

Support Vector Machine - K-Nearest Neighbors

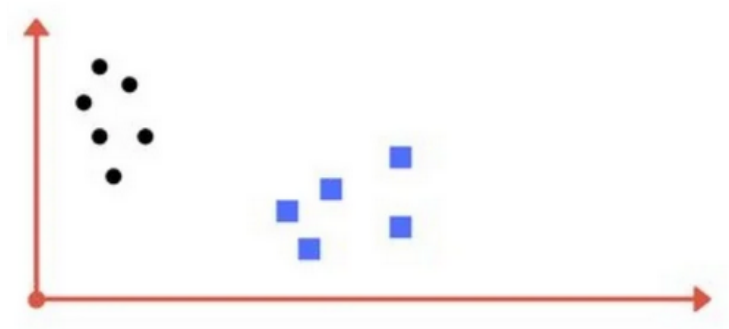
Tomás Fontecilla

2 de septiembre de 2022

Support Vector Machine

Support Vector Machine (SVM) o Vector de soporte de máquina es un clasificador discriminativo que toma datos de entrenamiento (para hacer aprendizaje supervisado) para generar un hiperplano óptimo que categoriza nuevos ejemplos

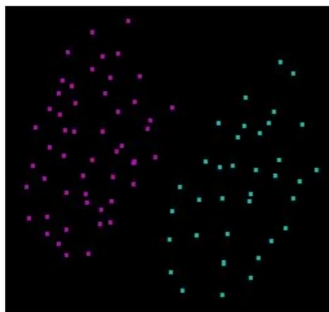
Support Vector Machine - ejemplo



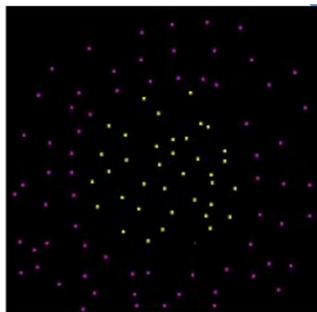
¿Qué se podría graficar para clasificar los puntos negros de los azules?

Support Vector Machine - ejemplo

Datos separados lineal y no linealmente



Linearly separable data

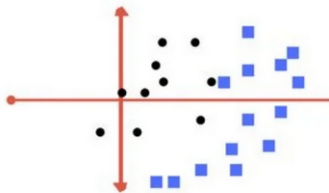


Non linearly separable data

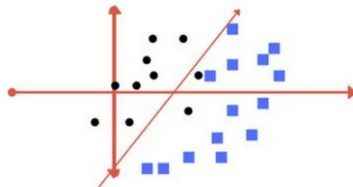
¿Y en este caso?

Support Vector Machine - Caso

Datos separados lineal y no linealmente



Raw Data



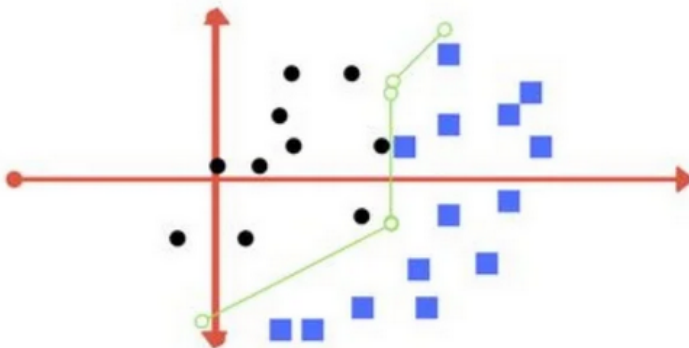
Line as Hyperplane

En este caso, si usamos el hiperplano lineal tenemos:

- Dos puntos negos que caen en la categoría de cuadrados azules
- La separación no es perfecta
- Tolera algunos *outliers* de clasificación

Support Vector Machine - Caso

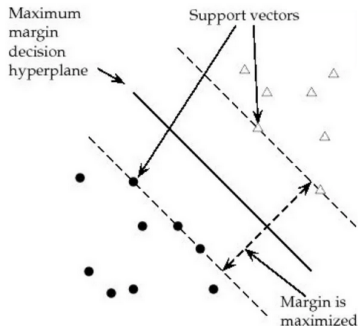
Parece ser óptimo algo así



Pero esto nos genera dos inconvenientes:

- Es difícil entrenar un modelo con estas características.
- Éste es un parámetro de regularización.

Support Vector Machine - Definiciones



SVM lo que quiere finalmente hacer es maximizar el margen de separación del hiperplano, esto es:

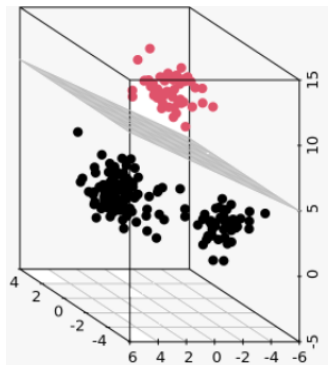
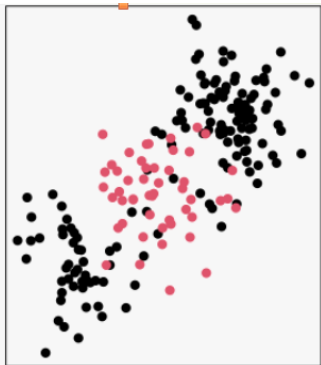
$$\min\left(\frac{\|W\|^2}{2}\right) + c \sum \epsilon \text{ sujeto a } \begin{aligned} x_i \cdot w + b &\geq 1 & \forall y_i = 1 \\ x_i \cdot w + b &\leq -1 & \forall y_i = -1 \\ \text{donde } \epsilon_i &\geq 0 & \forall i \end{aligned} \quad (1)$$

Desafortunadamente los universos a estudiar no se suelen presentar en casos idílicos de dos dimensiones. En general, un algoritmo SVM debe tratar con:

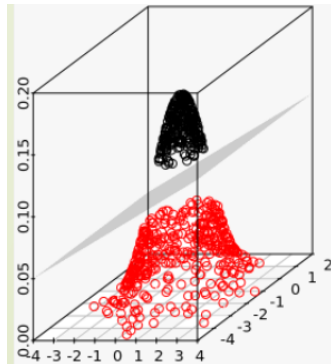
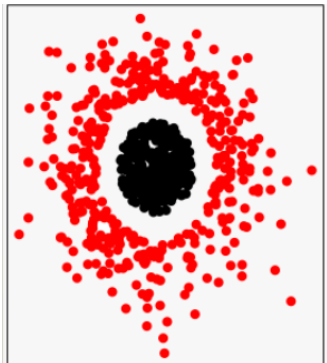
- Más de 2 variables predictoras
- Curvas no lineales de separación
- Casos donde los conjuntos de datos no pueden ser completamente separados
- Clasificaciones en más de 2 categorías

¿Y si mapeamos el conjunto de datos a un nuevo espacio de mayor dimensionalidad?

Support Vector Machine - Incrementar las dimensiones



Support Vector Machine - Incrementar las dimensiones



Support Vector Machine - Kernel

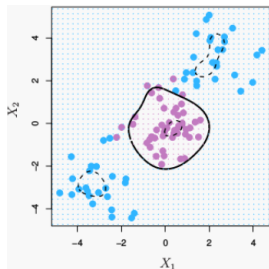
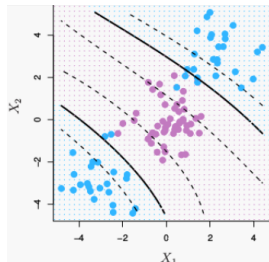
SVM se basa en la función dual lagrangeana, expresada como:

$$L_D = \sum_{i=1}^N \alpha_i y_i \langle h(x), h(x_i) \rangle + \beta_0$$

donde $\langle h(x), h(x_i) \rangle$ es la función kernel que calcula el producto punto en el espacio de la transformación $h(x)$.

Algunos kernel populares son:

- Polinomio grado d:
 $K(x, x') = (1 + \langle x, x' \rangle)^d$
- Base radial:
 $K(x, x') = \exp(-\|x - x'\|^2 / c)$
- Red Neuronal:
 $K(x, x') = \tanh(\kappa_1 \langle x, x' \rangle + \kappa_2)$



Suport Vector Machine - Ventajas y Desventajas

Ventajas

- Algoritmo clasificador en base a una sólida teoría. Teoremas de minimización de riesgo son el estado del arte en aprendizaje estadístico.
- Se puede aplicar a datos representados en cualquier espacio de Hilbert (donde pueda definir una medida de distancia).
- Relativamente pocos parámetros a estimar.
- Se pueden formular nuevas extensiones (flexibilidad).

Desventajas

- Determinar los Kernels a utilizar es complejo.
- Sólo es un clasificador binario. (¿Qué ocurre cuando tenemos un problema de más de dos clases?)

K Nearest Neighbors

“Dime con quién andas y te diré quien eres”

Un refrán que nos puede explicar cómo es el procedimiento de k vecinos más cercanos o KNN.

“Mayoría voto gana”

Es la frase célebre que nos permite entender cómo clasifica el modelo.

K Nearest Neighbors

Este es un algoritmo, que no requiere ajustar un modelo.

Dado un punto x_0 , la idea es encontrar los k puntos de entrenamiento x_r , $r = 1, \dots, k$ más próximos a x_0 y luego clasificar usando mayoría de votos entre los k vecinos.

Los empates se resuelven al azar.

En el caso que todas las características fueran reales, se puede usar la distancia euclídeana en el espacio de características:

$$d_i = ||x_i - x_0||$$

KNN - ¿Qué Es?

- Como hemos visto, es un metodo que simplemente busca en las observaciones más cercanas a la que está tratando de predecir, y clasifica el punto de interés basado en la mayoría de datos que lo rodean.
- Dado que requiere las etiquetas de los otros puntos, entonces este es un algoritmo supervisado.
- Dado que depende exclusivamente de las instancias de la observación, este algoritmo no aprende de un modelo explícito, si no que del “it is known” o “es sabido que”. Es decir, el algoritmo *memoriza* las instancias de entrenamiento y las usa como base de conocimiento para predecir.