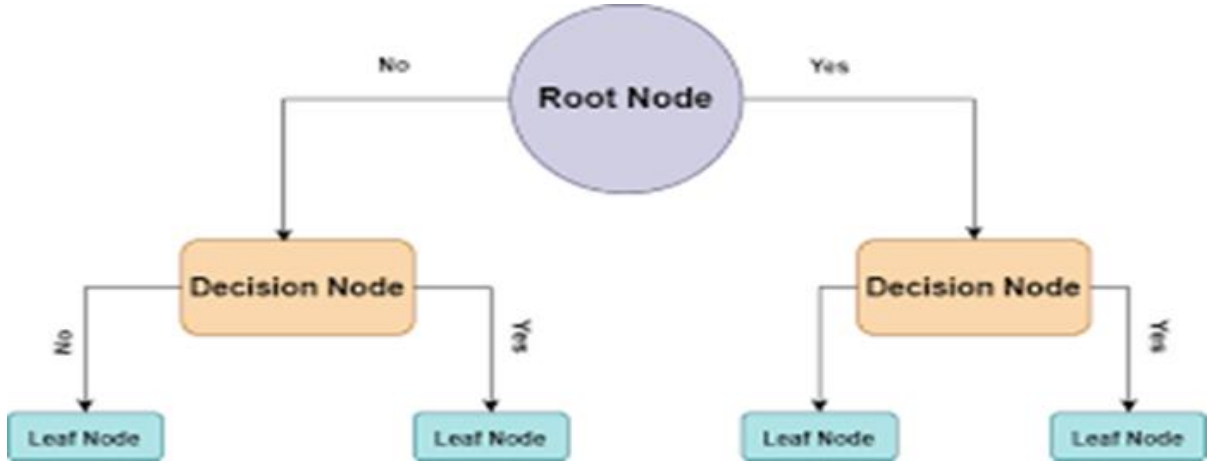
Los árboles de decisión son un método usado en distintas disciplinas como modelo de predicción. Estos son similares a diagramas de flujo, en los que llegamos a puntos en los que se toman decisiones de acuerdo a una regla.

En el campo del aprendizaje automático, hay distintas maneras de obtener árboles de decisión. La que usaremos en esta ocasión es conocida como **CART**: **Classification And Regression Trees**. Esta es una técnica de aprendizaje supervisado. Tenemos una variable objetivo (dependiente) y nuestra meta es obtener una **función** que nos permita predecir, a partir de variables predictoras (independientes), el valor de la variable objetivo para casos desconocidos.



# CART: Cómo funciona?

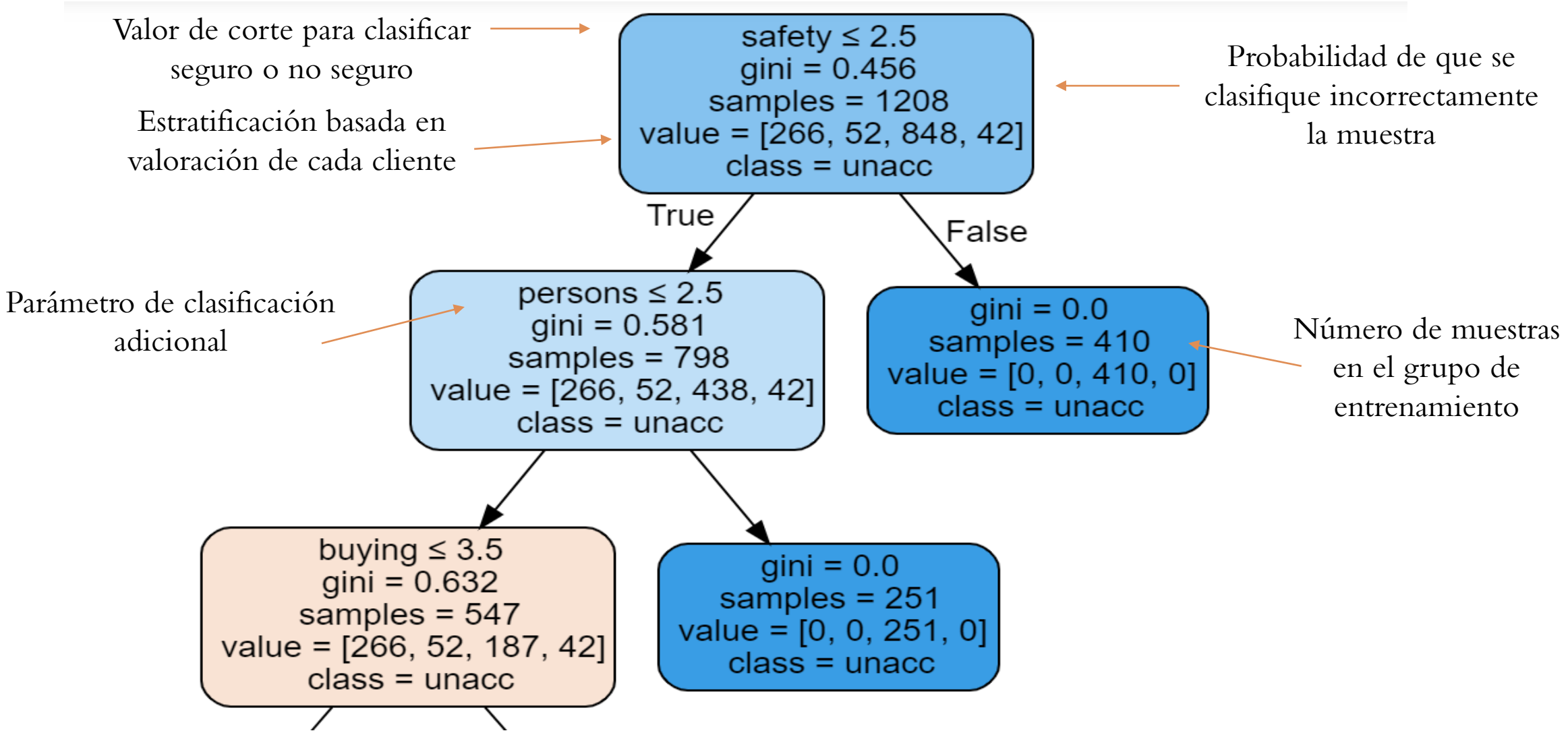
De manera general, lo que hace este algoritmo es encontrar la variable independiente que mejor separa nuestros datos en grupos, que corresponden con las categorías de la variable objetivo. Esta mejor separación es expresada con una regla. A cada regla corresponde un nodo.



# CART: Cómo funciona? Detalle

- Supongamos que nuestra variable objetivo tiene dos niveles, **comprar** y **no comprar**
- El algoritmo encuentra que la variable que mejor separa los datos es **ingreso mensual**, y la regla resultante es que ingreso mensual > *X pesos*.
- Una vez hecho esto, los datos son separados en grupos a partir de la regla obtenida. Después, para cada uno de los grupos resultantes, se repite el mismo proceso.
- El algoritmo busca la variable que mejor separa los datos en grupos, se obtiene una regla, y se separan los datos.
- El proceso se repite de manera **recursiva** hasta que no es imposible obtener una mejor separación. Cuando esto ocurre, el algoritmo se detiene.
- Cuando un grupo no puede ser partido mejor, se le llama **nodo terminal** u **hoja**.
- Una **característica muy importante** en este algoritmo es que una vez que alguna variable ha sido elegida para separar los datos, ya no es usada de nuevo en los grupos que ha creado. Se buscan variables distintas que mejoren la separación de los datos.

# División de un árbol de decisión



# CART (SELECCIÓN DE ATRIBUTOS)

Las medidas de selección de atributos más populares son:

- Information gain
- Gini index

Criterio information gain → los atributos son categóricos
Criterio Gini Index → los atributos son continuos.

1. Para entender el concepto de information gain, necesitamos conocer otro concepto llamado **Entropía.**

> La entropía es la cantidad de aleatoriedad en los datos. La entropía en un nodo depende de la cantidad de datos aleatorios que se encuentran en ese nodo y se debe calcular para cada nodo

$$Entropy = \sum_{i=1}^{C} -p_i * \log_2(p_i)$$

Se busca que en los árboles tengan la entropía más pequeña en sus nodos. E = 0 muestras homogéneas; E = 1 muestras divididas equitativamente

# CART (SELECCIÓN DE ATRIBUTOS)

## Gini index

El índice de Gini mide el grado de pureza de un nodo. Mide la probabilidad de no sacar dos registros de la misma clase del nodo.

A tener en cuenta:

- ✓ A mayor índice de Gini menor pureza, por lo que seleccionaremos la variable con menor Gini ponderado.
- ✓ Suele seleccionar divisiones desbalanceadas, donde normalmente aísla en un nodo una clase mayoritaria y el resto de clases los clasifica en otros nodos.

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

# CART

## VENTAJAS

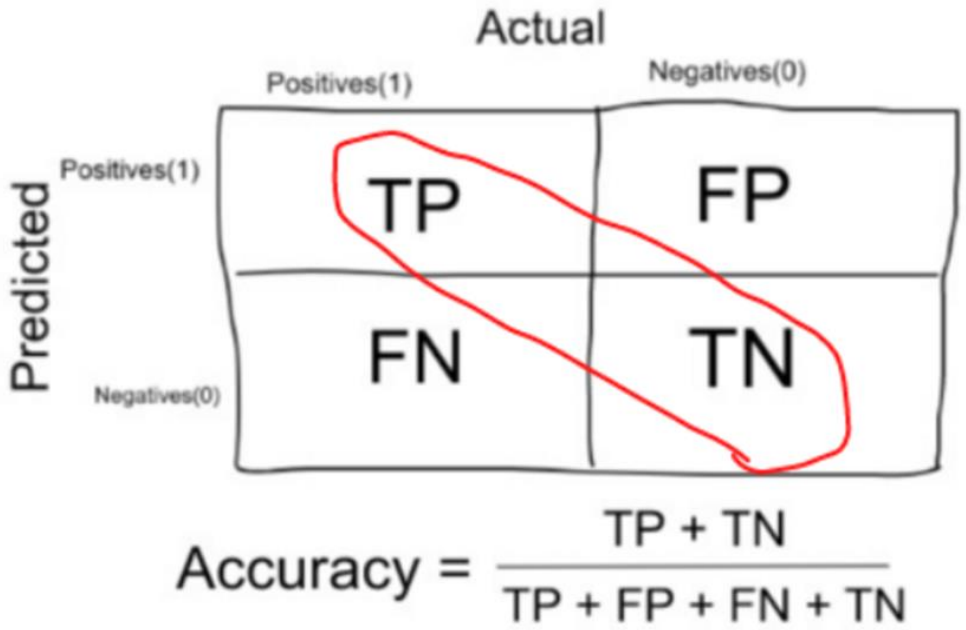
- Son fáciles de construir, interpretar y visualizar.
- Selecciona las variables más importantes y en su creación no siempre se hace uso de todos los predictores.
- Si faltan datos no podremos recorrer el árbol hasta un nodo terminal, pero sí podemos hacer predicciones promediando las hojas del sub-árbol que alcancemos.
- No es preciso que se cumplan una serie de supuestos como en la regresión lineal (linealidad, normalidad de los residuos, homogeneidad de la varianza, etc.).
- ✓ Permiten relaciones no lineales entre las variables explicativas y la variable dependiente.

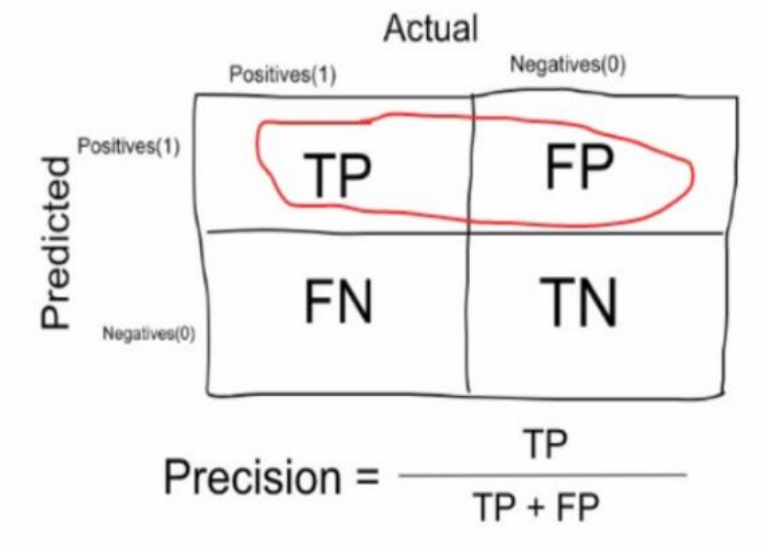
## DESVENTAJAS

- Tienden al sobreajuste u *overfitting* de los datos, por lo que el modelo al predecir nuevos casos no estima con el mismo índice de acierto.
- Se ven influenciadas por los *outliers*, creando árboles con ramas muy profundas que no predicen bien para nuevos casos. Se deben eliminar dichos *outliers*.
- No suelen ser muy eficientes con modelos de regresión.
- Crear árboles demasiado complejos puede conllevar que no se adapten bien a los nuevos datos. La complejidad resta capacidad de interpretación.

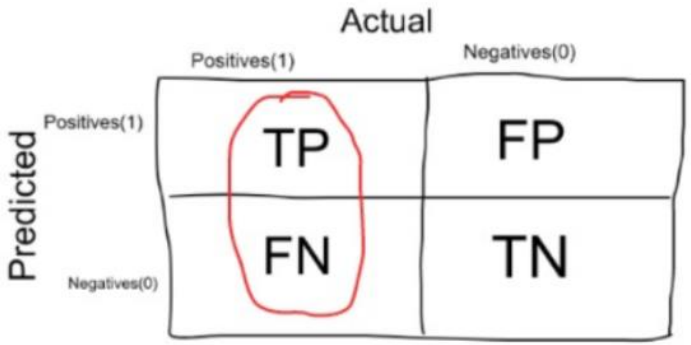
# Métricas

1. **True Positives (TP):** cuando la clase real del punto de datos era 1 (Verdadero) y la predicha es también 1 (Verdadero)
2. **Verdaderos Negativos (TN):** cuando la clase real del punto de datos fue 0 (Falso) y el pronosticado también es 0 (Falso).
3. **False Positives (FP):** cuando la clase real del punto de datos era 0 (False) y el pronosticado es 1 (True).
4. **False Negatives (FN):** Cuando la clase real del punto de datos era 1 (Verdadero) y el valor predicho es 0 (Falso).

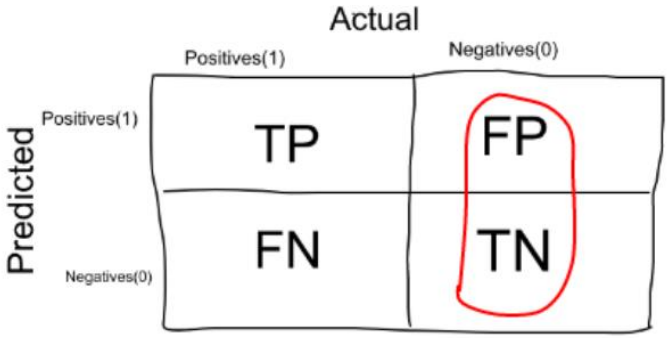





$$\text{Precision} = \frac{TP}{TP + FP}$$




$$\text{Recall} = \frac{TP}{TP + FN}$$




$$\text{Specificity} = \frac{TN}{TN + FP}$$

# Métricas

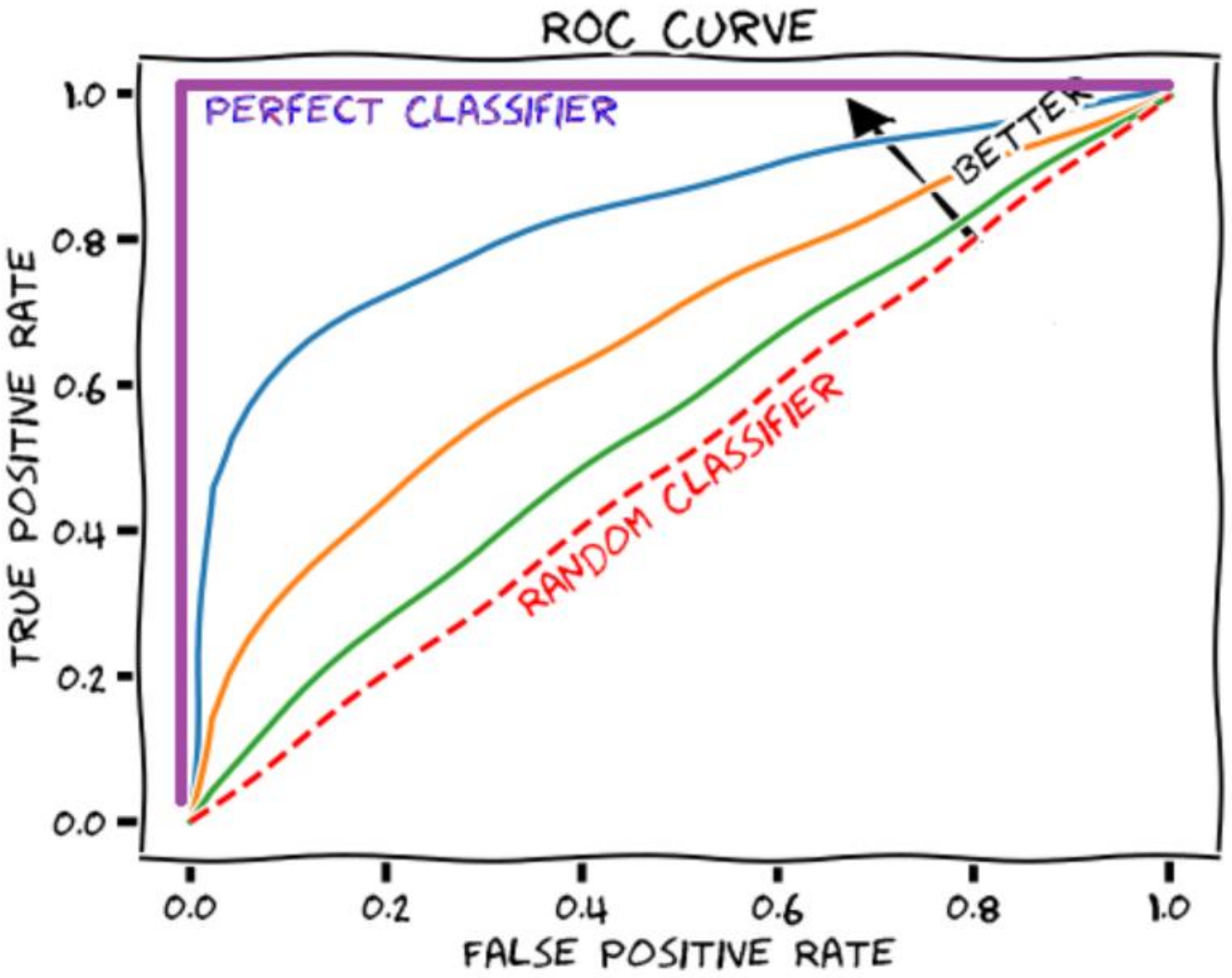
# Métricas

**Curva ROC**

ROC es un acrónimo para Receiver Operating Characteristic (Característica Operativa del Receptor). Es una gráfica que enfrenta el ratio de falsos positivos (eje x) con el ratio de falsos negativos (eje y).

La curva ROC es útil por dos principales motivos:

- Permite comparar diferentes modelos para identificar cual otorga mejor rendimiento como clasificador
- El área debajo de la curva (AUC) puede ser utilizado como resumen de la calidad del modelo



# Bibliotecas


Manipulación y análisis de datos


Creación de vectores y matrices
Colección de funciones matemáticas


Generación de gráficos


Biblioteca de visualización basada en Matplotlib


Biblioteca de aprendizaje automático