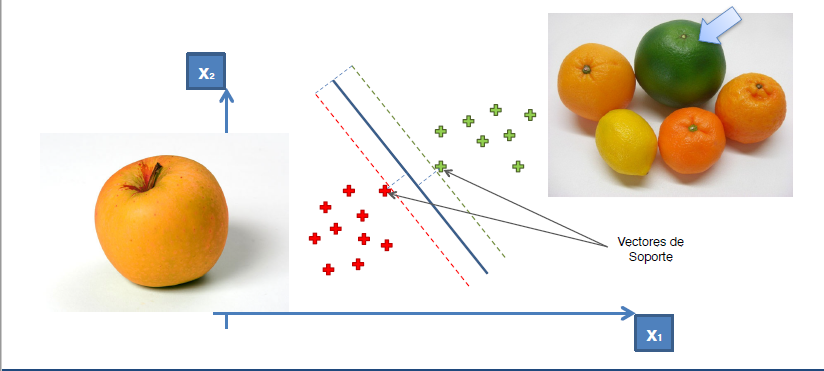
Universidad Galileo

Maestría en Ciencia de Datos

SVM



Jaime Ariel Chitay

19002065

Luis Leal

# 1. ¿Qué es SVM?

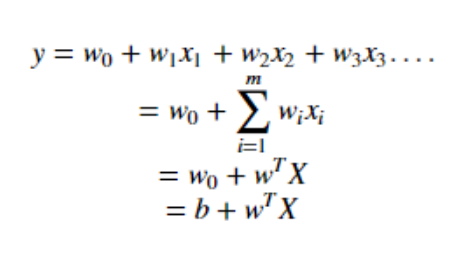
Las máquinas de vectores de soporte, llamadas SVM, son un algoritmo de aprendizaje supervisado que se puede usar para problemas de clasificación y regresión como clasificación de vectores de soporte (SVC) y regresión de vectores de soporte (SVR). Se utiliza para conjuntos de datos más pequeños, ya que lleva demasiado tiempo procesarlos. En este conjunto, nos centraremos en SVC.

SVM se basa en la idea de encontrar un hiperplano que separe mejor las características en diferentes dominios.

# 2. Hiperplano (Superficie de decisión)

El hiperplano es una función que se utiliza para diferenciar entre características. En 2-D, la función utilizada para clasificar entre entidades es una línea, mientras que la función utilizada para clasificar las entidades en un 3D se llama como un plano similar a la función que clasifica el punto en una dimensión superior como un hiperplano.

La ecuación del hiperplano en la dimensión 'M' se puede dar como



dónde,

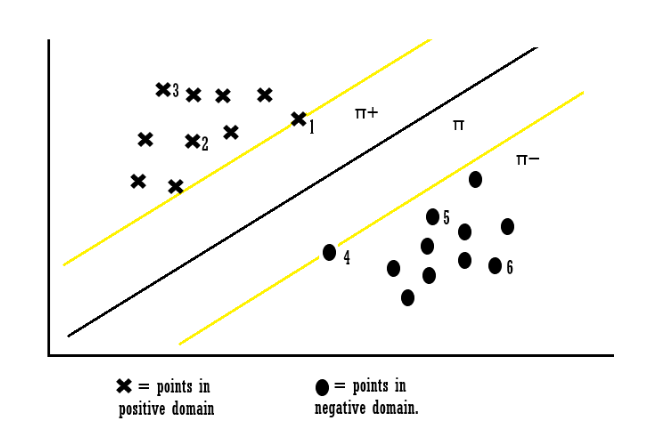
Wi = vectores (W0, W1, W2, W3 ...... Wm)

b = término sesgado (W0)

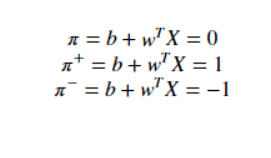
X = variables.

# 3. Hard margin SVM:

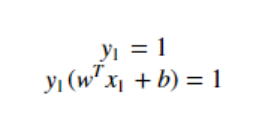
Suponga 3 hiperplanos a saber (π, π +, π−) de modo que 'π +' es paralelo a 'π' que pasa a través de los vectores de soporte en el lado positivo y 'π−' es paralelo a 'π' que pasa a través de los vectores de soporte en el lado negativo



Las ecuaciones de cada hiperplano se pueden considerar como:

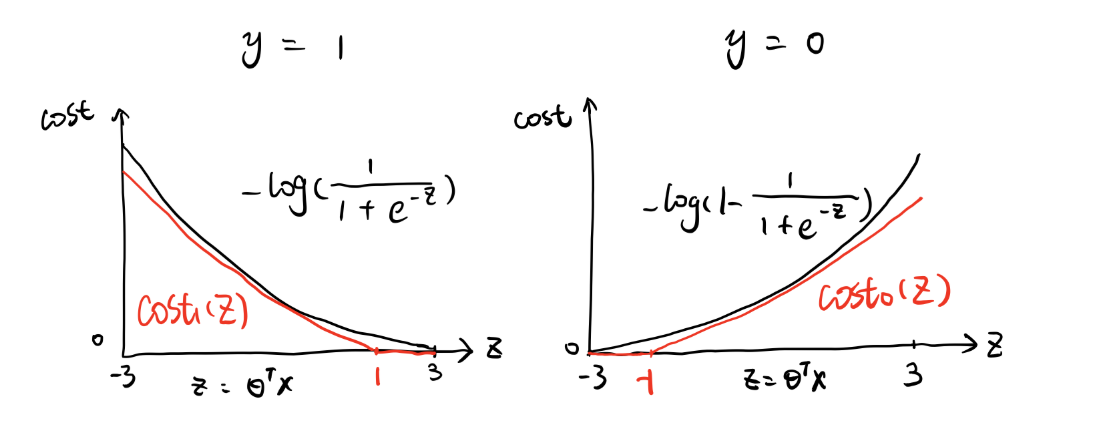


para el punto X1:



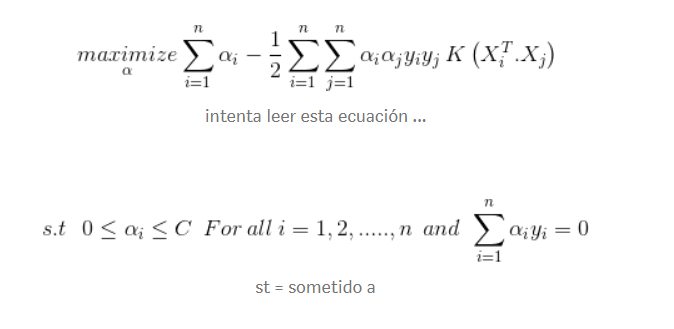
# 4. Función de Costo

El vector de soporte es una muestra que está clasificada incorrectamente o una muestra cercana a un límite. Mirando la trama a continuación. Las muestras con círculos rojos son exactamente el límite de decisión. En SVM, solo los vectores de soporte tienen un impacto efectivo en el entrenamiento del modelo, es decir, eliminar el vector sin soporte no tiene ningún efecto en el modelo. ¿Por qué? Lo resolveremos a partir de su función de costo.

La función de pérdida de SVM es muy similar a la de Regresión logística. Mirándolo por y = 1 e y = 0 por separado en la siguiente gráfica, la línea negra es la función de costo de la Regresión logística, y la línea roja es para SVM. Tenga en cuenta que el eje X aquí es la salida del modelo en bruto, θᵀx. Recuerde que poner la salida del modelo en bruto en la función Sigmoide nos da la hipótesis de la regresión logística. ¿Cuál es la hipótesis para SVM? Es simple y directo. Cuando θᵀx ≥ 0, predice 1, de lo contrario, predice 0.

# 5. ¿Qué es el truco de Kernel?

Llegando a la mayor parte del SVM por el que es más famoso, el truco del kernel . El núcleo es una forma de calcular el producto escalar de dos vectores x e y en algún espacio de características (de muy alta dimensión), razón por la cual las funciones del núcleo a veces se denominan "producto escalar generalizado".



Aplicar el truco del kernel significa solo reemplazar el producto punto de dos vectores por la función del kernel.

# 6. Tipos de Kernels

* núcleo lineal
* núcleo polinomial
* Núcleo de función de base radial (RBF) / Núcleo gaussiano

# 7. Diferencias con otros algoritmos

* Una de las principales diferencias entre este método de clasificación y KNN visto en clase, es que éste es un método paramétrico por lo que si requiere un proceso de entrenamiento.
* Con respoecto al algoritmo de regresión logística tradicional, este algoritmo puede realizar clasificación para multiples grupos de datos.
* Contrario al algoritmo de regresión logística(otro modelo de clasificación que si es paremétrico), este modelo no nos brinda una distribución de probabilidad en cuatno al valor de la variable a predecir.

# 8. Ventajas

* Es un algoritmo capaz de ser utilizado con pocos datos
* Es capaz de reconocer fronteras de decisión no lineales y con formas mas complejas sin necesidad de recurrir a Feature Engineering. Esto lo logra a través de sobredimensionarl os datos (kernel Trik)
* Es un algoritmo simple de implementar por lo que se puede utilizar en problemas que no requieran mucha precisión.
* Puede usarse también para regresión.
* Funciona para n>2 dimensiones, dado el concepto de hiperplano

# 9. Desventajas

* Debido a la implementación de Kernel Trick, también se tiene que utilizar mas recursos computacionales.
* Bajo desempeño cuando el número de Features es mayor que el número de muestras.
* No provee distribuciones de probabilidad.