INTELIGENCIA ARTIFICIAL CON DEEP LEARNING

ING. JORGE ALBERTO CASTELLANOS

Universidad de La Sabana

FACULTAD DE INGENIERÍA

Aprendizaje de máquina supervisado

- Se define como el uso de datos **etiquetados** para entrenar algoritmos que **clasifican** los datos o **predicen** los resultados.
- El aprendizaje supervisado, tiene varios conjuntos de datos que son necesarios para que el modelo funcione, como los datos de entrenamiento.
- Estos datos permiten ajustar los parámetros hasta que ola función de perdida se haya minimizado lo suficiente.

Tipos de problemas

- **Clasificación**: Reconocer entidades especificas dentro de un conjunto de datos e intentar agruparlo según sus características.

  - ✓Clasificadores Lineales
  - ✓Maquina de soporte vectorial(SVM)
  - ✓Arboles de Decisión
  - ✓K vecinos
  - ✓Ramdom forest

Tipos de problemas

- **Regresión**: se usa para relacionar las variables dependientes e independientes. Esto permite hacer proyecciones, como por ejemplo el valor de un servicio de transporte.

  - Regresión Lineal
  - Regresión logística
  - Regresión polinómica

Algoritmos de aprendizaje supervisado

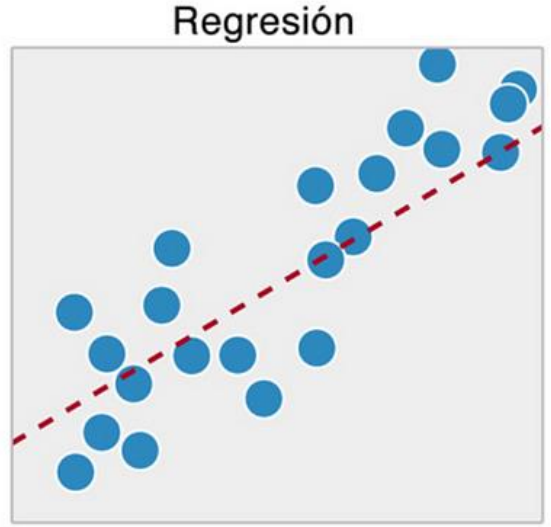
- Regresión Lineal
- Regresión Logistica
- Naive Bayes
- Máquinas de vectores de soporte (SVM)
- K vecinos más cercanos (K-NN)
- Árbol de Decisión (Decision Tree)

Regresión Lineal

- Modela la relación lineal entre una variable dependiente (Y) y una o más variables independientes (X) mediante una línea recta.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \epsilon$$

- Y : Variable dependiente.
- β_0: Intercepto.
- $\beta_1, \beta_2, \ldots, \beta_n$: Coeficientes.
- ϵ : Error.



Regresión Lineal

- **Ejemplo:**

Contexto: Predecir el precio de una casa (y*y*) basado en su tamaño (x*x*).

1. Datos: Tamaño (m²) y precio (USD) de 100 casas.
2. Entrenamiento: Ajustar la línea $y=\beta0+\beta1x$ minimizando el error cuadrático.
3. Predicción:
   Para $x=120\ m^2$, $y=50{,}000+300\times120=86{,}000$ USD

Universidad de La Sabana
FACULTAD DE INGENIERÍA

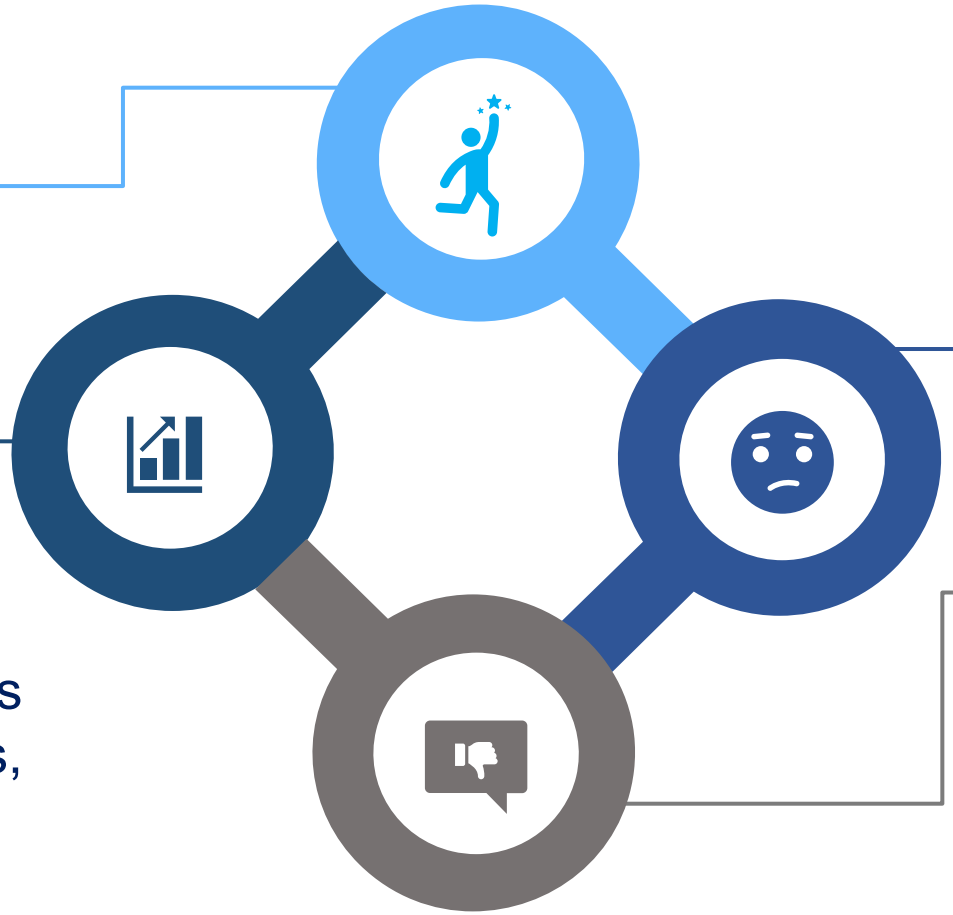
Regresión Lineal

Ventajas

*Simple e interpretable.
*Computacionalmente eficiente.

Usos

*Predecir valores continuos (ej.: precios de casas, ventas futuras).
* Análisis de tendencias.



Desventajas

*Sensible a outliers.
*Asume linealidad y homocedasticidad.

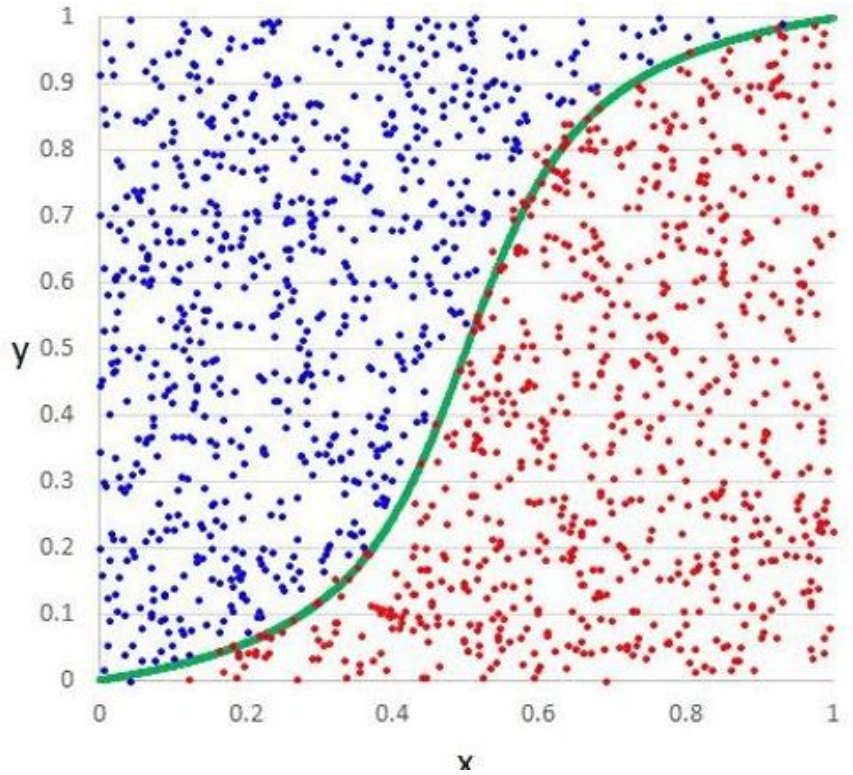
Cuando no usarlo

*Relaciones no lineales.
*Datos con alta varianza o outliers extremos.

Funciona bien con múltiples características, pero requiere regularización (Ridge/Lasso) si hay multicolinealidad.

Regresión Logística

- Utilizado para problemas de clasificación binaria (0 o 1).
- Aplica la función sigmoide para mapear las predicciones a probabilidades.



- $P(Y = 1|X)$: Probabilidad de que Y sea 1 dado X

$$P(Y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n)}}$$

Regresión Logística

Ejemplo

- **Contexto**: Diagnosticar diabetes (y=1) basado en niveles de glucosa (x).

1. Datos: Pacientes con glucosa y diagnóstico.
2. Entrenamiento: Ajustar β para maximizar la verosimilitud.
3. Predicción: Si $P(y=1) \geq 0.5$, clasificar como "diabetes".

Regresión Logística

Universidad de La Sabana
FACULTAD DE INGENIERIA

Regresión Logística

Métodos para el ajuste de hiperparámetros

1. Grid Search (Búsqueda en Cuadrícula)

- Evalúa todas las combinaciones posibles de hiperparámetros definidos en una grilla preestablecida.
- Es exhaustivo, pero computacionalmente costoso en espacios de búsqueda grandes.

2. Random Search (Búsqueda Aleatoria)

- Muestra combinaciones aleatorias de hiperparámetros dentro de rangos definidos.
- Más eficiente que Grid Search en espacios amplios, aunque no garantiza encontrar el óptimo global.

3. Optimización Bayesiana

- Utiliza modelos probabilísticos para guiar la búsqueda hacia regiones prometedoras, reduciendo el número de evaluaciones necesarias.
- Herramientas comunes: Optuna, scikit-optimize, Hyperopt.

Universidad de La Sabana
FACULTAD DE INGENIERÍA

Regresión Logística

Métodos para el ajuste de hiperparámetros

4. Búsqueda por Gradiente (Gradient-based Optimization)

- Aplicable cuando el modelo y la función de pérdida son diferenciables.
- Menos común en regresión logística, pero útil en variantes avanzadas (ej: regresión logística con regularización personalizada).

5. Validación Cruzada Anidada (Nested Cross-Validation)

- Combina selección de hiperparámetros y evaluación del modelo en particiones anidadas de datos.
- Reduce el sobreajuste en la estimación del rendimiento.

Regresión Logística

Hiperparámetros Principales a Ajustar

- **C** (inverso de la regularización).
- **penalty** (l_1, l_2).
- **solver** (elección depende del tipo de regularización y tamaño del dataset).
- **max_iter** (límite de iteraciones para convergencia).

Regresión Logística

Tipos de regularización aplicados para evitar el sobreajuste (overfitting) y mejorar la generalización del modelo

1. Regularización L_1 (Lasso Regression)

Definición:

- También llamada **regularización de Lasso** (*Least Absolute Shrinkage and Selection Operator*).
- Penaliza los coeficientes del modelo añadiendo a la función de pérdida la **suma de los valores absolutos** de los coeficientes (norma L_1):

$$\text{Pérdida} = \text{Pérdida original} + \lambda \sum_{i=1}^{n} |w_i|$$

Donde λ es el hiperparámetro de regularización y wi son los coeficientes.

Regresión Logística

Tipos de regularización

1. Regularización L_1 (Lasso Regression)

Efectos clave

- **Sparsidad**: Tiende a reducir algunos coeficientes **exactamente a cero**, lo que efectivamente **elimina características irrelevantes** (*feature selection*).
- Útil cuando hay muchas variables predictoras, pero solo unas pocas son relevantes.

Limitaciones:

- No funciona bien con solvers que requieren derivadas segundas (como newton-cg o lbfgs).
- Requiere solvers compatibles como liblinear o saga.

Regresión Logística

Tipos de regularización aplicados para evitar el sobreajuste (overfitting) y mejorar la generalización del modelo

1. Regularización L_2 (Ridge Regression)

Definición:

- Conocida como **regularización de Ridge**.
- Penaliza los coeficientes añadiendo a la función de pérdida la **suma de los cuadrados** de los coeficientes (norma L_2):

$$\text{Pérdida} = \text{Pérdida original} + \lambda \sum_{i=1}^{n} w_i^2$$

Regresión Logística

Tipos de regularización

1. Regularización L_2 (Ridge Regression)

Efectos clave

- **Suavizado**: Reduce la magnitud de todos los coeficientes, pero **no los lleva a cero**.
- Funciona bien cuando hay **multicolinealidad** (variables correlacionadas entre sí).
- Más estable numéricamente que L1.

Limitaciones:

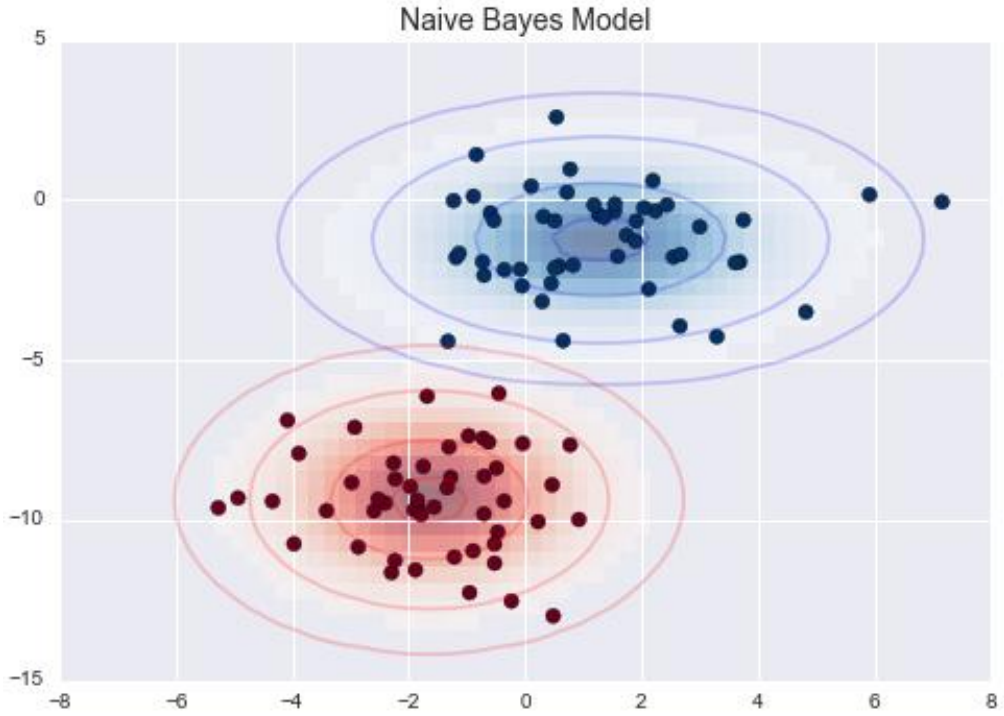
- No realiza selección de características (todas las variables se mantienen en el modelo).
- Compatible con casi todos los solvers (lbfgs, newton-cg, sag, etc.).

Naive Bayes

Basado en el teorema de Bayes, asume independencia entre las características.

$$P(Y|X) = \frac{P(X|Y) \cdot P(Y)}{P(X)}$$

- $P(Y|X)$: Probabilidad posterior
- $P(X|Y)$: Verosimilitud
- $P(Y)$: Probabilidad previa



Naive Bayes

Ejemplo

- **Contexto**: Clasificar emails como spam (y=1) basado en palabras (*xi*).

1. Datos: Frecuencia de palabras en emails etiquetados.
2. Entrenamiento: Calcular P("oferta"|y=1).
3. Predicción: Si P(y=1|X)>P(y=0|X), es spam.

Universidad de La Sabana
FACULTAD DE INGENIERÍA

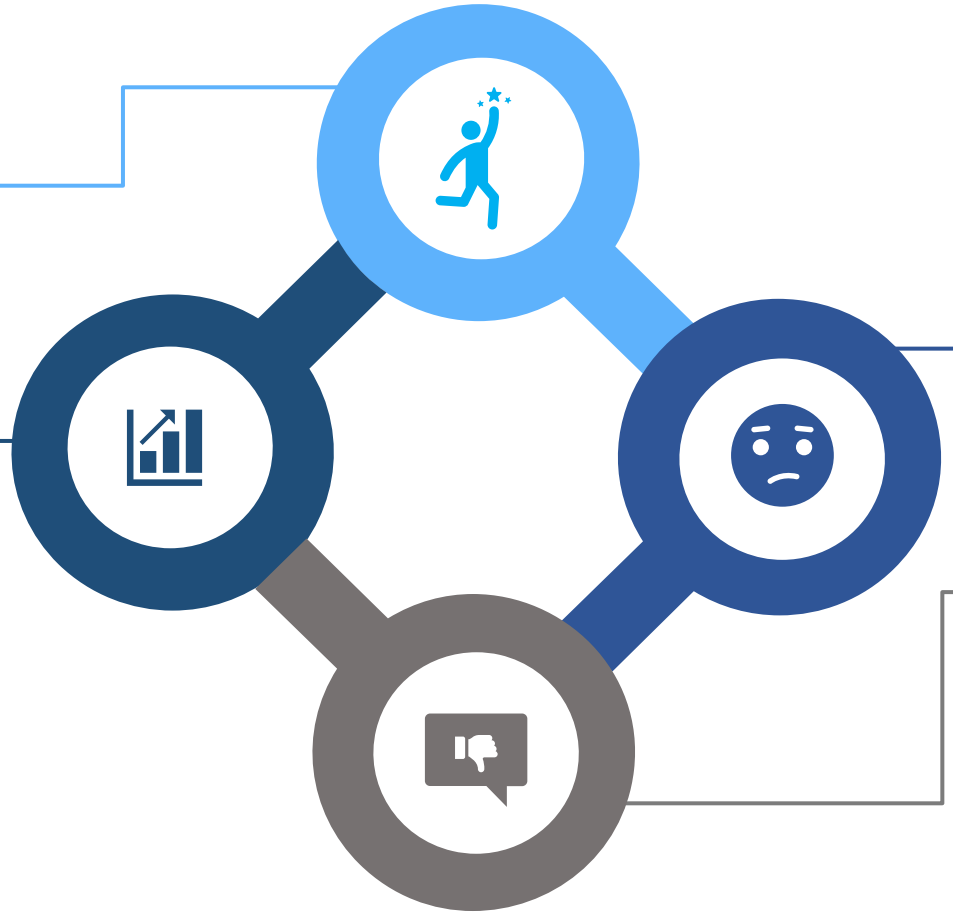
Naive Bayes

Ventajas

*Rápido y escalable.
*Funciona bien con pocos datos.

Usos

*Clasificación de texto (spam, análisis de sentimientos).
*Datos con alta dimensionalidad.



Desventajas

*Supuesto de independencia poco realista.

Cuando no usarlo

*Características correlacionadas.

Múltiples características. Maneja muchas características, pero su rendimiento baja si hay correlaciones.

https://scikit-learn.org/stable/api/sklearn.naive_bayes.html#module-sklearn.naive_bayes

Universidad de La Sabana
FACULTAD DE INGENIERIA

Naive Bayes

1. GaussianNB (Naive Bayes Gaussiano)

Descripción:

- Este método asume que las características (variables predictoras) siguen una distribución gaussiana (normal).
- Es adecuado para datos continuos, donde los valores pueden tomar cualquier valor dentro de un rango.

Cuándo usarlo:

- Cuando tus datos tienen características numéricas continuas que se distribuyen aproximadamente como una campana de Gauss.
- Ejemplos: datos de sensores, mediciones físicas.

Universidad de La Sabana
FACULTAD DE INGENIERIA

Naive Bayes

2. BernoulliNB (Naive Bayes de Bernoulli)

Descripción:

- Este método se utiliza para características binarias (0 o 1).
- Asume que las características siguen una distribución de Bernoulli.

Cuándo usarlo:

- Cuando tus datos representan la presencia o ausencia de algo.
- Ejemplos: clasificación de texto (presencia o ausencia de palabras), datos de encuestas (sí/no).

Naive Bayes

3. CategoricalNB (Naive Bayes Categórico)

Descripción:

- Este método está diseñado para características categóricas, donde los valores son discretos y no necesariamente ordenados.
- Asume que cada característica sigue una distribución categórica.

Cuándo usarlo:

- Cuando tus datos tienen características que representan categorías o grupos.
- Ejemplos: clasificación de colores (rojo, verde, azul), tipos de animales (perro, gato, pájaro).

Naive Bayes

4. ComplementNB (Naive Bayes Complementario)

Descripción:

- Es una adaptación de MultinomialNB que está diseñada para manejar conjuntos de datos desequilibrados.
- En lugar de calcular la probabilidad de una característica dada una clase, calcula la probabilidad de una característica dada el complemento de la clase (todas las demás clases).

Cuándo usarlo:

- Cuando tienes un conjunto de datos donde algunas clases tienen muchos más ejemplos que otras.
- Es especialmente útil en clasificación de texto cuando algunas categorías tienen muchos más documentos que otras.

Naive Bayes

5. MultinomialNB (Naive Bayes Multinomial)

Descripción:

- Este método se utiliza para características discretas que representan conteos, como la frecuencia de palabras en un documento.
- Asume que las características siguen una distribución multinomial.

Cuándo usarlo:

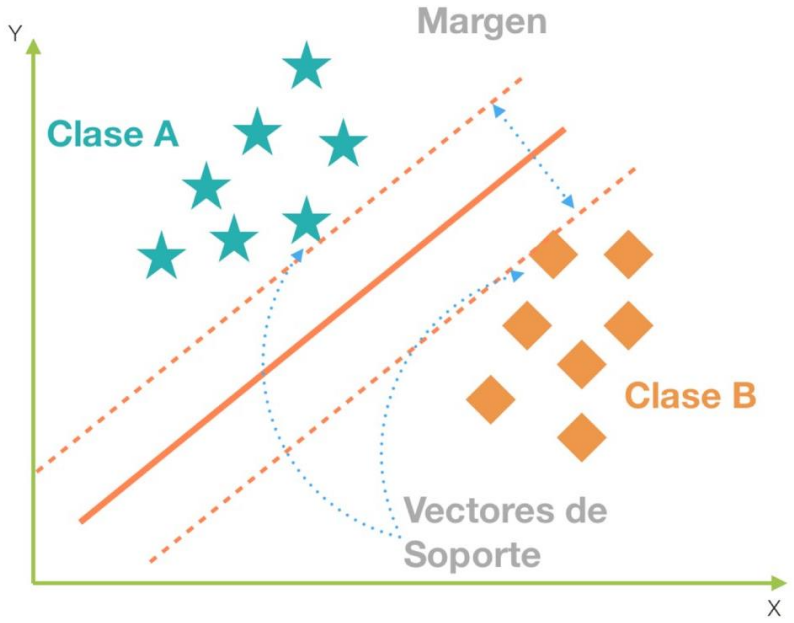
- Principalmente en clasificación de texto, donde las características son la frecuencia de las palabras en los documentos.
- También se puede usar para otros tipos de datos de conteo.

Máquinas de Vectores de Soporte (SVM)

- Busca el hiperplano óptimo que separa las clases con el mayor margen.
- Esta técnica se utiliza principalmente en la clasificación de imágenes.

$$w \cdot X + b = 0$$

- ω: Vector de pesos
- b: Sesgo

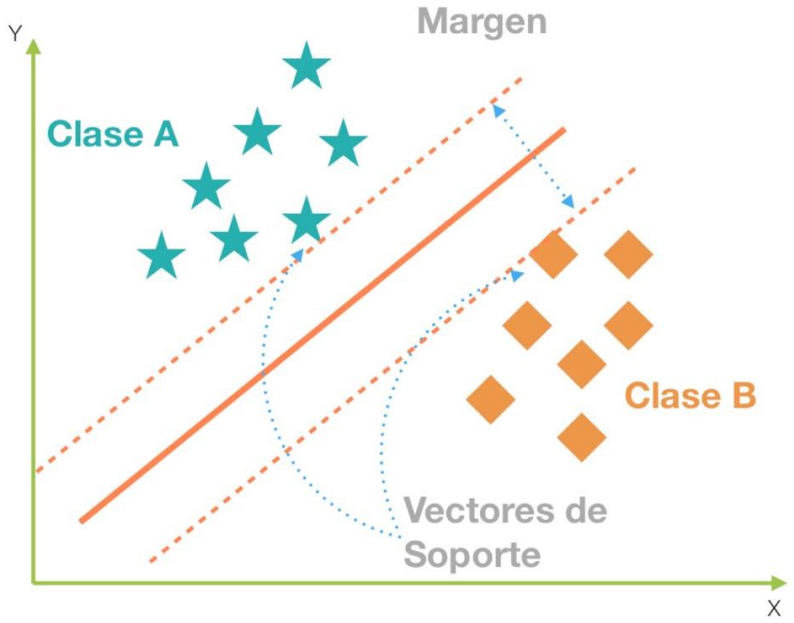


Máquinas de Vectores de Soporte (SVM)

- Busca el hiperplano óptimo que separa las clases con el mayor margen.
- Esta técnica se utiliza principalmente en la clasificación de imágenes.

$$w \cdot X + b = 0$$

- ω: Vector de pesos
- b: Sesgo



Universidad de La Sabana
FACULTAD DE INGENIERÍA

Máquinas de Vectores de Soporte (SVM)

Ventajas

Efectivas en espacios de alta dimensionalidad.

Aún efectivas en casos donde el número de dimensiones es mayor que el número de muestras.

Utilizan un subconjunto de puntos de entrenamiento en la función de decisión (llamados vectores de soporte), por lo que también son eficientes en términos de memoria.

Versátiles: se pueden especificar diferentes funciones de núcleo para la función de decisión. Se proporcionan núcleos comunes, pero también es posible especificar núcleos personalizados.

Universidad de La Sabana
FACULTAD DE INGENIERIA

Máquinas de Vectores de Soporte (SVM)

Desventajas

Si el número de características es mucho mayor que el número de muestras, evitar el sobreajuste al elegir las funciones de núcleo y el término de regularización es crucial.

Las SVM no proporcionan directamente estimaciones de probabilidad; estas se calculan utilizando una costosa validación cruzada de cinco pliegues (ver Puntajes y probabilidades, más abajo).

Usos

•Clasificación binaria, imágenes, texto.

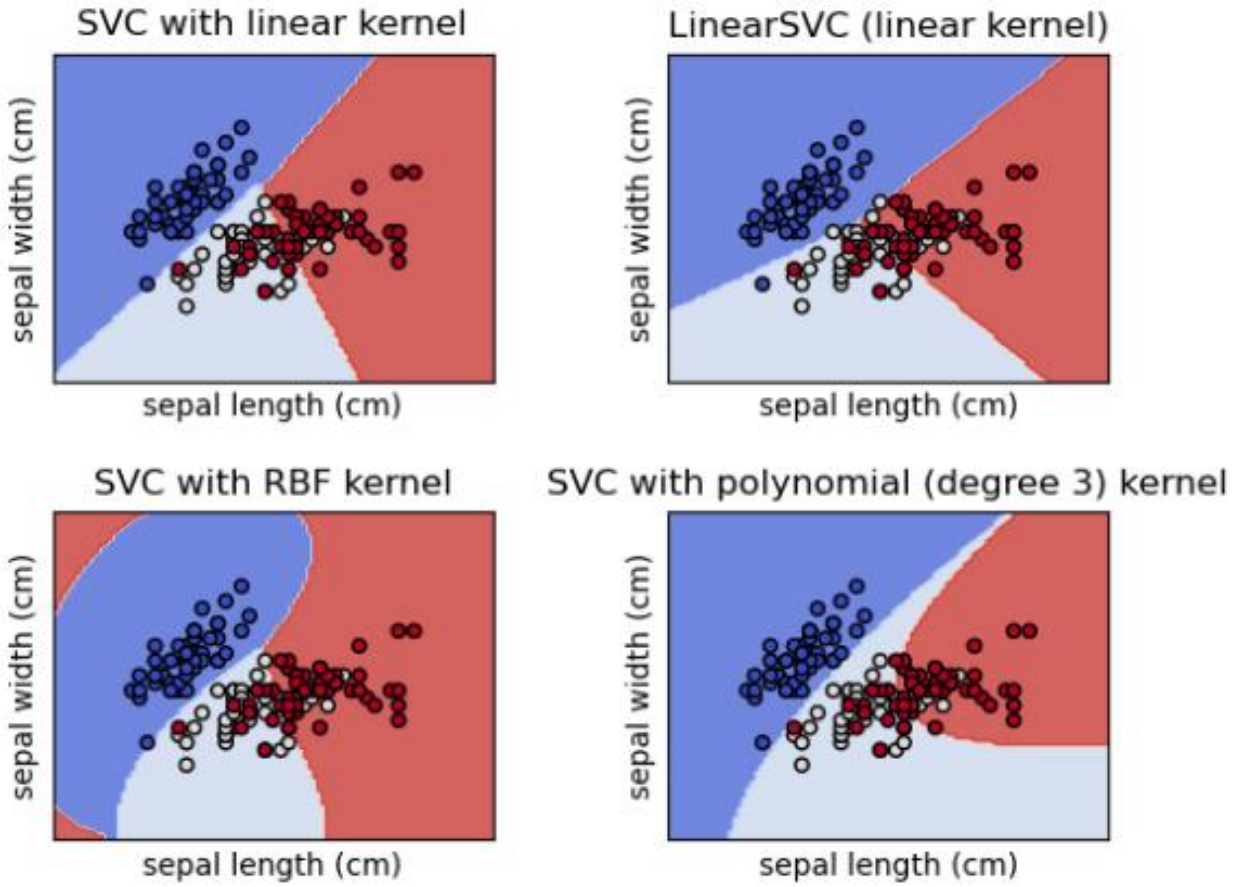
Cuándo no usarlo

•Conjuntos de datos muy grandes (lento).

Universidad de La Sabana
FACULTAD DE INGENIERIA

Máquinas de Vectores de Soporte (SVM)

Clasificación



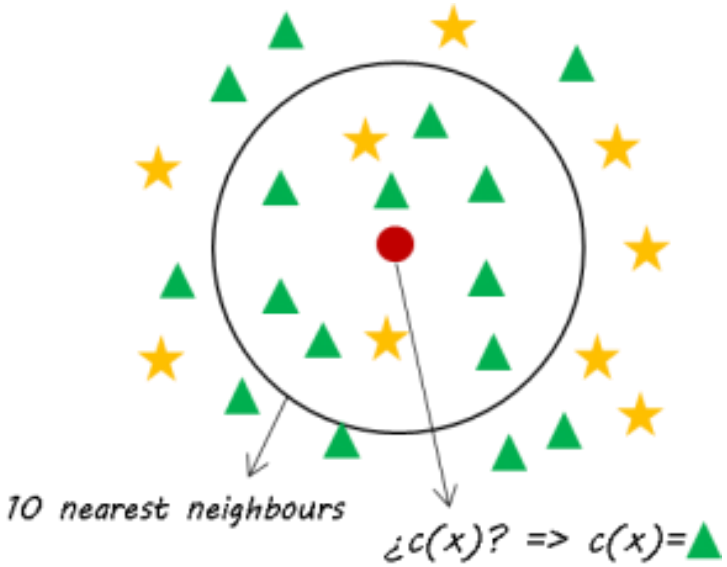
Universidad de La Sabana
FACULTAD DE INGENIERIA

K vecinos más cercanos (K-NN)

- Algoritmo basado en instancias que clasifica un punto basado en la mayoría de sus k vecinos más cercanos.
- Distancia común euclidiana.

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

- Esta técnica se utiliza principalmente en la clasificación de imágenes y seguimiento de patrones.



K vecinos más cercanos (K-NN)

Ejemplo

- **Contexto**: Clasificar una flor como setosa o versicolor basado en medidas de pétalos.

1. Datos: Dataset Iris con 150 flores etiquetadas.
2. Entrenamiento: Almacenar todos los datos (k-NN es *lazy*).
3. Predicción: Para una nueva flor, encontrar los 5 vecinos más cercanos (k=5). Si 4 son setosa, clasificar como setosa.

Universidad de La Sabana
FACULTAD DE INGENIERÍA

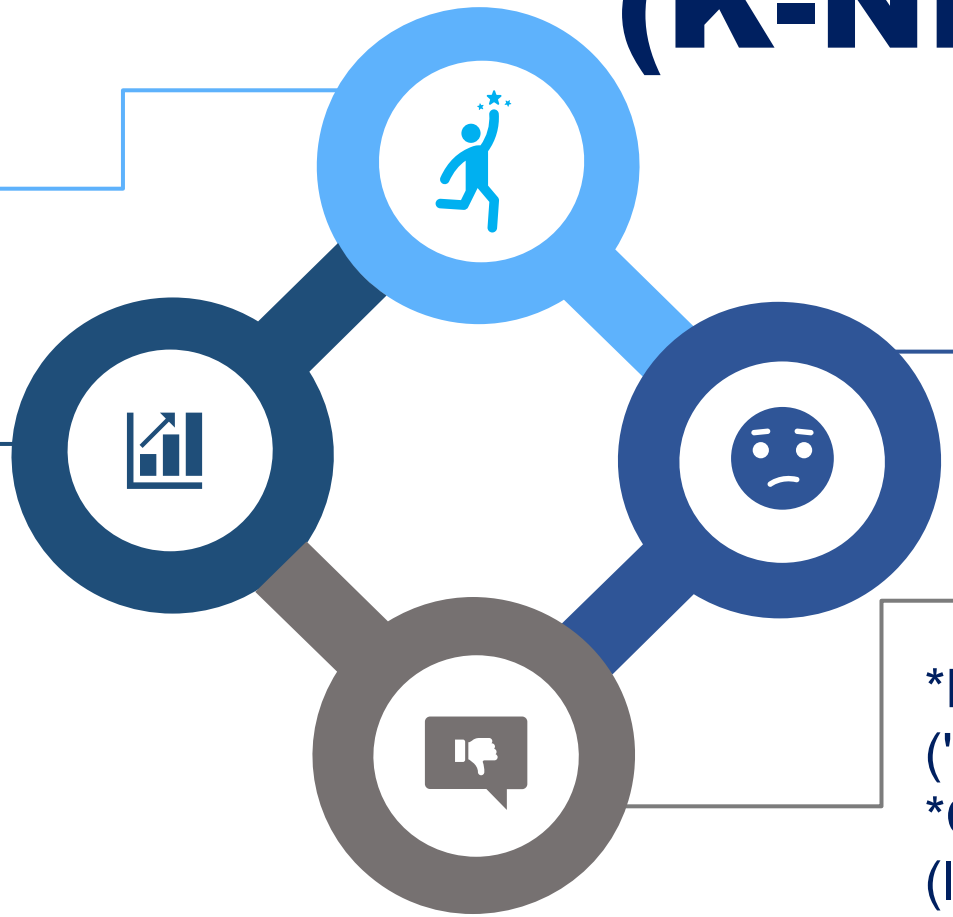
K vecinos más cercanos (K-NN)

Ventajas

*Simple e intuitivo.
*No asume distribución de datos.

Usos

*Sistemas de recomendación (ej.: "usuarios similares").
*Clasificación de imágenes o texto.



Desventajas

*Costoso computacionalmente en predicción (*O(n)* por consulta)
*Sensible a características no relevantes.

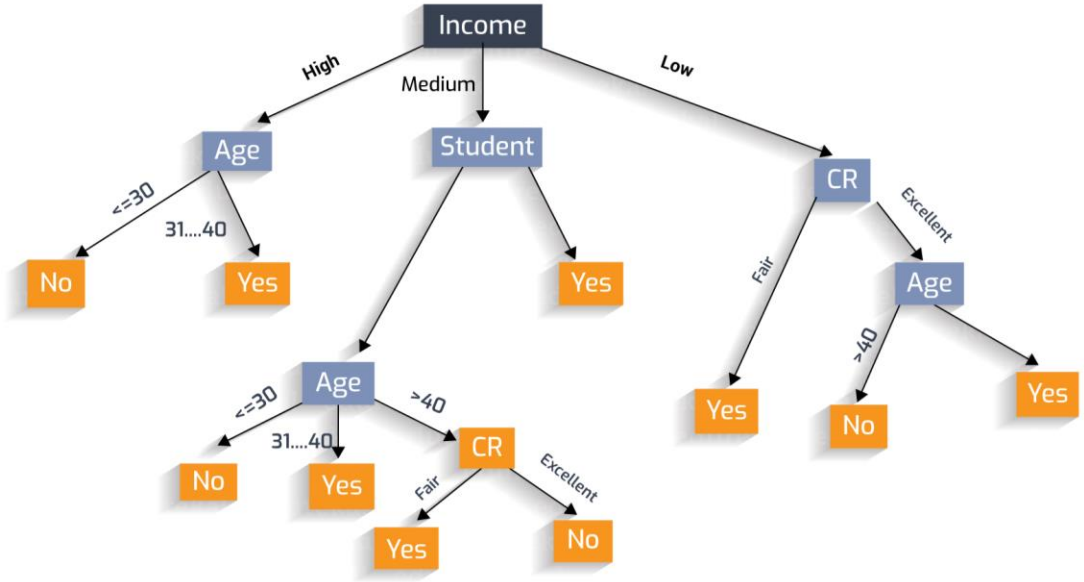
Cuando no usarlo

*Datos de alta dimensionalidad ("maldición de la dimensionalidad").
*Conjuntos de datos muy grandes (lento en predicción).

Múltiples características. Requiere normalización para que ninguna característica domine la distancia. Rendimiento disminuye con alta dimensionalidad.

Árbol de Decisión (Decision Tree)

Un Árbol de Decisión es un modelo de aprendizaje supervisado que divide recursivamente los datos en subconjuntos basados en reglas de decisión, hasta alcanzar nodos hoja que representan una predicción (clasificación o regresión).



Universidad de La Sabana
FACULTAD DE INGENIERÍA

Árbol de Decisión (Decision Tree)

Ecuación (Criterio de División)

El algoritmo selecciona la mejor división maximizando la pureza de los nodos hijos.

Para Clasificación (Gini/Entropía)

$$\text{Gini}(D) = 1 - \sum_{i=1}^{k} p_i^2$$

$$\text{Entropía}(D) = -\sum_{i=1}^{k} p_i \log_2(p_i)$$

Donde:

D: Subconjunto de datos en un nodo.

pi: Proporción de muestras de clase *i* en *D*.

Para Regresión (Reducción de Varianza)

$$\text{Varianza}(D) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2$$

Donde:

Yi: Valor objetivo de la i-ésima muestra.

ȳ: Media de los valores en D.

Universidad de La Sabana
FACULTAD DE INGENIERÍA

Árbol de Decisión (Decision Tree)

Explicación de Términos

- **Nodo raíz**: Contiene todos los datos iniciales.
- **Nodos internos**: Aplican una condición (ej: Edad≤30).
- **Nodos hoja**: Predicción final (clase o valor continuo).
- **Criterio de división**:
  - **Gini**: Mide la impureza (0 = nodo puro).
  - **Entropía**: Mide el desorden (0 = nodo ordenado).

Universidad de La Sabana
FACULTAD DE INGENIERIA

Árbol de Decisión (Decision Tree)

Ejemplo

Problema: Clasificar si un préstamo será pagado (Sí/No) basado en Ingresos, Edad y Historial Crediticio.

1. **División inicial**:

- Mejor división: Historial Crediticio=Bueno.
  1. **Rama Izquierda**: 90% paga (Sí), 10% No.
  2. **Rama Derecha**: 20% paga (Sí), 80% No.

1. **Predicción**:

- Nuevo cliente: Historial=Bueno, Edad=40 → **Sí**.

Universidad de La Sabana
FACULTAD DE INGENIERIA

Árbol de Decisión (Decision Tree)

Usos comunes

- ✅ Clasificación binaria/multiclase (ej: diagnóstico médico).
- ✅ Regresión (ej: predecir precios de casas).
- ✅ Sistemas de reglas interpretables (ej: aprobación de créditos).

Cuándo no usarlo

- ❌ Datos con relaciones complejas no lineales (mejor: Random Forest/Redes Neuronales).
- ❌ Datos con mucho ruido o outliers (sobreajusta fácilmente).
- ❌ Si se necesita alto rendimiento en datos multidimensionales (mejor: métodos de ensamble).

Universidad de La Sabana
FACULTAD DE INGENIERÍA

Árbol de Decisión (Decision Tree)

Ventajas

- ✅ Fácil interpretación visual.
- ✅ No requiere normalización de datos.
- ✅ Rápido entrenamiento y predicción.

Desventajas

- ❌ Propenso a overfitting (sin poda).
- ❌ Inestable (pequeños cambios afectan estructura).
- ❌ Sesgo alto en datos complejos.

Universidad de La Sabana
FACULTAD DE INGENIERIA

Árbol de Decisión (Decision Tree)

Comportamiento con Múltiples Características

Ventaja: Selecciona automáticamente las features más importantes.

Desventaja:

- Puede ignorar relaciones entre features si no son óptimas para divisiones.
- En alta dimensionalidad, tiende a sobreajustar (requiere ajuste de profundidad máxima).

Comparación de Algoritmos

Algoritmo	Tipo de Problema	Ventajas	Desventajas
Regresión Lineal	Regresión	Simple y rápido	Sensible a outliers
Regresión Logística	Clasificación	Probabilístico	Solo para problemas binarios
Naive Bayes	Clasificación	Eficiente con datos grandes	Asume independencia entre features
SVM	Clasificación/Regresión	Efectivo en espacios de alta dimensión	Lento con grandes datasets
K-NN	Clasificación/Regresión	Simple de implementar	Costoso computacionalmente
Random Forest	Clasificación/Regresión	Reduce sobreajuste	Menos interpretable
Redes Neuronales	Clasificación/Regresión	Alta precisión en problemas complejos	Requiere muchos datos y recursos