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			

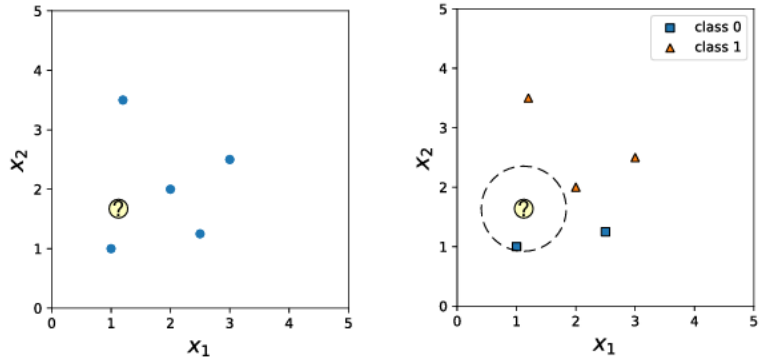
1

〈 K-NN Vecinos próximos.

K-Nearest Neighbors

Algoritmo K-NN Vecinos más cercanos

El algoritmo de k vecinos más cercanos, también conocido como KNN o k-NN, es un clasificador de aprendizaje supervisado no paramétrico, que utiliza la proximidad para hacer clasificaciones o predicciones sobre la agrupación de un punto de datos individual. Si bien se puede usar para problemas de regresión o clasificación, generalmente se usa como un algoritmo de clasificación, partiendo de la suposición de que se pueden encontrar puntos similares cerca uno del otro.



Características:

• El algoritmo k-nn es uno de los algoritmos de aprendizaje automático que es muy fácil de entender y funciona increíblemente bien en la práctica.

• Es un algoritmo no paramétrico. *No paramétrico significa que el algoritmo no hace suposiciones sobre la distribución de probabilidad de los datos de la muestra.*

• El algoritmo k-nn toma su nombre del hecho de que usa información sobre los k vecinos más cercanos de un ejemplo dado para clasificar ejemplos no etiquetados.

• La letra k es un término variable que implica que se podría usar cualquier número de vecinos más cercanos

Características:

Después de elegir k

- El algoritmo requiere un conjunto de datos de entrenamiento compuesto por ejemplos que ya están clasificados en varias categorías, según la etiqueta de una variable categórica (Y).
- Para cada registro no etiquetado, k-nn identifica los registros k en la data de entrenamiento que son los “más cercanos” en similitud.
- Al registro no etiquetado se le asigna la clase de la mayoría de los k vecinos más cercanos.

Tipos de distancia

- Distancia euclidiana:

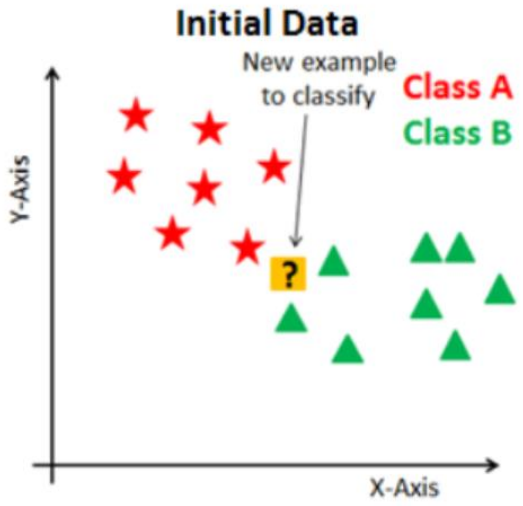
$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

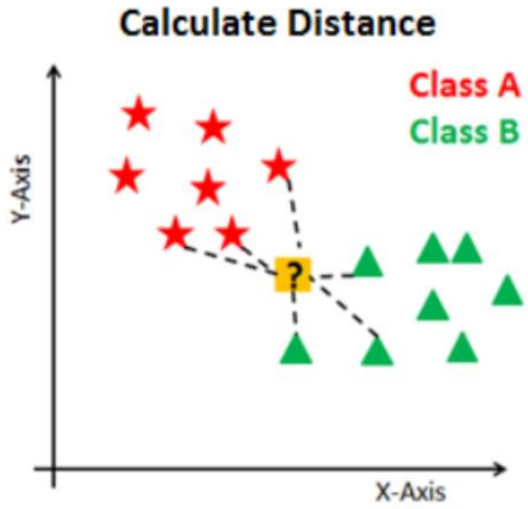
- Distancia manhattan:

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

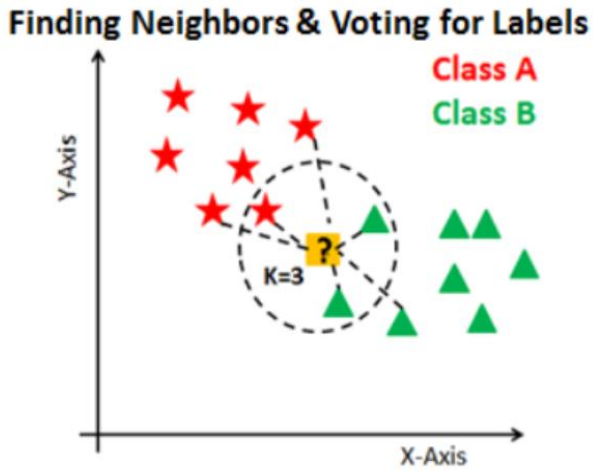
- Distancia Minkowsky:

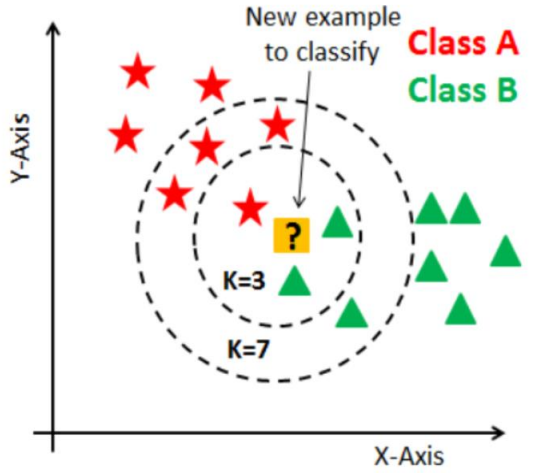
$$(\sum_{i=1}^{k}(|x_i - y_i|)^q)^{\frac{1}{q}}$$





¿CÓMO FUNCIONA?:





Elección de K:

- La decisión de cuántos vecinos usar para k-nn determina qué tan bien el modelo generalizará para futuros datos.
- El balance entre el overfitting y el underfitting de los datos de entrenamiento es un problema conocido como bias-variance tradeoff.
- La elección de un k grande reduce el impacto o la varianza causada por la data con ruido, pero puede sesgar el aprendizaje con el riesgo de ignorar patrones pequeños pero importantes.
- Suponiendo que se elija un k tan grande como el número total de observaciones en los datos de entrenamiento. Con cada instancia de entrenamiento representado en la votación final, la clase más común siempre tiene la mayoría de votos. Por lo tanto, el modelo siempre predeciría la clase mayoritaria, independientemente de los vecinos más cercanos.
- En el extremo opuesto, el uso de un k=1 permite la data con ruido u outliers que influyen indebidamente en la clasificación de ejemplos.

K-NN Vecino más próximo

Ventajas:

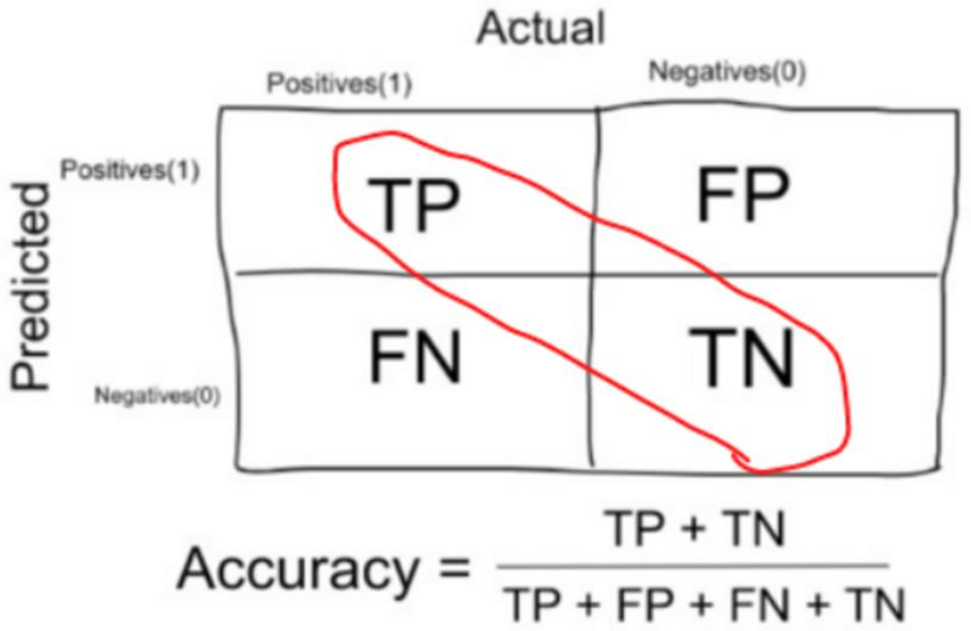
- Es simple y eficaz.
- No hace ninguna suposición sobre la distribución de los datos.
- La fase de entrenamiento es rápida.
- Pocos hiperparámetros.

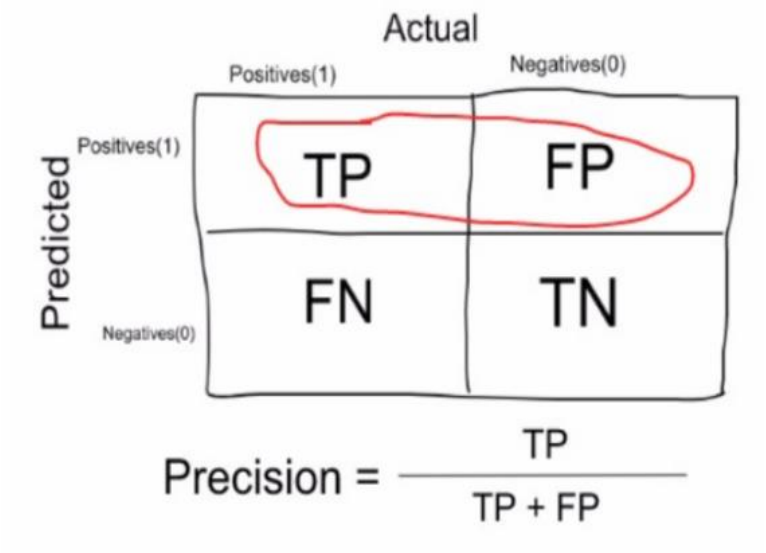
Desventajas:

- No produce un modelo, limitando la capacidad de entender como las variables predictoras (X´s) están relacionadas con la clase a predecir (Y).
- Requiere la selección de un K apropiado.
- La fase de clasificación es lenta.
- Variables cualitativas y datos perdidos requeren de procesado adicional.
- Propenso al sobreajuste.

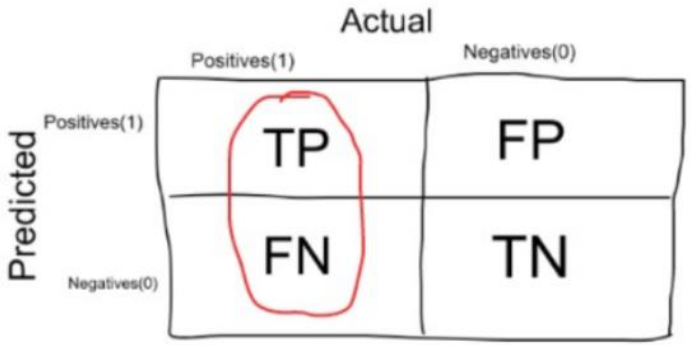
Métricas

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).

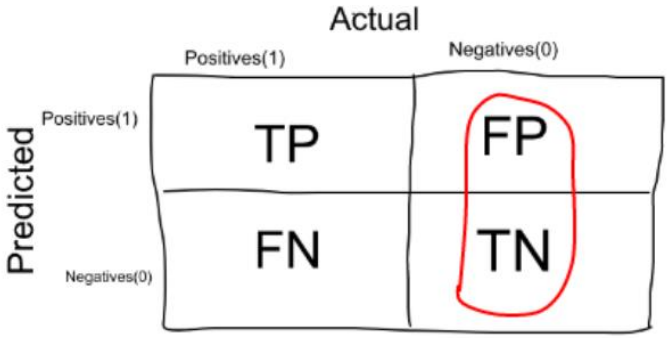




$$\text{Precision} = \frac{TP}{TP + FP}$$



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



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

Métricas

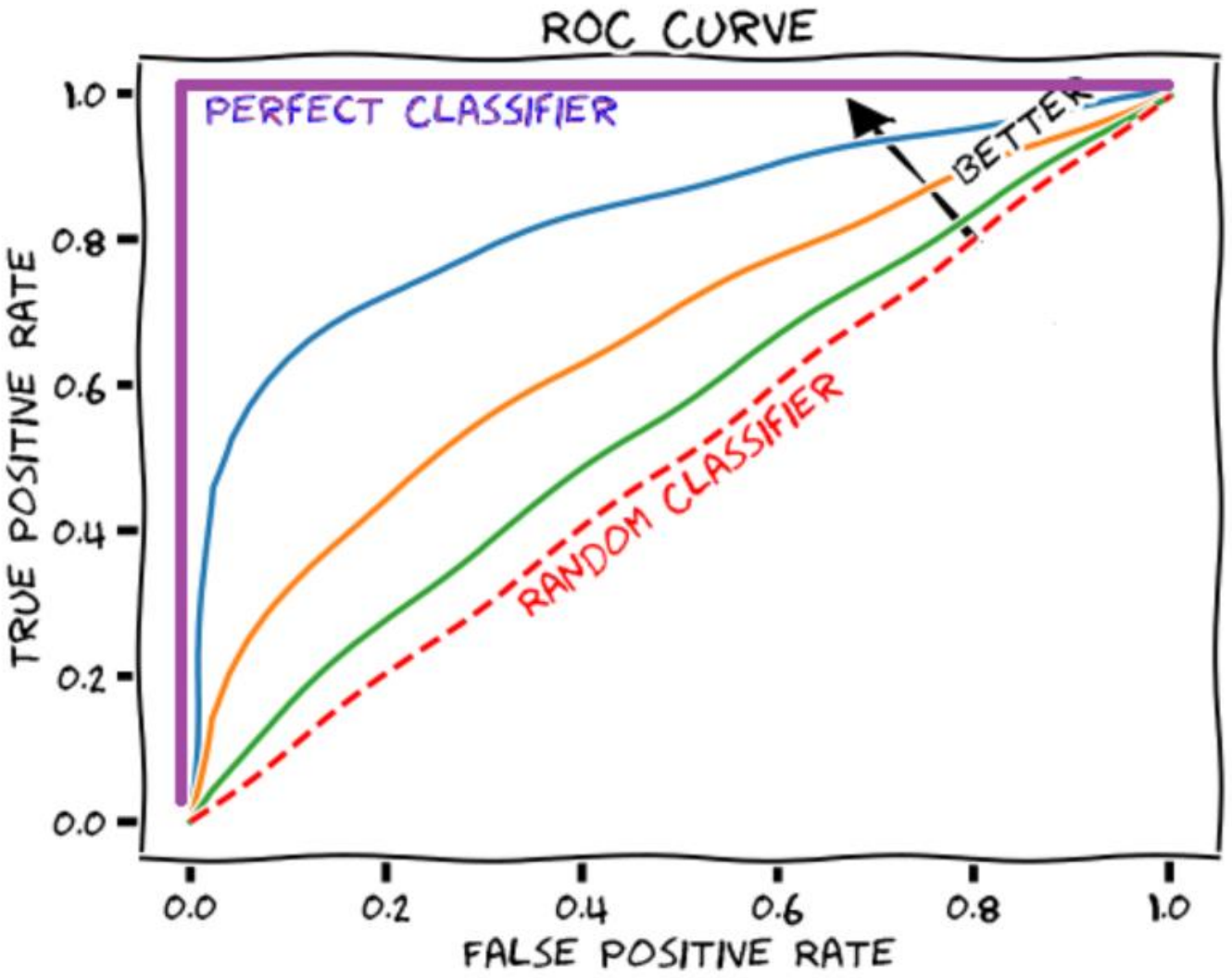
Métricas

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



Aplicaciones del algoritmo K-NN

- ✓ **Motores de recomendación:** utilizando datos de flujo de clics de sitios web, el algoritmo KNN se ha utilizado para proporcionar recomendaciones automáticas a los usuarios sobre contenido adicional.
- ✓ **Finanzas**: Ayuda a los bancos a evaluar el riesgo de un préstamo para una organización o individuo. Se utiliza para determinar la solvencia crediticia de un solicitante de préstamo. También se emplea en comercio de futuros y análisis de lavado de dinero.
- ✓ **Cuidado de la salud**: Dentro de la industria de la salud, se ha utilizado haciendo predicciones sobre el riesgo de ataques cardíacos y cáncer de próstata. El algoritmo funciona calculando las expresiones genéticas más probables.
- ✓ **Reconocimiento de patrones**: ha ayudado a identificar patrones, como en texto y clasificación de dígitos. Esto ha sido particularmente útil para identificar números escritos a mano que puede encontrar en formularios o sobres de correo.

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