

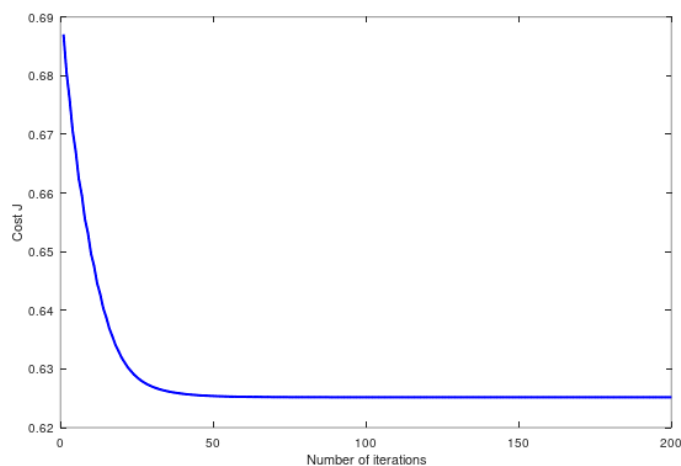


Estudiante 1: Germán Alejo Domínguez
Estudiante 2: Sebastián Pedrosa Granados

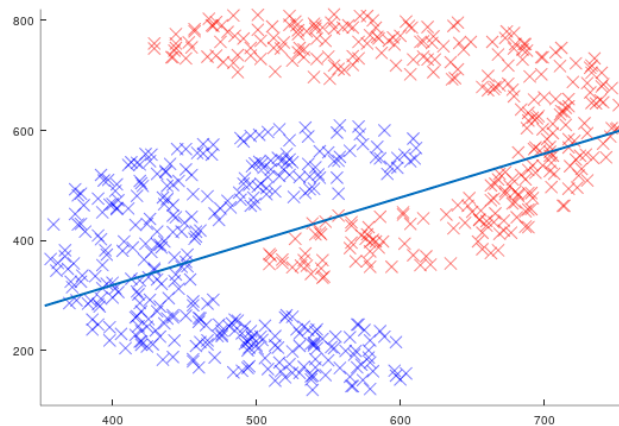
1. Implementar una regresión logística. En este caso no se hará una evaluación del modelo mediante un conjunto de test sino que simplemente queremos analizar como la frontera de decisión aprendida a partir de todo el conjunto de datos separa las dos clases representadas por color azul y rojo. Para ello, obtenga un modelo usando todo el conjunto de datos y haga una predicción de todo el conjunto de datos, imprimiendo por pantalla la tasa de acierto y la gráfica con la frontera de decisión usando para ello la función `plotDecisionBoundary`.

- Alfa: 0.000015
- Número de iteraciones: 200
- Tasa acierto: 54.29

Mostrar la gráfica de convergencia de descenso del gradiente



Mostrar una gráfica con los datos y la frontera de decisión obtenida





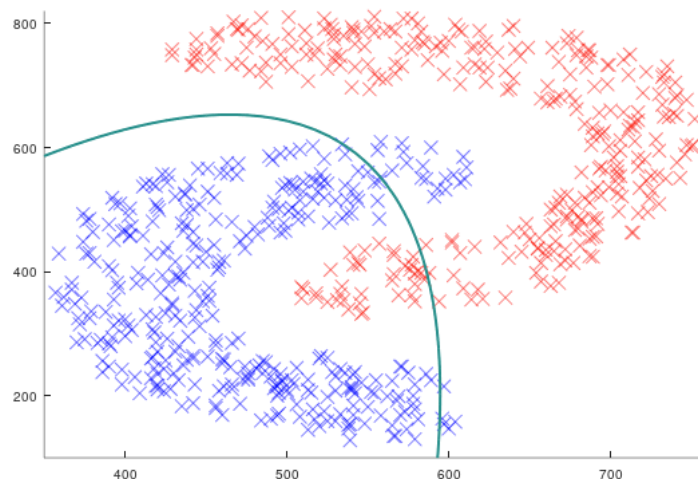
¿Crees que podrías mejorar estos resultados? Responda en una única línea cómo.

Se podrían mejorar los resultados consiguiendo una frontera de decisión curva, es decir, obteniendo una ecuación de grado $n > 1$.

2. Como se puede comprobar en la figura del apartado 2, los datos no son separables linealmente y se necesita una frontera de decisión mucho más compleja que una simple recta. Una solución es crear más atributos mediante la función `mapFeature.m`, que mapea los atributos en términos polinomiales de x_1 y x_2 hasta el grado 2 (x_1 , x_2 , x_1^2 , x_2^2 , $x_1 \cdot x_2$) y devuelve una matriz con 6 atributos (incluyendo el atributo $x_0=1$), permitiendo de esta manera obtener una frontera de decisión definida por una curva de grado 2 que se pueda adaptar mejor a los puntos. Modificar la regresión logística del apartado 2, para obtener un modelo a partir de los datos mapeados con 6 atributos. Una vez obtenido el modelo, haga una predicción de todo el conjunto de datos, imprimiendo por pantalla la tasa de acierto y la gráfica con la frontera de decisión. En este apartado, usar una función de optimización avanzada para obtener los parámetros θ óptimos.

- Número de iteraciones: 4000
- Tasa acierto: 90.680473

Mostrar una gráfica con los datos y la frontera de decisión obtenida





Observe la frontera de decisión obtenida, y responda a las siguientes preguntas:

1) ¿Crees que podrías mejorar estos resultados (la tasa de acierto)? Responda: SI/NO

SI

En caso de respuesta afirmativa:

2) ¿Cómo se podrían mejorar los resultados? Responda en una única línea

En esta ocasión se ha empleado una frontera de decisión de grado 2, si aumentásemos el grado de la frontera podríamos ajustar más esta a los datos pero también podría producirse un overfitting dada la tasa de acierto de la de grado 2.

3) ¿Qué líneas de código debes modificar para hacer lo que has indicado en la pregunta 2)?

En la función `mapFeature` debería cambiar el valor de la variable `degree`, que se encuentra por defecto en 2, por el grado escogido para el ajuste.