# Minería de Datos para el Análisis de Big Data

Por: Carlos Carreño

ccarrenovi@Gmail.com

Abril,2021

# Modulo 11 Maquina de Vector Soporte

- Introducción
- Hiperplano y Minimal Margin Classifier
- Clasificación binaria usando el hiperplano
- Clasificador de Vector Soporte
- Maquinas de Vector Soporte

# Introducción

- El ***método de clasificación-regresión*** **Máquinas de Vector Soporte** (*Vector Support Machines, SVMs*) fue desarrollado en la década de los 90, dentro de campo de la ciencia computacional.
- Si bien originariamente se desarrolló como un método de clasificación binaria, su aplicación se ha extendido a ***problemas de clasificación múltiple y regresión***.
- En R, las librerías ***e1071*** y ***LiblineaR*** contienen los algoritmos necesarios para obtener modelos de clasificación simple, múltiple y regresión, basados en ***Support Vector Machines.***

# Hiperplano y Maximal Margin Classifier

- En un espacio ***p-dimensional***, un hiperplano se define como un subespacio plano y afín de dimensiones ***p−1***. El término afín significa que el subespacio no tiene por qué pasar por el origen. En un espacio de dos dimensiones, el hiperplano es un subespacio de 1 dimensión, es decir, una recta. En un ***espacio tridimensional***, un ***hiperplano*** es un subespacio de dos dimensiones, ***un plano convencional***.
- En dos dimensiones la ecuación del hiperplano será: $\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$
- Dados los parámetros $\beta_0$, $\beta_1$ y $\beta_2$, todos los pares de valores $x=(x_1,x_2)$ para los que se cumple la igualdad son puntos del hiperplano. Esta ecuación puede generalizarse para ***p-dimensiones***: $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p = 0$
- Todos los puntos definidos por el vector ($x=x_1,x_2,\ldots,x_p$) que cumplen la ecuación pertenecen al hiperplano.

- Cuando x no satisface la ecuación:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p < 0$$

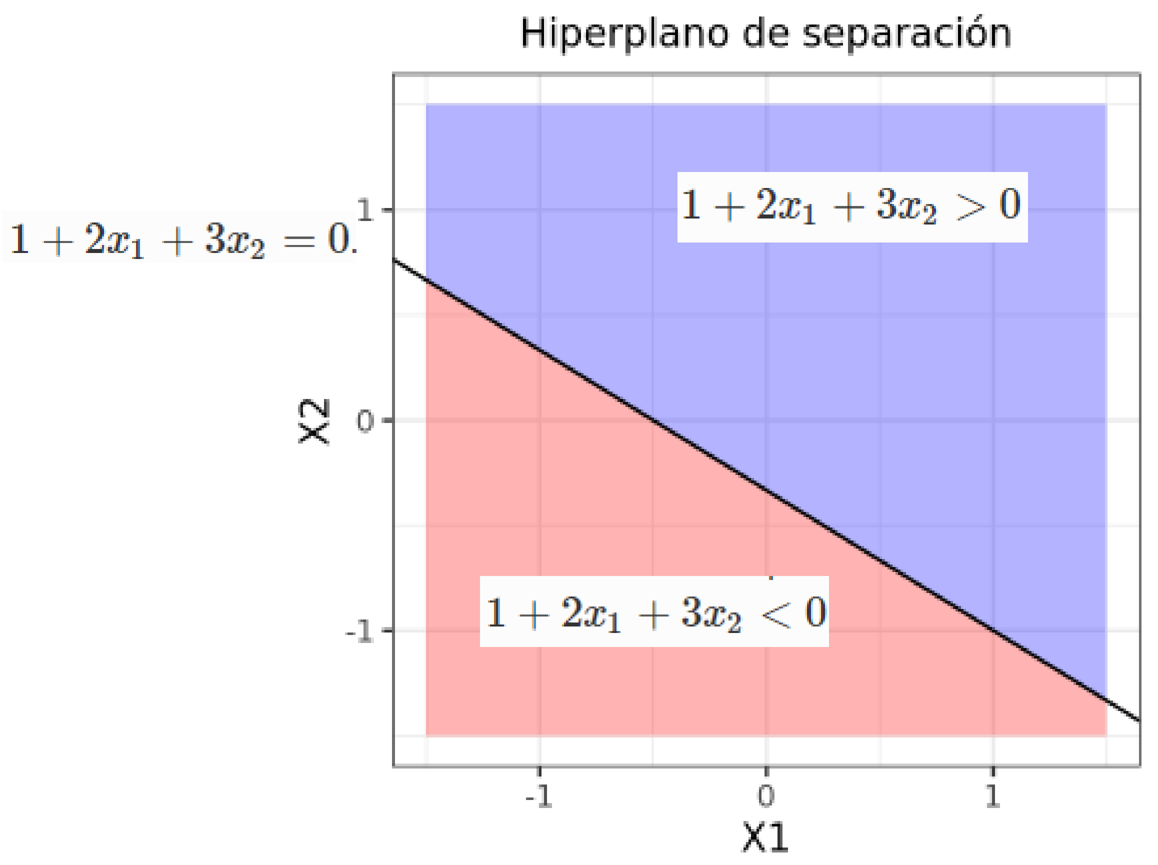
- o bien

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p > 0$$

- el punto x cae a un lado o al otro del hiperplano. Así pues, se puede entender que un hiperplano divide un espacio p-dimensional en dos mitades. Para saber en qué lado del hiperplano se encuentra un determinado punto x, solo hay que calcular el signo de la ecuación.

# Ejemplo: Hiperplano

- El hiperplano divide el espacio en dos mitades



# Clasificación binaria empleando un hiperplano

- Cuando se dispone de ***n observaciones***, cada una con ***p predictores*** y cuya variable respuesta tiene ***dos niveles*** (de aquí en adelante identificados como $+1$ y $-1$), se pueden emplear hiperplanos para ***construir un clasificador*** que permita predecir a que grupo pertenece una observación en función de sus predictores.

# CASOS PERFECTAMENTE SEPARABLES LINEALMENTE

- Si la distribución de las observaciones es tal que se pueden separar linealmente de forma perfecta en las dos clases (+1 y −1), entonces, un hiperplano de separación cumple que:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p > 0, \ si \ y_i = 1$$

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p < 0, \ si \ y_i = -1$$

- Al identificar cada clase como +1 o −1, y dado que multiplicar dos valores negativos resultan en un valor positivo, las dos condiciones anteriores pueden simplificarse en una única:

$$y_i(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p) > 0, \ \ para \ i = 1 \ldots n$$

- Bajo este escenario, el clasificador más sencillo consiste en asignar cada observación a una clase dependiendo del lado del hiperplano en el que se encuentre. Es decir, la observación ***x^**** se clasifica acorde al signo de la función $f(x^*)=\beta_0+\beta_1 x_1*+\beta_2 x_2*+...+\beta_p x_p*$. Si $f(x*)$ es positiva, la observación se asigna a la clase +1, si es negativa, a la clase −1. Además, la magnitud de $f(x*)$ permite saber cómo de lejos está la observación del hiperplano y con ello la confianza de la clasificación.

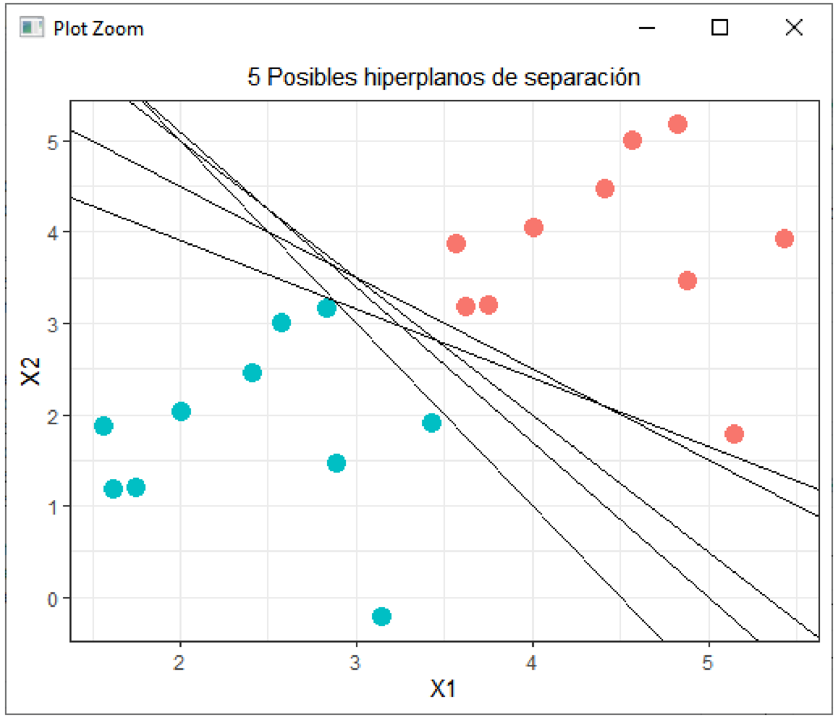
# Hiperplano optimo de separación

- La definición de hiperplano para casos perfectamente separables linealmente resulta en un número infinito de posibles hiperplanos, lo que hace necesario un método que permita seleccionar uno de ellos como clasificador óptimo.

```r
library(ggplot2)
set.seed(68)
X1 <- rnorm(n = 10, mean = 2, sd = 1)
X2 <- rnorm(n = 10, mean = 2, sd = 1)

observaciones <- data.frame(X1 = c(X1, X1 + 2), X2 = c(X2, X2 + 2) ,
                            clase = rep(c(1, -1), each = 10))
observaciones$clase <- as.factor(observaciones$clase)

ggplot() +
  geom_point(data = observaciones, aes(x = X1, y = X2, color = clase), size = 4) +
  geom_abline(intercept = 9, slope = -2) +
  geom_abline(intercept = 8.5, slope = -1.7) +
  geom_abline(intercept = 8, slope = -1.5) +
  geom_abline(intercept = 6.5, slope = -1) +
  geom_abline(intercept = 5.4, slope = -0.75) +
  theme_bw() +
  labs(title = "5 Posibles hiperplanos de separación") +
  theme( legend.position = "none",
         plot.title = element_text(hjust = 0.5, size = 11))
```



- La solución a este problema consiste en seleccionar como clasificador óptimo al que se conoce como ***maximal margin hyperplane*** o hiperplano óptimo de separación, que se corresponde con el hiperplano que se encuentra más alejado de todas las observaciones de entrenamiento. Para obtenerlo, se tiene que calcular la distancia perpendicular de cada observación a un determinado hiperplano. La menor de estas distancias (conocida como margen) determina como de alejado está el hiperplano de las observaciones de entrenamiento.
- El maximal margin hyperplane se define como el hiperplano que consigue un mayor margen, es decir, que la distancia mínima entre el hiperplano y las observaciones es lo más grande posible.
- Aunque esta idea suena razonable, no es posible aplicarla, ya que habría infinitos hiperplanos contra los que medir las distancias. En su lugar, se recurre a métodos de optimización.

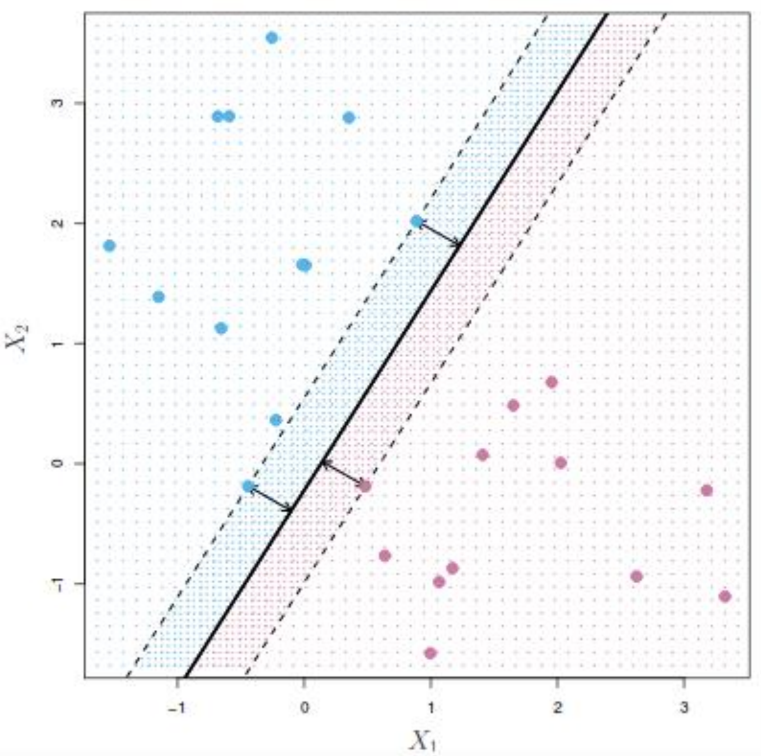
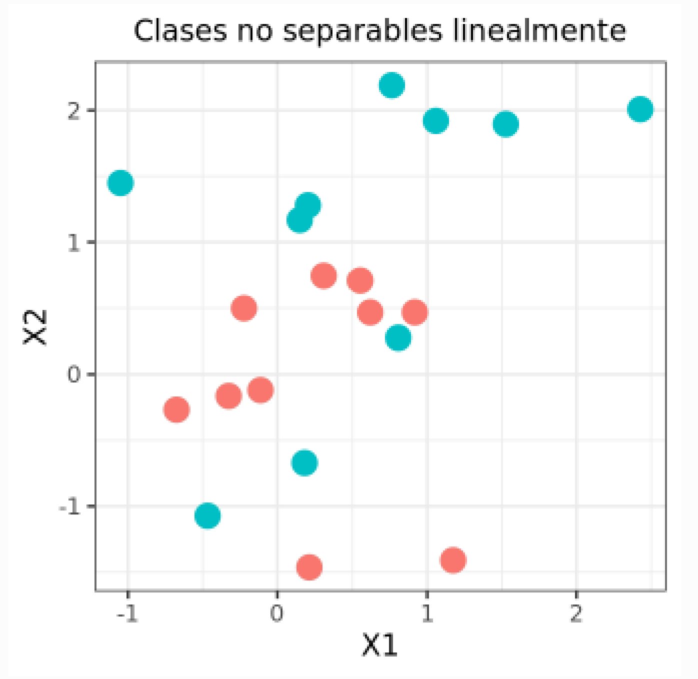


*Imagen maximal margin hyperplane obtenida del libro ISLR*

- La imagen anterior muestra el maximal margin hyperplane para un conjunto de datos de entrenamiento.
- Las tres observaciones equidistantes respecto al maximal margin hyperplane se encuentran a lo largo de las líneas discontinuas que indican la anchura del margen.
- A estas observaciones se les conoce como vectores soporte, ya que son vectores en un espacio p-dimensional y soportan (definen) el maximal margin hyperplane.
- Cualquier modificación en estas observaciones (vectores soporte) conlleva cambios en el maximal margin hyperplane. Sin embargo, modificaciones en observaciones que no son vector soporte no tienen impacto alguno en el hiperplano.

# CASOS CUASI-SEPARABLES LINEALMENTE

- En la gran mayoría de casos reales, los datos no se pueden separar linealmente de forma perfecta, por lo que no existe un hiperplano de separación y no puede obtenerse un maximal margin hyperplane.
- Para solucionar estas situaciones, se puede extender el concepto de maximal margin hyperplane para obtener un hiperplano que casi separe las clases, pero permitiendo que cometa unos pocos errores. A este tipo de hiperplano se le conoce como Support Vector Classifier o Soft Margin.



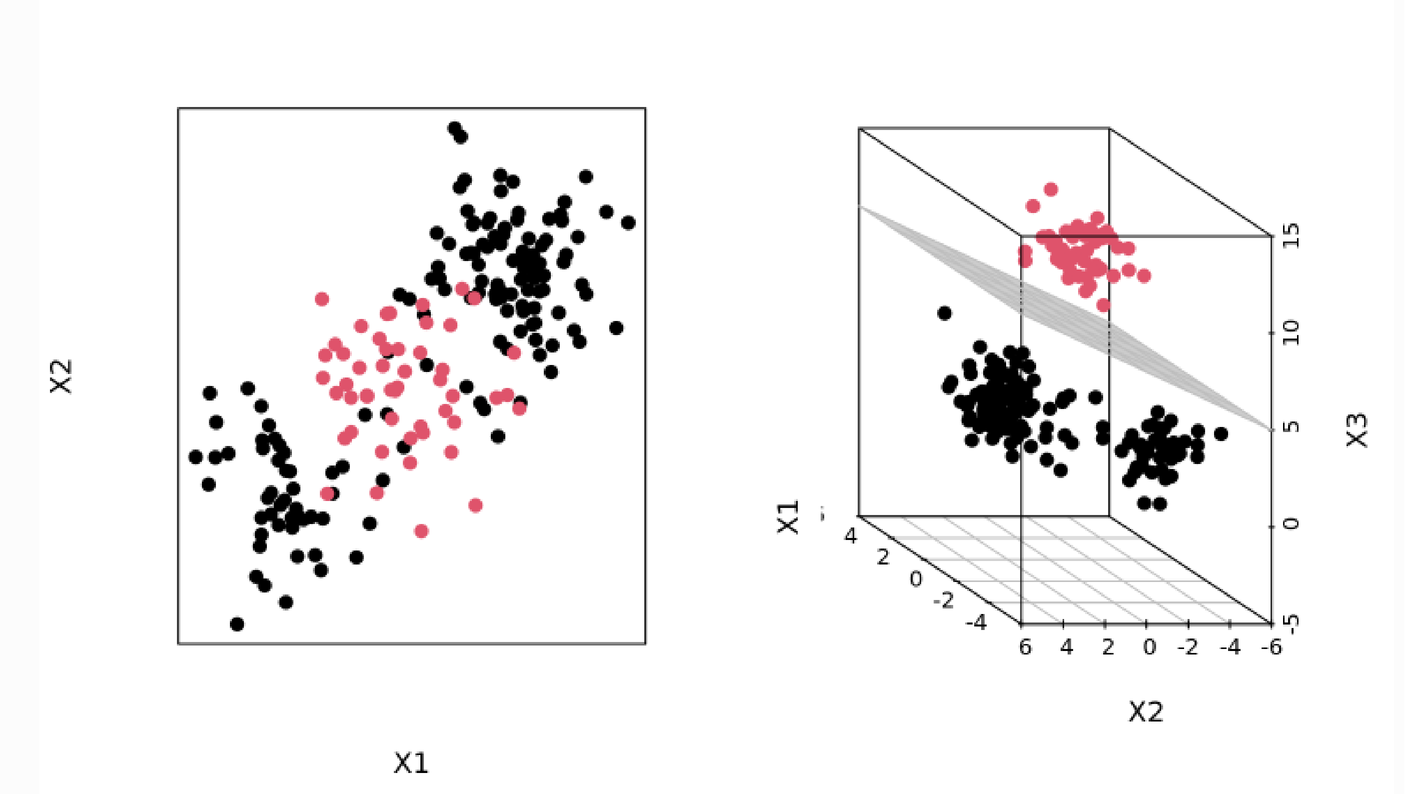
# Support Vector Classifier o Soft Margin SVM

- Es preferible crear un clasificador basado en un hiperplano que, aunque no separe perfectamente las dos clases, sea más robusto y tenga mayor capacidad predictiva al aplicarlo a nuevas observaciones (menos problemas de overfitting). Esto es exactamente lo que consiguen los clasificadores de vector soporte, también conocidos como soft margin classifiers o Support Vector Classifiers.
- Para lograrlo, en lugar de buscar el margen de clasificación más ancho posible que consigue que las observaciones estén en el lado correcto del margen; se permite que ciertas observaciones estén en el lado incorrecto del margen o incluso del hiperplano.
- La identificación del hiperplano de un clasificador de vector soporte, que clasifique correctamente la mayoría de las observaciones a excepción de unas pocas, es un problema de optimización convexa.

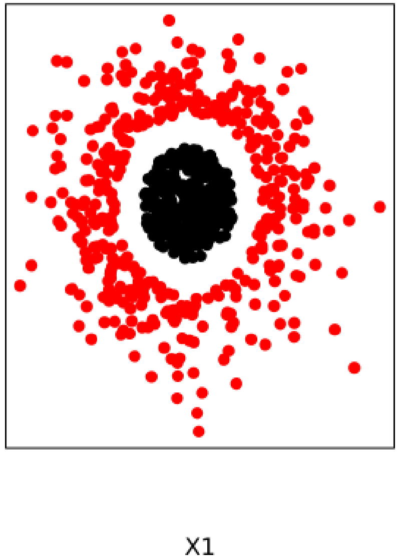
# Máquinas de Vector Soporte

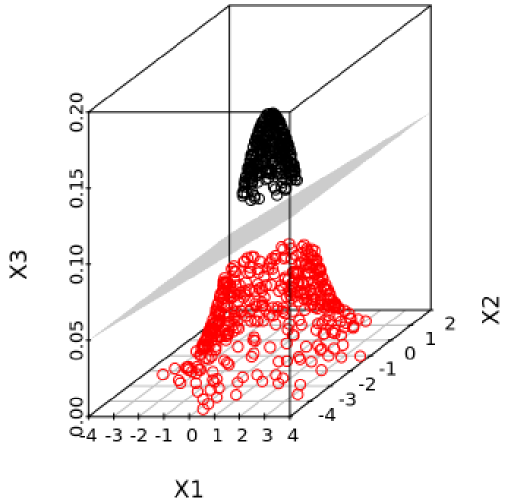
- El Support Vector Classifier descrito en los apartados anteriores consigue buenos resultados cuando el límite de separación entre clases es aproximadamente lineal. Si no lo es, su capacidad decae drásticamente. Una estrategia para enfrentarse a escenarios en los que la separación de los grupos es de tipo no lineal consiste en expandir las dimensiones del espacio original.

- Las imágenes siguientes muestran como dos grupos, cuya separación en dos dimensiones no es lineal, sí lo es al añadir una tercera dimensión.



- El método de Máquinas Vector Soporte (SVM) se puede considerar como una extensión del Support Vector Classifier obtenida al aumentar la dimensión de los datos. Los límites de separación lineales generados en el espacio aumentado se convierten en límites de separación no lineales al proyectarlos en el espacio original.



# Aumento de la dimensión, kernels

- Un kernel (K) es una función que devuelve el resultado del dot product entre dos vectores realizado en un nuevo espacio dimensional distinto al espacio original en el que se encuentran los vectores.

# Kernel lineal

- Si se emplea un Kernel lineal, el clasificador Support Vector Machine obtenido es equivalente al Support Vector Classifier.

$$K(\mathbf{x}, \mathbf{x'}) = \mathbf{x} \cdot \mathbf{x'}$$

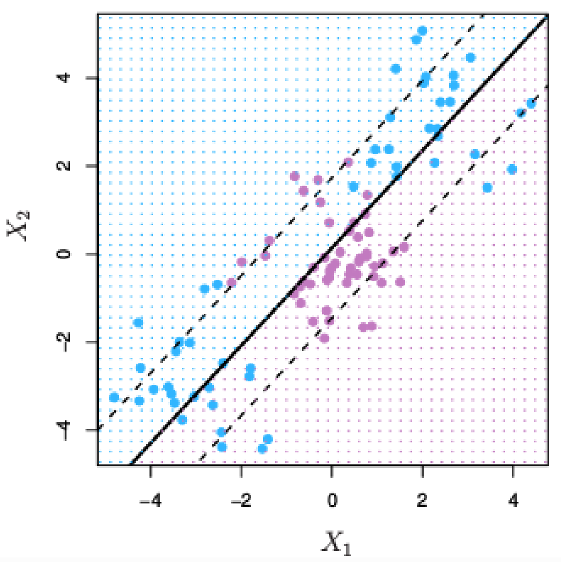



*Imagen SVM con kernel lineal obtenida del libro ISLR*

# Kernel polinómico

- Cuando se emplea d=1 y c=0, el resultado es el mismo que el de un kernel lineal. Si d>1, se generan límites de decisión no lineales, aumentando la no linealidad a medida que aumenta d. No suele ser recomendable emplear valores de d mayores 5 por problemas de overfitting.

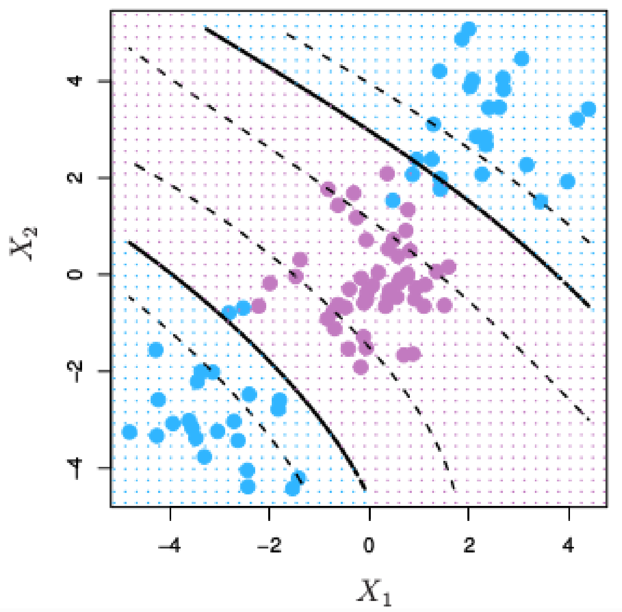



Imagen SVM con una kernel polinómico de grado 3 obtenida del libro ISLR

# Gaussian Kernel (RBF)

- El valor de γ controla el comportamiento del kernel, cuando es muy pequeño, el modelo final es equivalente al obtenido con un kernel lineal, a medida que aumenta su valor, también lo hace la flexibilidad del modelo.

$$K(\mathbf{x}, \mathbf{x}') = exp(-\gamma||\mathbf{x} - \mathbf{x}'||^2)$$

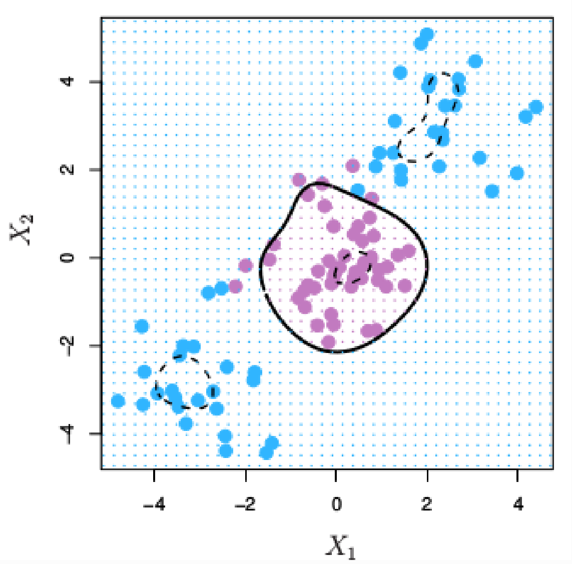
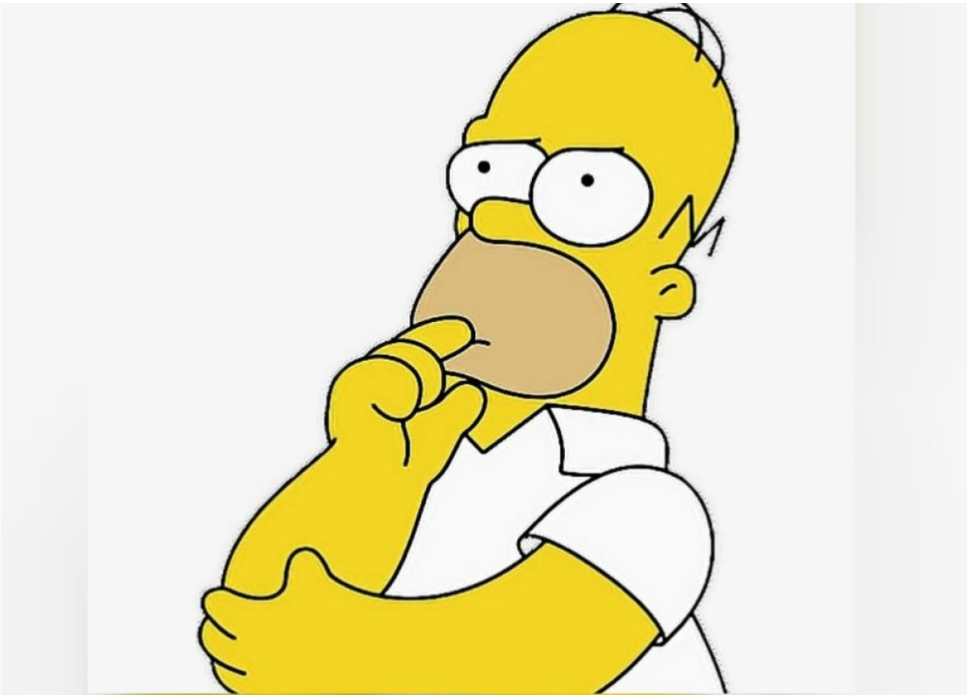



Imagen SVM con una kernel radial obtenida del libro ISLR

# Preguntas

- Alguna pregunta?



# Demo

- Realizar el ejercicio de Maquinas de Vector Soporte