Aprendizaje automático

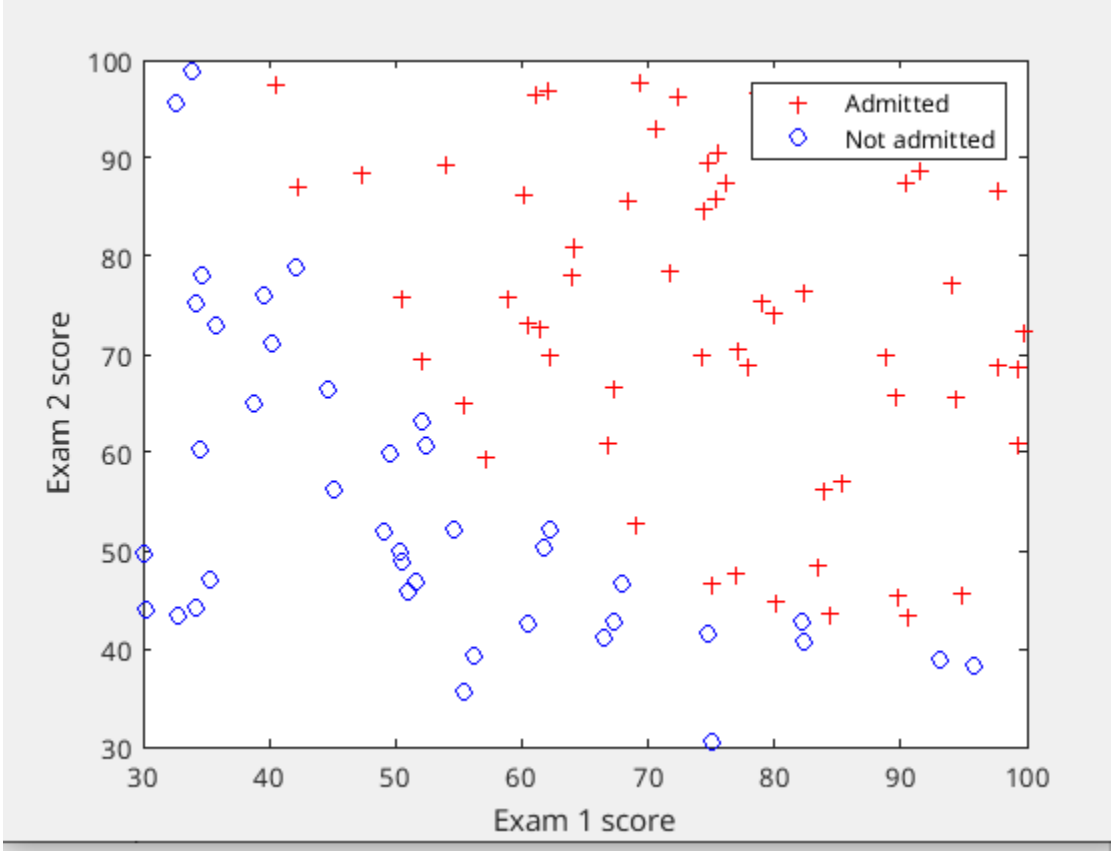
- práctica 3 -

Por:

Diego Caballé Casanova (738712)

Apartado 2

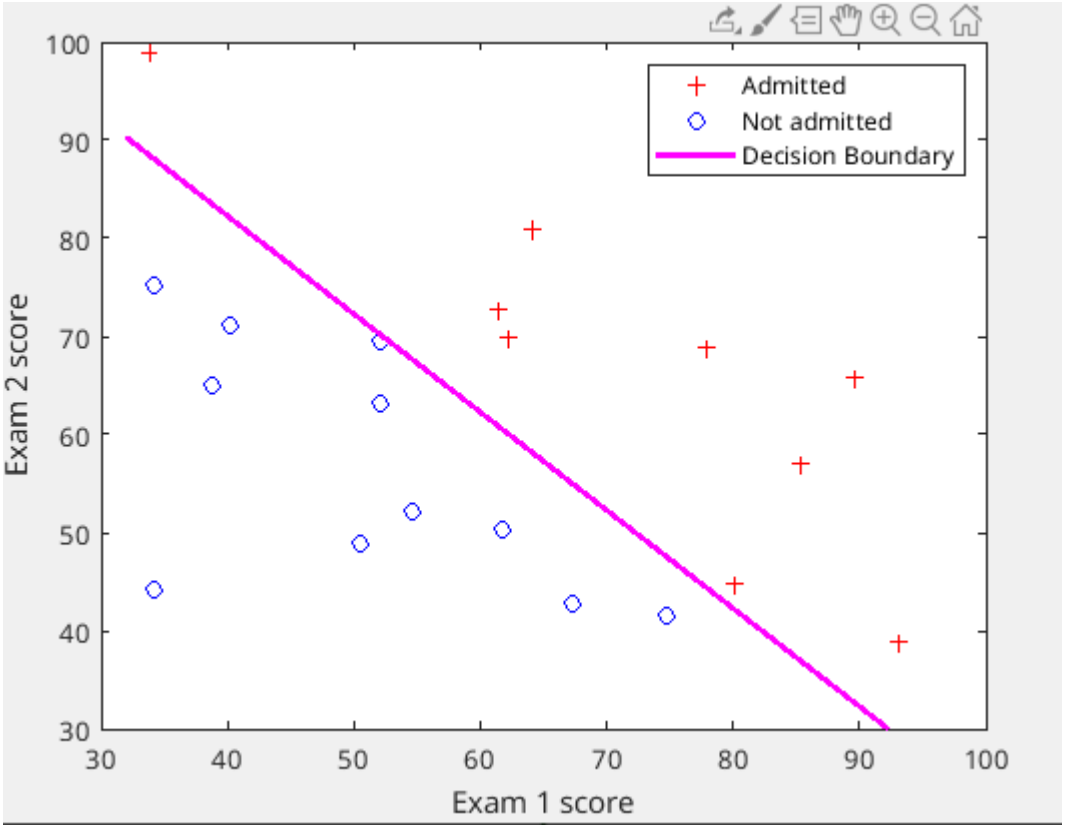
En este primer apartado, tenemos que clasificar por regresión logística un conjunto de datos, que representan las notas en dos exámenes. Los datos son estos:



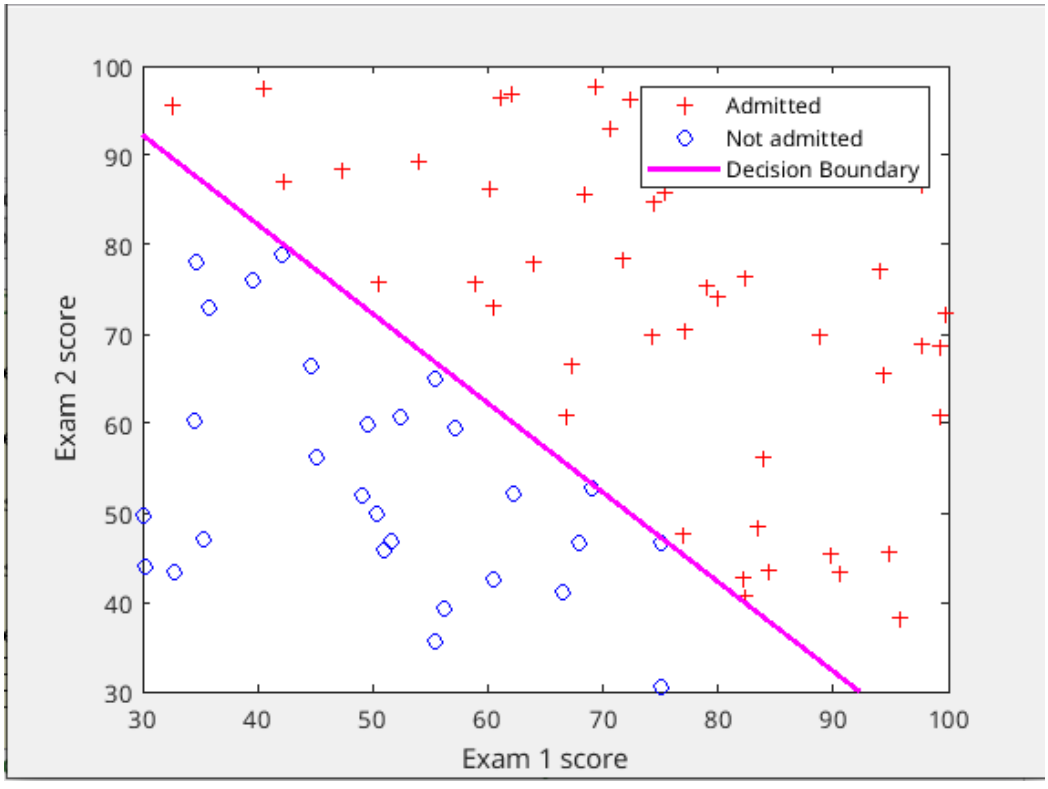
como podemos intuir, hay una división entre los aprobados y los suspensos y podemos intuir ya la frontera entre aprobar y suspender, pero dejaremos que esta frontera la decida el algoritmo.

Para empezar, he utilizado minFunc para resolver la regresión de forma más optima, que no utilizando el algoritmo de descenso de gradiente. De esta manera, obtengo el mejor theta posible que será la frontera de los datos, que utilizaremos para crear la hipotesis y así llevar a cabo la clasificación.

Podemos ver la frontera que ha decidido el algoritmo sobre los datos utilizados para el test:

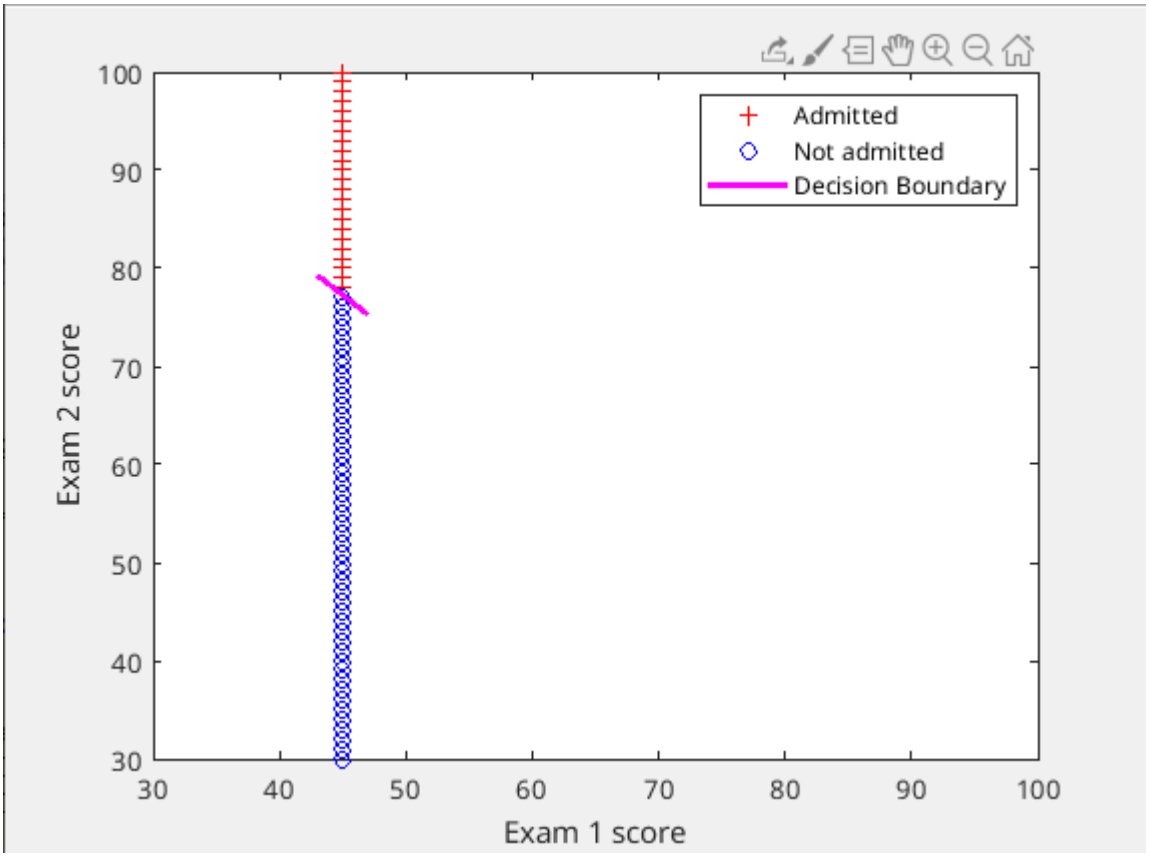


Aquí la frontera la tenemos sobre los datos de Train:



Esta frontera, nos da una tasa de error del 10% sobre los datos de entrenamiento y del 15% sobre los datos de test. Una tasa bastante buena, ya que la frontera era bastante sencilla.

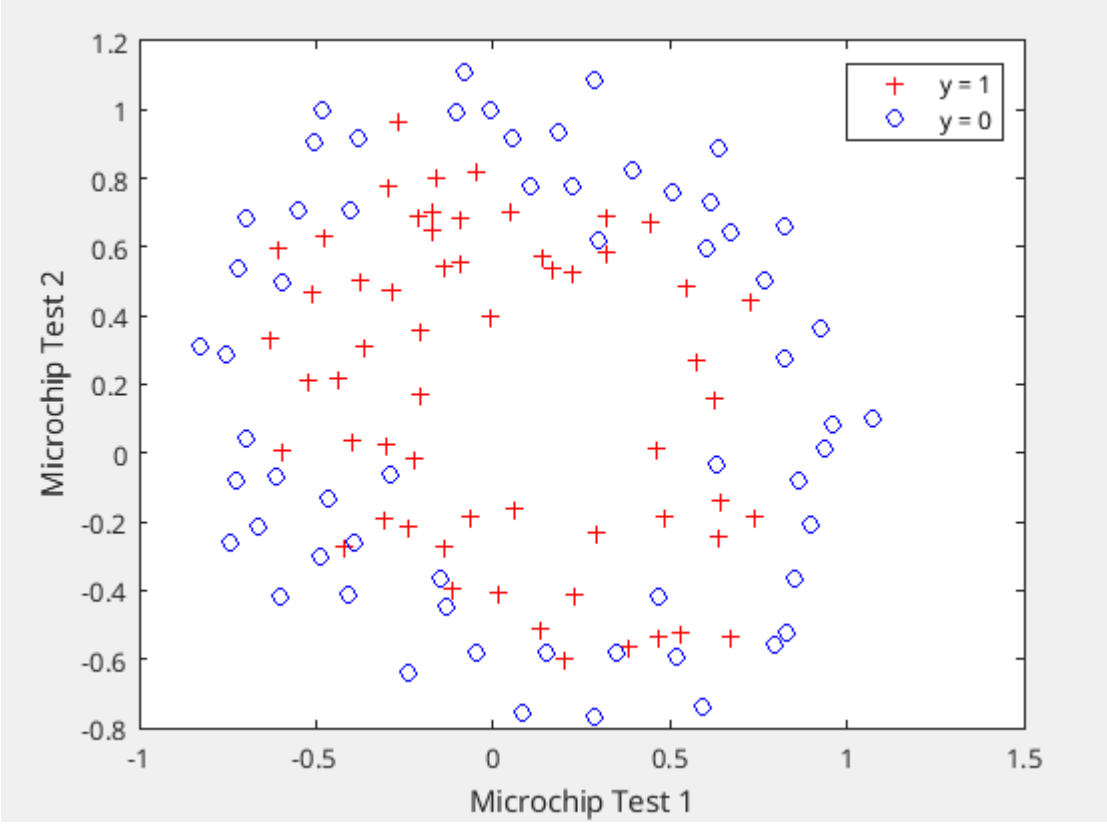
Después se nos pregunta que representemos la probabilidad de aprobar sacando un 45 en el primer examen. La gráfica quedaría así:



Como vemos, tenemos que sacar cerca de un 78 para poder aprobar.

Apartado 3

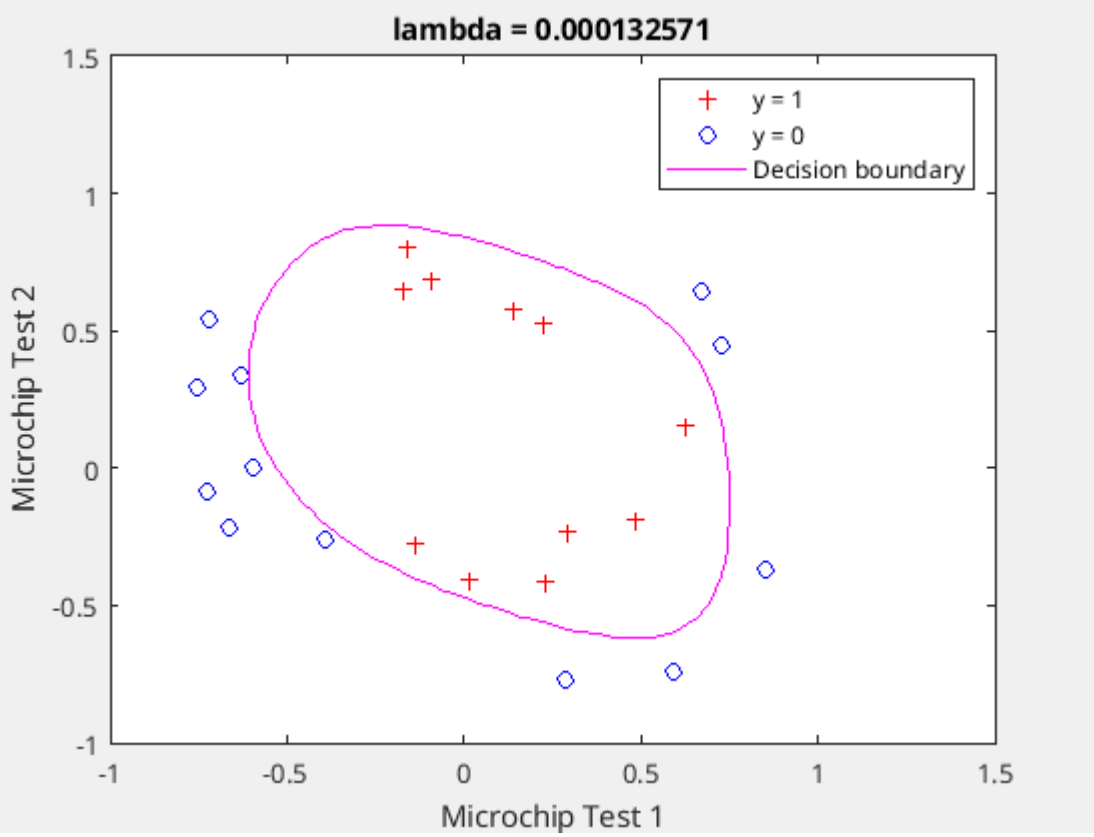
En este caso tenemos una fábrica de microchips y la posibilidad de aceptar unos o desecharlos.



En este caso, la frontera no es tan clara, y podríamos sufrir de sobreajuste porque es un círculo ciertamente extraño.

En este caso, usaremos regresión logística regularizada, luego lo primero que haremos será encontrar el mejor lambda para nuestro problema y evitar el sobreajuste.

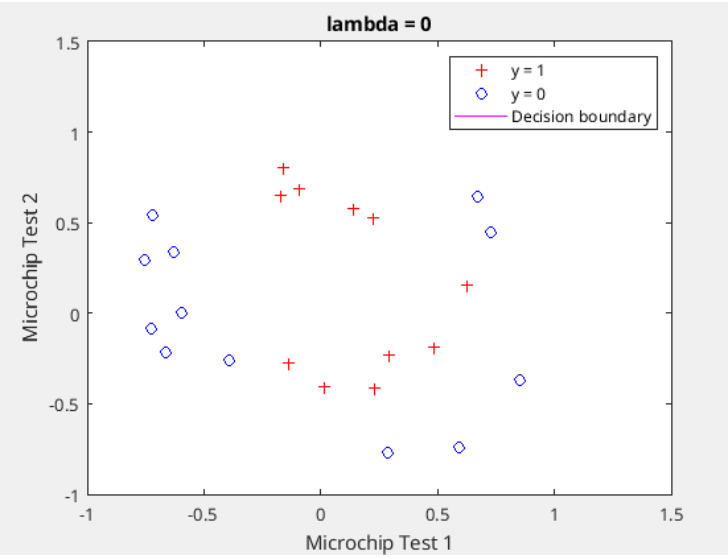
El lambda obtenido es 1.325711365590108e-04 y nos da una frontera de decisión de este tipo:



Este lambda, además, nos da una tasa de error del 16% sobre los datos de entrenamiento y del 13% sobre los datos de test, de nuevo, una tasa bastante aceptable.

También se nos pide comparar este modelo con uno con lambda = 0

En este caso, el sobreajuste es altisimo y la frontera ni si quiera se dibuja, pues el theta entrenado está sobreajustado a los datos que se dan, luego no podría clasificar más que estos datos. La gráfica queda así:



Obviamente, una frontera de decisión tan sobreajustada, tiene una tasa de error demasiado alta, en este caso de un 53% con los datos de entrenamiento y de un 39% con los datos de test.

De esta manera concluimos que el modelo con la mejor lambda, la que permite reducir el sobreajuste, es el mejor modelo de los dos.

Apartado 4

En este apartado, después de decidir el mejor modelo, tenemos que calcular su matriz de confusión y sus valores de precision, recall.

Usando los datos de test obtenemos esta matriz de confusión:

```
matrizDeConfusion =

     9     0     9
     3    11    14
    12    11    23
```

Vemos que hay ligeros errores de clasificación, de ahí la bajisima tasa de error.

Los valores de precision y recall son de 0.8750 y 0.8929 respectivamente.

Y ante la pregunta de que habría que hacer para que el 95% de los chips aceptados fueran buenos, habría que reducir los false negatives. O en otras palabras aumentar el recall a 95%.