



Módulo V | Aprendizaje supervisado

Clase 13

Algoritmos más comunes parte 2





¿Ponemos a grabar el
taller?

¿Qué vamos a ver hoy?



- Repasamos Árboles de Decisión
- Random Forest

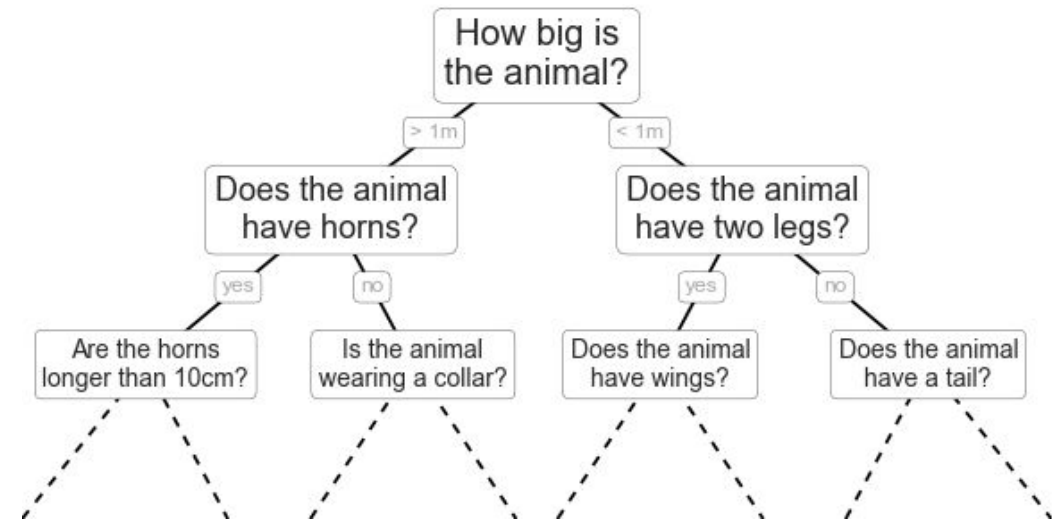
- K-Nearest Neighbors
- Support Vector Machine



Repasamos

Árboles de decisión

Un árbol de decisión **hace preguntas** y va clasificando de acuerdo a las respuestas.



Árboles de decisión



- Clasificación y Regresión
- Simple de entender, interpretar y visualizar
- Modelo base para modelos más complejos (Random Forest, etc)
- **Parámetros para nodos:**
 - Impureza de Gini
 - Ganancia de información
- Solo toman atributos numéricos
- Tendencia a ajustarse a los datos de entrenamiento

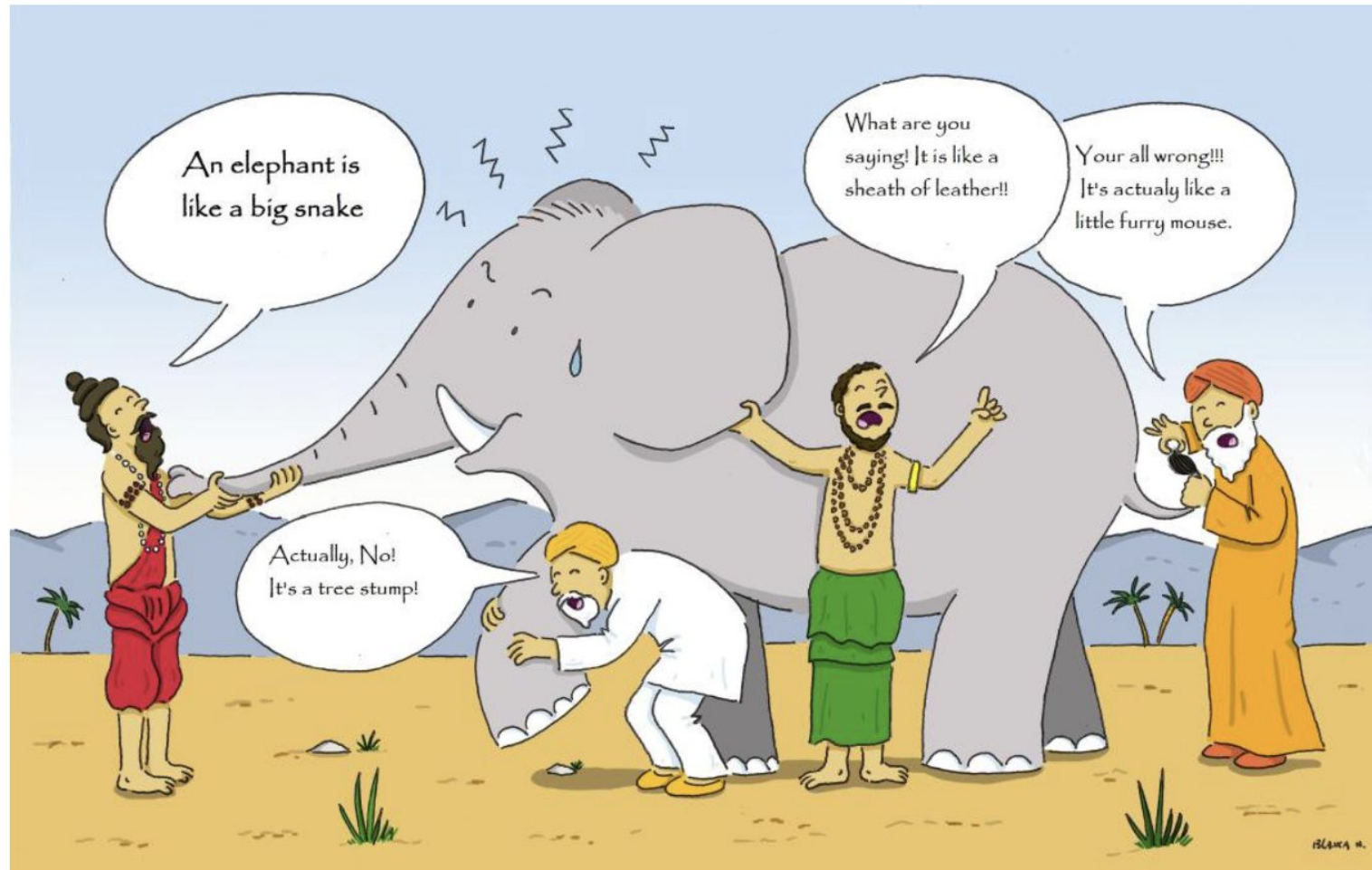


¿Alguna consulta?



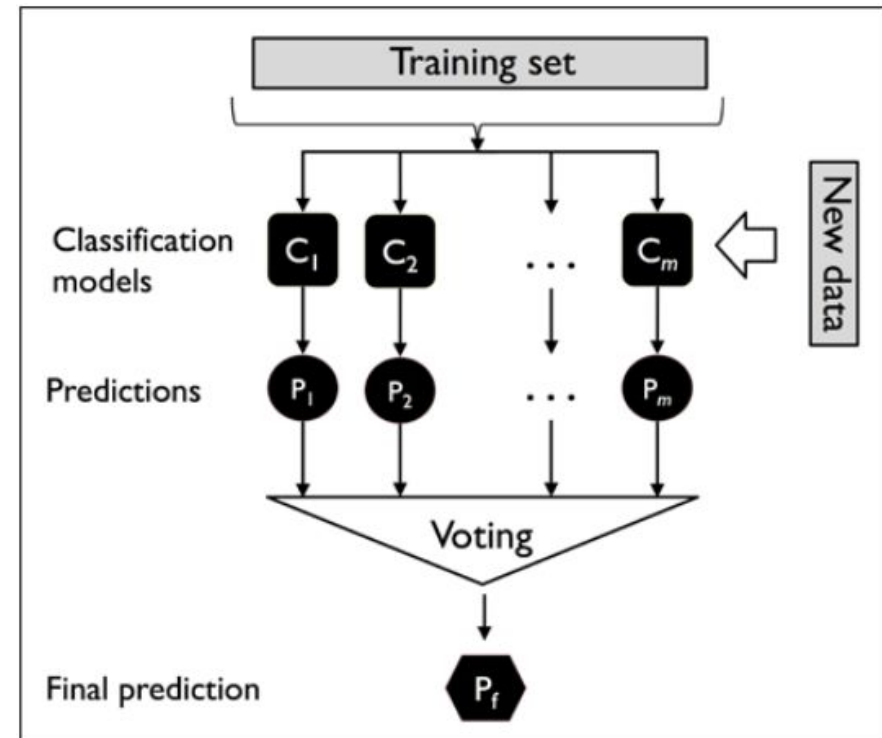
Random Forest

Random Forest



Random Forest

- Entrenar muchos modelos y cada uno tendrá un resultado.
- La clasificación o valor resultante es el resultado más frecuente.



Random Forest

- Si todos los modelos son muy parecidos, no agregan mucha información nueva en la votación final.
- Se necesitan **modelos diferentes entre sí, poco correlacionados**.
- Regresión y Clasificación
- **Ventajas**
 - Robusto frente a outliers y ruido
 - Buenos estimadores de error e importancia de variables



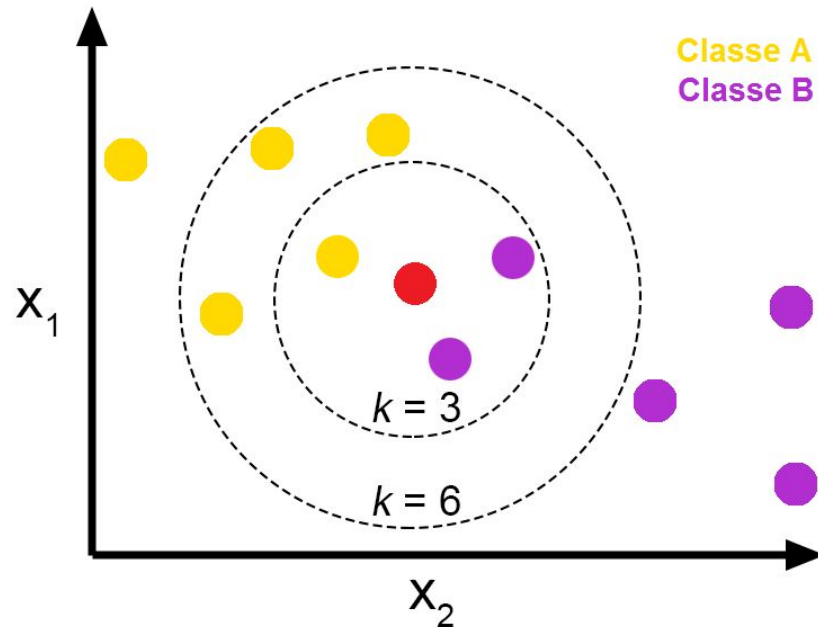
K-Nearest Neighbors (kNN)



K Nearest Neighbors (kNN)

El método de **K Nearest Neighbors (kNN)** consiste en etiquetar muestras en función de la etiqueta que tengan los **K** vecinos más cercanos

K Nearest Neighbors (kNN)



- Calcula la **distancia** a cada uno de los puntos existentes.
- Selecciona la etiqueta que más frecuente aparece en las K clases

Support kNN Machines

- Clasificación y Regresión
- **Ventajas**
 - Rápido porque no tiene nada que ajustar, solo almacena los datos
 - Es fácil de entender sobre todo en clasificación
- **Desventajas**
 - La elección del número de vecinos no es trivial
 - No tiene buen desempeño en datasets con desbalance de clases
 - Requiere homogeneizar features



Descanso

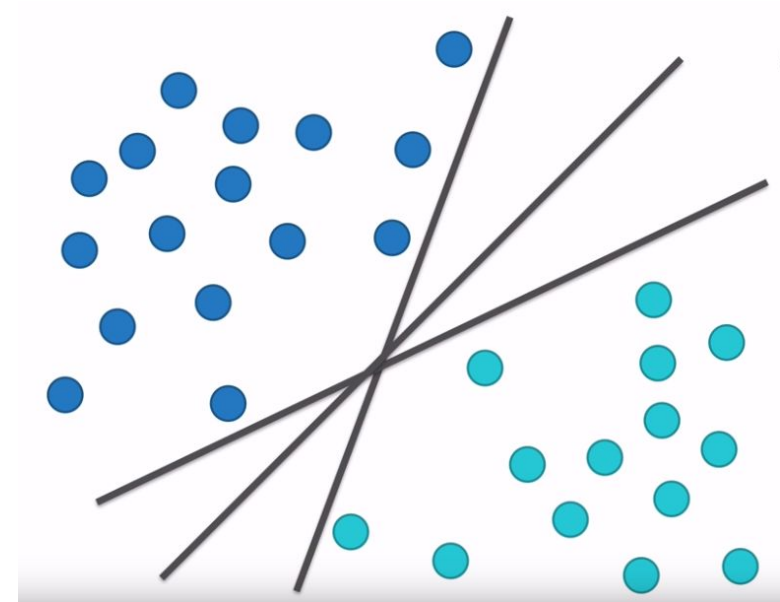
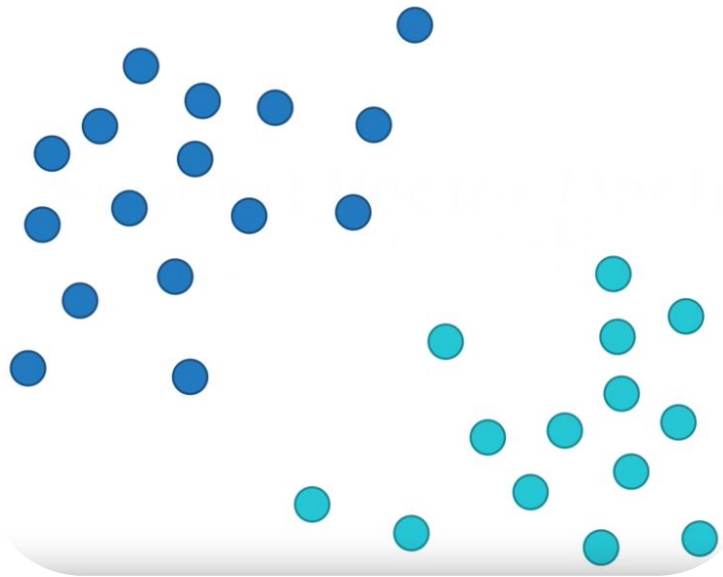
Nos vemos en 10 minutos



Support Vector Machines

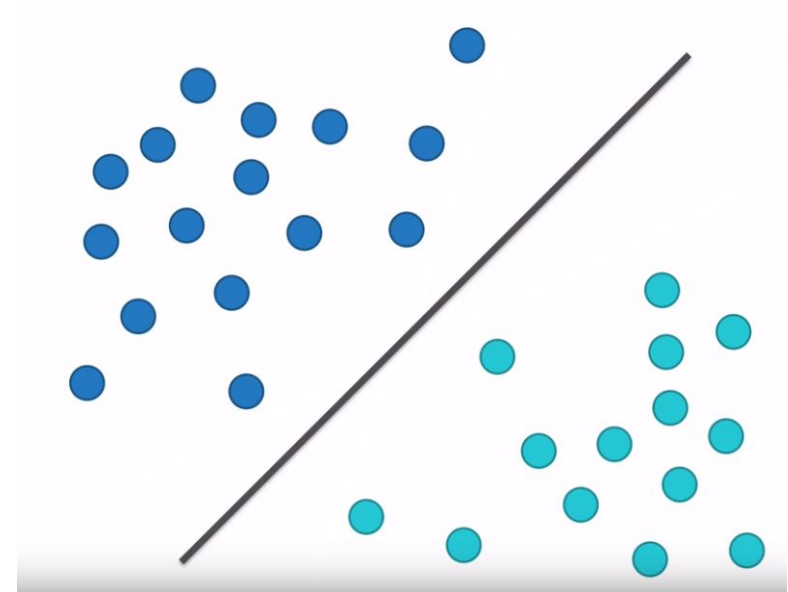
Support Vector Machines

Tarea: Separar los puntos azules de los celestes

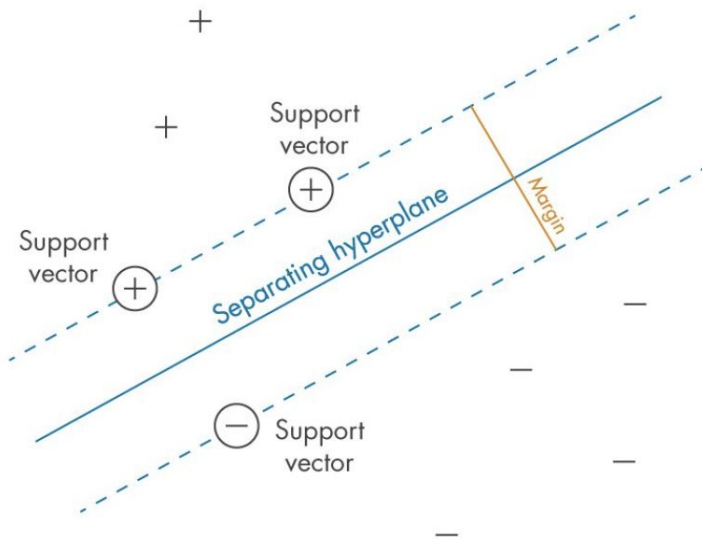


Support Vector Machines

- SVM elige la recta que separa los puntos de la “**mejor manera posible**”

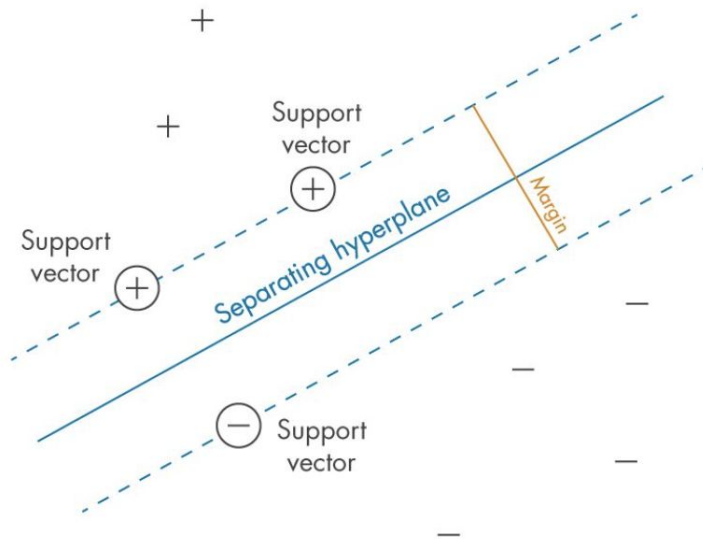


Support Vector Machines



- Establece un hiperplano que separa los puntos **maximizando** el margen.
 - La distancia del margen al vector de soporte es la mayor que se puede encontrar.
 - Es el mayor margen que se puede encontrar.

Support Vector Machines



Margen: Es la distancia de las instancias más cercanas a la recta de decisión.

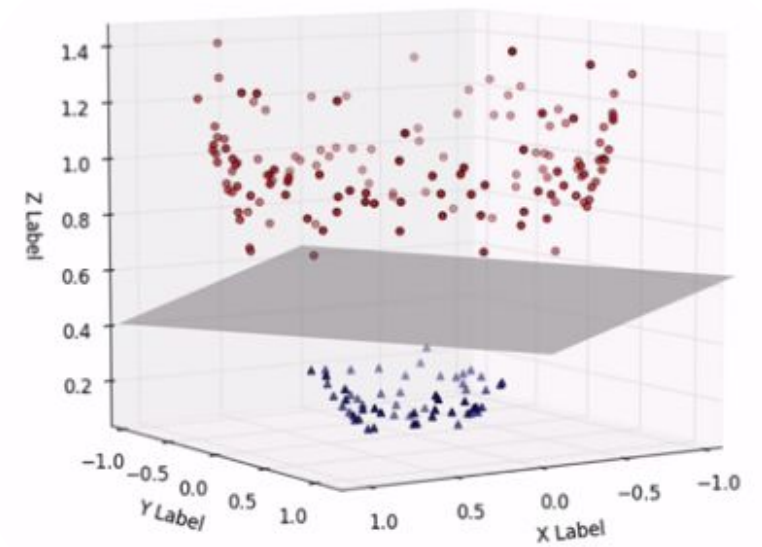
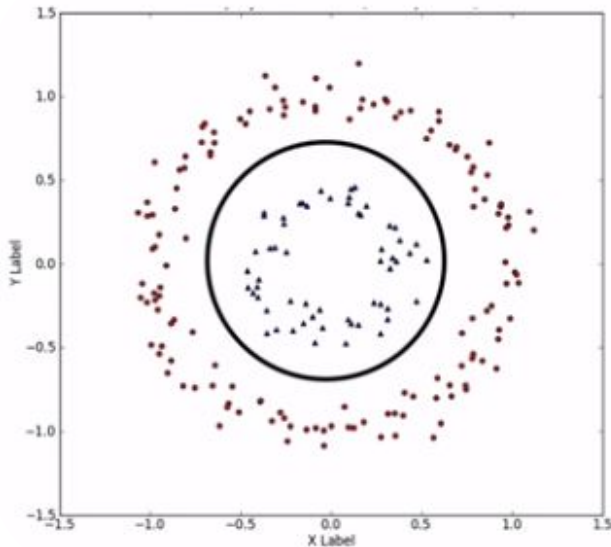
Vectores de soporte: Las instancias más cercanas al margen. Es una de las pocas características de los datos que le importa al algoritmo.

Lo que hace el algoritmo es encontrar la recta que maximice el margen.

Es un problema de optimización, lo que conlleva a una solución computacionalmente eficiente.

Support Vector Machines

Kernels: Se utilizan cuando los datos directamente no se pueden separar con una línea



Support Vector Machines

- Clasificación y Regresión
- **Ventajas:**
 - Eficaz en espacios de alta dimensión
 - Eficiente en memoria
 - El uso de kernels lo convierten en muy versátil
- **Desventajas:**
 - El uso de kernels tiende a llevar al algoritmo a overfitting
 - Su mejor desempeño se da para problemas de clasificación



Sección práctica:

Trabajamos con la Notebook 15
ajustando los primeros modelos

En la sala general

Aprendizaje Supervisado - Parte 2

Trabajamos con la Notebook
15

Demostraremos cómo ajustar
los modelos de regresión y
clasificación vistos



Desafío 11

- Para la siguiente clase, repasar y ejercitar la notebook 15.





¿Alguna consulta?

FUNDACIÓN
YPF

¡Muchas gracias!

