**Práctica 6**

**Pablo Arranz Ropero**

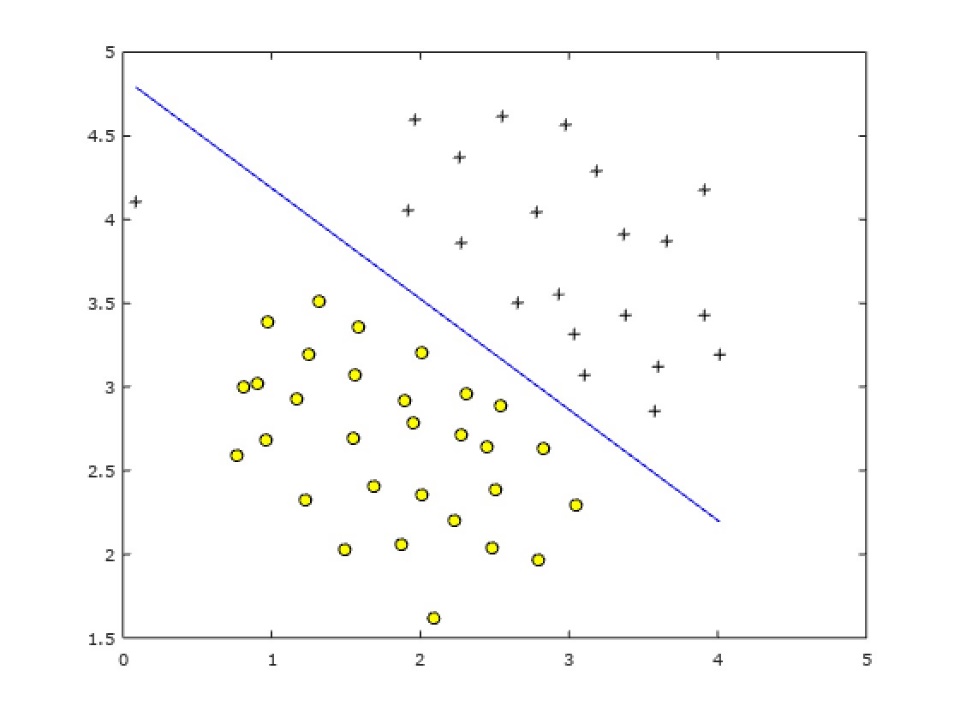
**Juan Alberto Camino Sáez**

**Grupo 2**

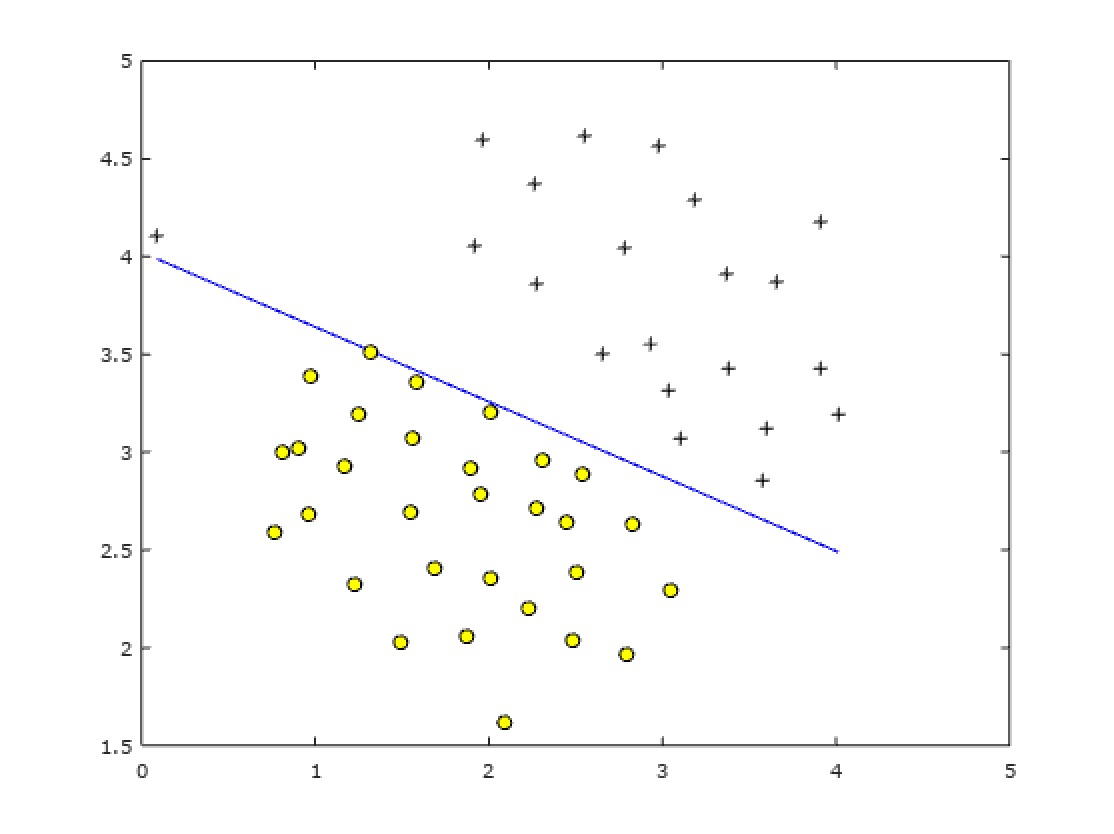
**Práctica 6: Support Vector Machines**

En esta práctica nos familiarizaremos con el uso del clasificador SVM para aplicarlo en la segunda parte de la práctica.

En primer lugar, hemos aplicado SVM para comprobar el efecto del parámetro C. En las siguientes gráficas se observa la separación aplicando distintos valores de C sobre unos datos linealmente separables. En primer lugar, con C = 1

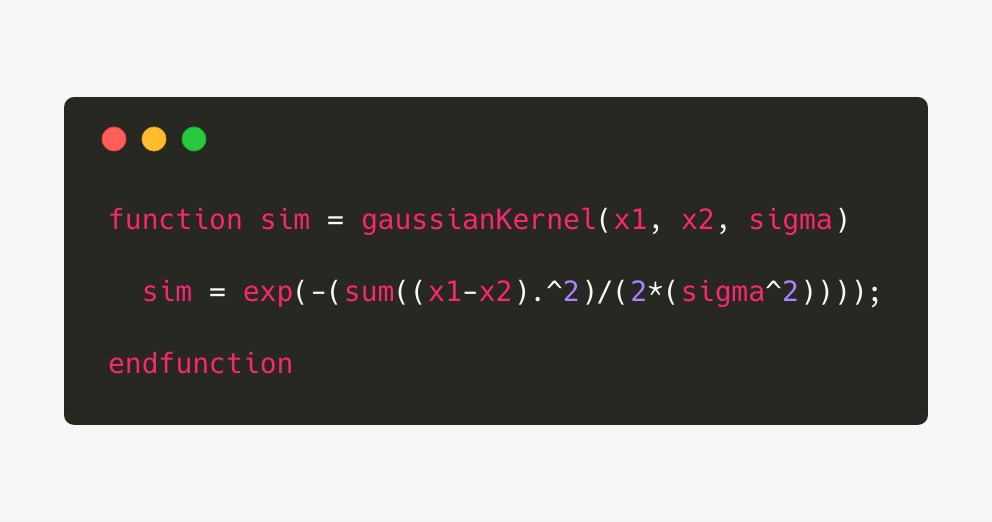


En el siguiente caso, con C = 100:

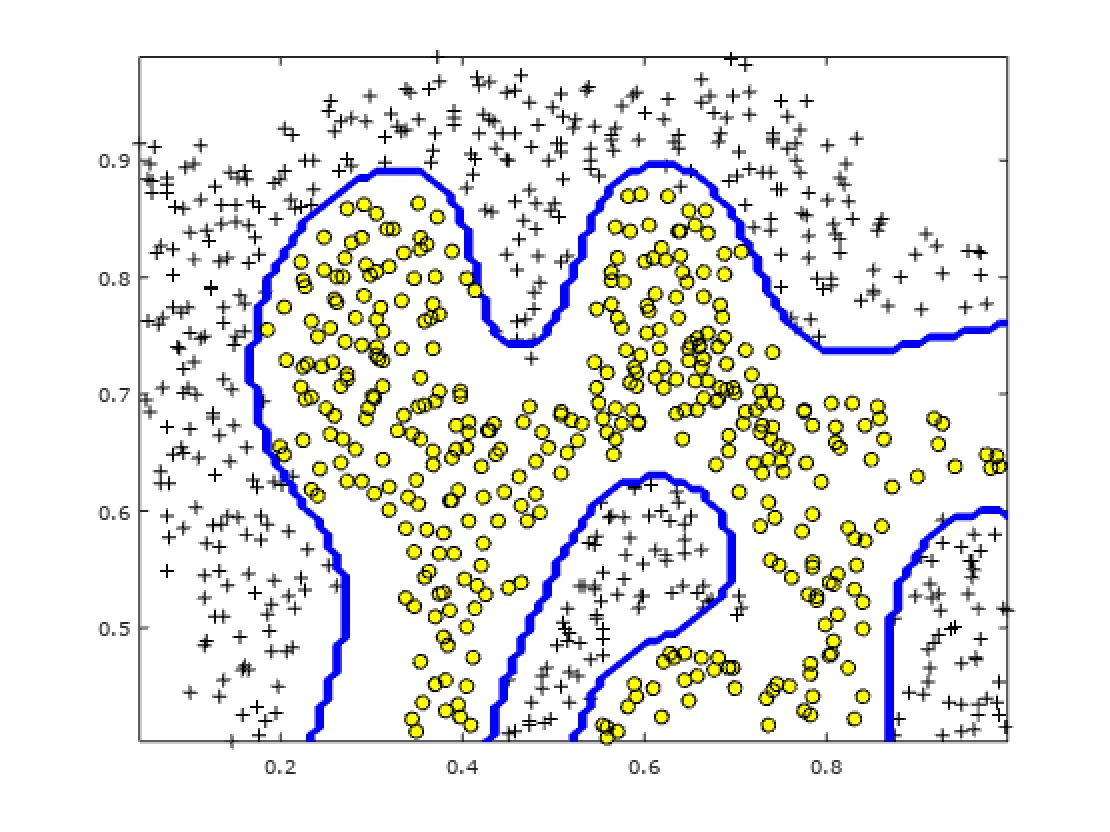


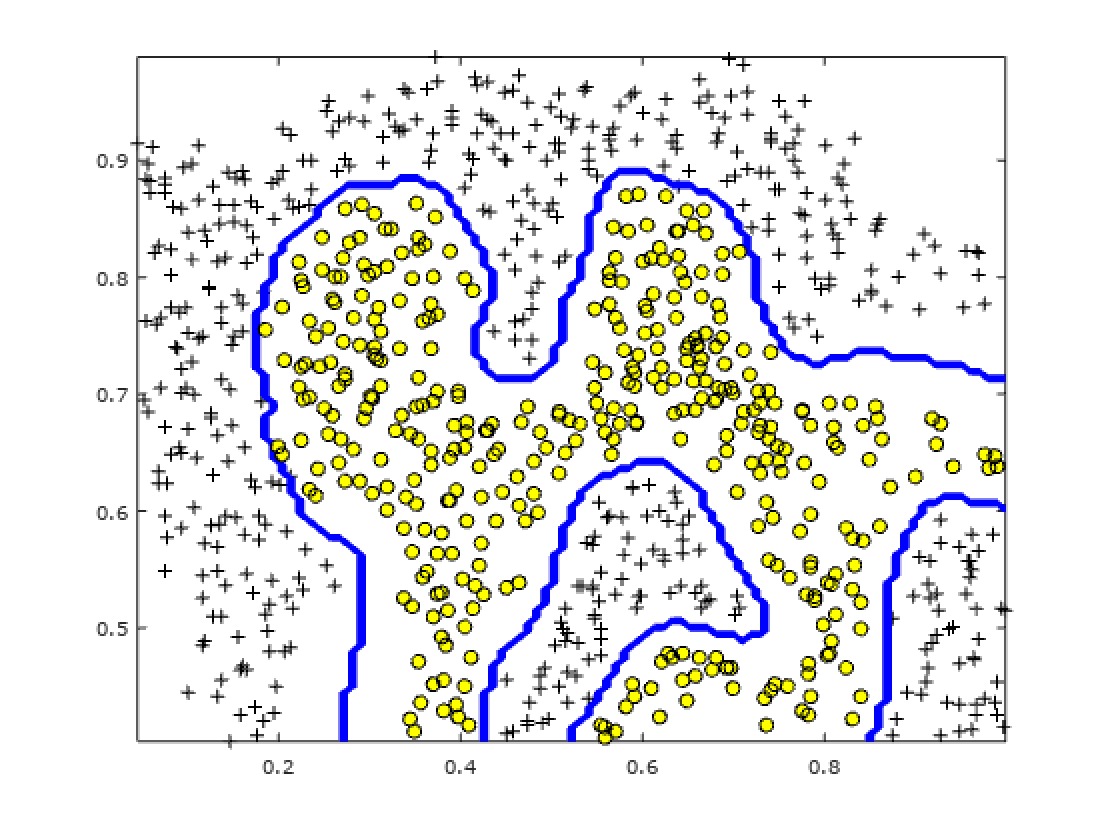
**KERNEL GAUSSIANO**

A continuación, procedemos a realizar un kernel Gaussiano implementando la función *gaussianKernel.m,* al que se le pasan x1, x2 y sigma y te devuelve el kernel Gaussiano entre los dos puntos.

Este el código de la función: 

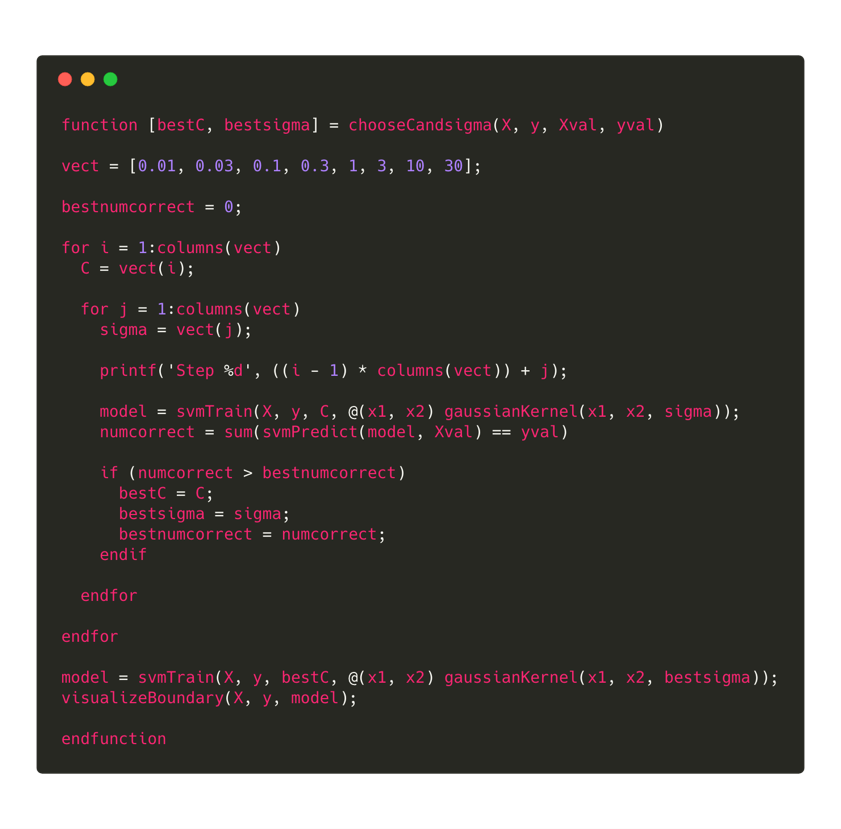
Además, tras aplicar SVM con el kernel Gaussiano, nos presenta la siguiente separación en la gráfica. Usando C = 1 y :



En el siguiente caso. Usamos C = 1 y . Se puede observar un poco más de overfitting:

**ELECCIÓN DE LOS PARÁMETROS C Y**

En la siguiente parte de la práctica hemos creado una función llamada *chooseCandsigma.m,* al que le pasamos los datos de entrenamiento, de validación, el inicio de C y y el número de pasos (veces que hay que multiplicar por 3 C y , y se encarga de calcular la similitud de todos los modelos diferentes con los datos de validación y devuelve el mejor C y el mejor . Este es el código:



Los mejores valores de C y calculados son 1 y 0.1 respectivamente. Las últimas líneas de nuestra función pintan la gráfica usando el modelo elegido y los datos proporcionados quedando lo mostrado en la siguiente imagen:

