

Lista 2

Regressão Logística

Pedro Henrique G. Carvalho (362982)

Instruções

Deverá ser enviado ao professor, um arquivo texto contendo os gráficos, resultados e comentários requeridos em cada item.

1. Regressão Logística

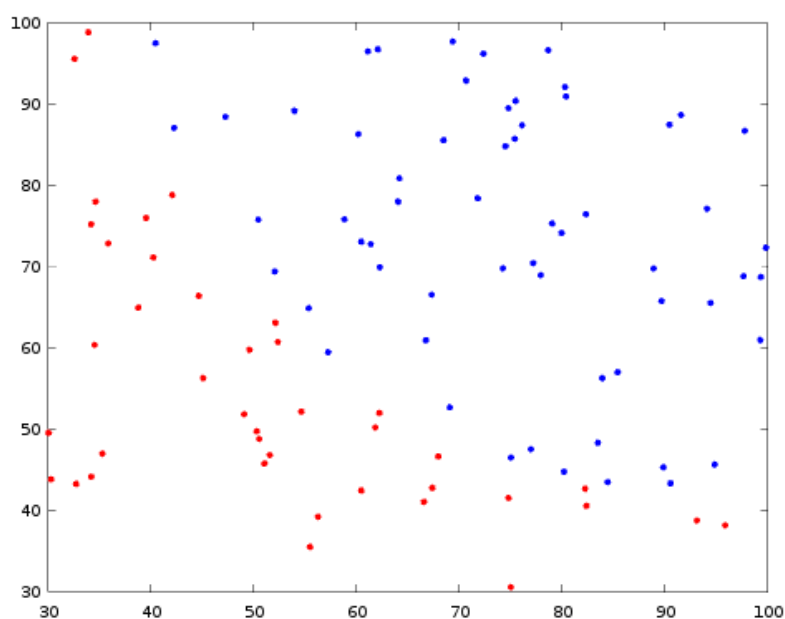
- Carregue os dados contidos no arquivo ex2data1.txt.

O arquivo contém 100 linhas e 3 colunas de dados. Cada coluna se refere a uma variável. Neste problema, deve-se desenvolver um modelo de classificação capaz de reproduzir as classes apresentadas na terceira coluna dos dados.

O problema consiste em um sistema de admissão em de alunos em uma universidade. Os dados das colunas 1 e 2 representam as notas de cada aluno em dois testes. A coluna 3 indica se este aluno foi ou não admitido na universidade.

Os dados apresentados são dados históricos de alunos aceitos ou não. Deseja-se fazer um sistema que faça a avaliação dos alunos automaticamente.

Apresentar: Figura com os dados

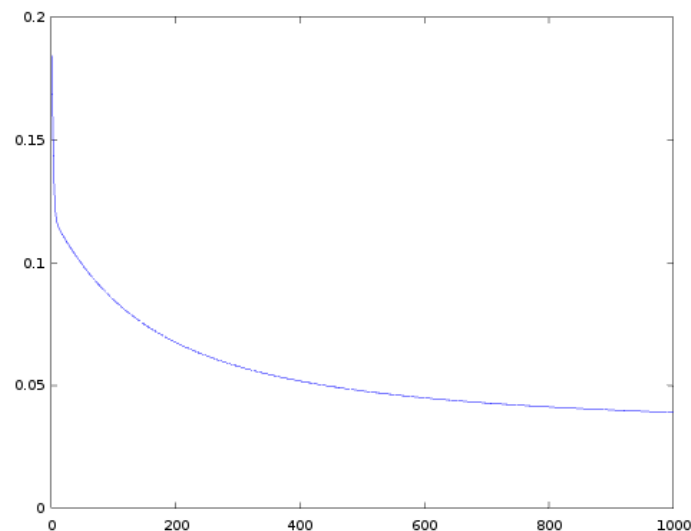


- Divida o conjunto de dados entre treino e teste. Para este problema, utilize 70 dados para treino e o restante para teste
- Implemente o algoritmo do gradiente descendente estocástico para encontrar os coeficientes do classificador

Para este algoritmo utilize $\alpha = 0.01$ e 1000 épocas de treinamento. Para cada época de treinamento, calcule o erro de classificação no conjunto de teste. Plote o gráfico “épocas x Erro”

Apresentar: Valor final dos coeficientes, o gráfico épocas x Erro e o valor final do erro de classificação para o conjunto de testes

$$\Rightarrow w = [-10.0396, 8.7152, 7.2491]'$$



Comentários: Através do gráfico “épocas x Erro” é possível verificar que o algoritmo está “aprendendo” ? Comente.

\Rightarrow O EQM está diminuindo com o passar das épocas (Até se tornar mais estável), o que mostra que, a cada época que passa, as aproximações estão ficando cada vez melhores, portanto, podemos inferir que o algoritmo está aprendendo.

- Construa um modelo utilizando o algoritmo do gradiente descendente estocástico e utilize o k-fold para validação cruzada do resultado.

Para este algoritmo utilize $\alpha = 0.01$ e 1000 épocas de treinamento.

Apresentar: Valor final dos coeficientes

=> Separei os dados, de maneira aleatória, em dois grupos, um com 80 dados para treino e um com 20 dados para teste. No fim da execução, obtive:

$$w = [-10.5552, 8.6017, 8.3597]'$$

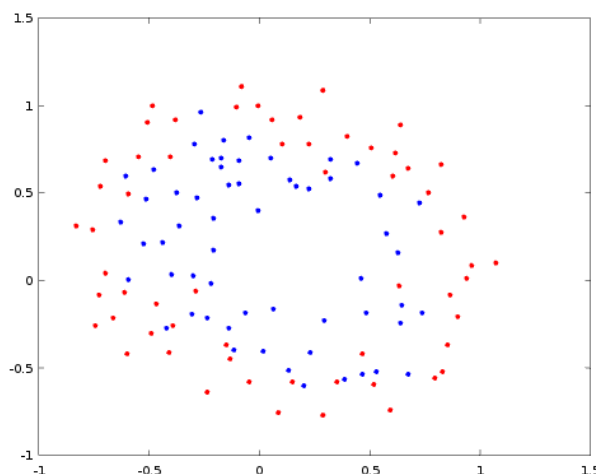
2. Regressão Logística Regularizada

- Carregue os dados contidos no arquivo ex2data2.txt.

O arquivo contém 118 linhas e 3 colunas de dados. Cada coluna se refere a uma variável. Neste problema, deve-se desenvolver um modelo de classificação capaz de reproduzir as classes apresentadas na terceira coluna dos dados

Os dados apresentados referem-se a um problema de controle de qualidade de microchips em uma indústria. As colunas 1 e 2 correspondem aos scores obtido por um lote de microchips quando submetidos a dois testes. A coluna 3 indica se estes foram aprovados ou não.

Apresentar: Figura com os dados



Comentários: É possível desenvolver uma regressão logística para classificar corretamente os dados apresentados? Comente

=> Como podemos ver na figura, os dados azuis (aqueles que são classificados como 1) estão quase que encapsulados dentro dos

dados vermelhos. Portanto temos dois grupos 'bem' definidos, que podem ser separados por uma superfície de decisão.

Por isso, um modelo de regressão logística parece razoável para esse conjunto de dados.

- Com base nos dados, é possível verificar que 2 dimensões não são suficientes para classificar os dados. Tendo em vista esse problema, pode-se gerar mais atributos a partir da combinação dos atributos existentes. A função `mapFeature.m` irá mapear as características existentes em todas os termos polinomiais até o grau 30. O vetor abaixo apresenta o resultado até o grau 6.

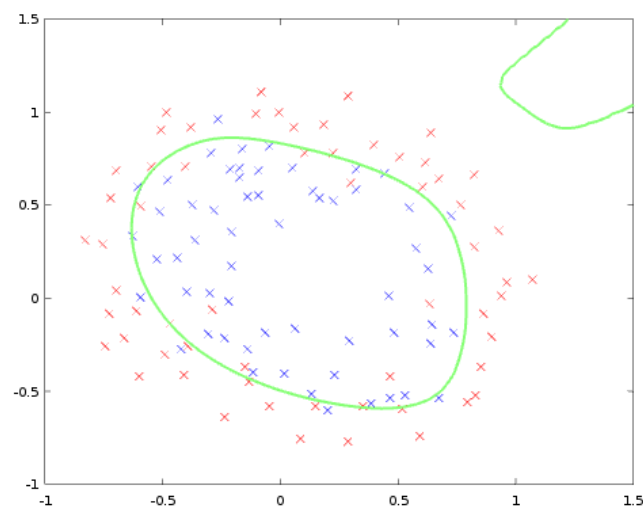
$$\text{mapFeature}(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_1x_2 \\ x_2^2 \\ x_1^3 \\ \vdots \\ x_1x_2^5 \\ x_2^6 \end{bmatrix}$$

Após utilizar a função `mapFeature` no dados, teremos agora um conjunto de 118 exemplos de 496 atributos além da variável que determina a classe.

- Implemente o algoritmo do gradiente descendente estocástico para encontrar os coeficientes da regressão.

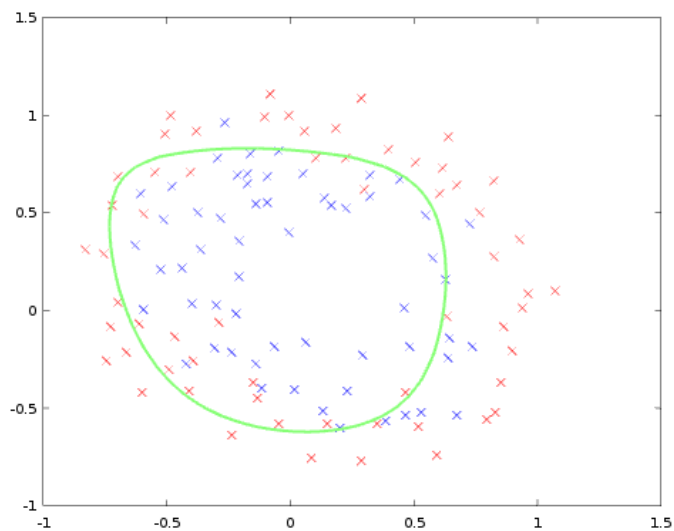
Para este algoritmo utilize $\alpha = 0.01$ e utilize 1000 épocas de treinamento. Desenvolva modelos com os seguintes valores de $\lambda = [0 \ 0.01 \ 0.25]$

Apresentar: Figuras apresentando os dados e as superfícies de decisão de cada modelo ($\lambda = [0 \ 0.01 \ 0.25]$). Para a geração da superfície de decisão, utilize a função `plotDecisionBoundary`.

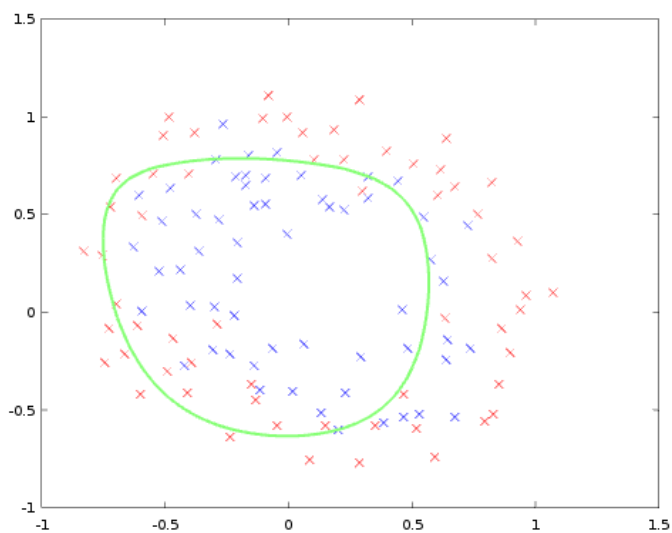


=>Para $\lambda = 0$:

Para $\lambda = 0.1$:



Para $\lambda = 0.25$:



Comentários: Analise os três gráficos e comente sobre o tema bias-variância.

=> Quanto mais aumentamos o valor do parâmetro λ , mais o modelo se sente “confortável” para usar as componentes de grau mais elevado, fazendo com que a superfície de decisão tome formas

mais curvas e detalhadas. Porém, se damos liberdade demais para o modelo, então entraremos em situações de overfitting (o modelo se adapta bem demais a aquele conjunto de dados, porém se entrarmos com outros dados, ele pode se tornar ruim). Portanto precisamos ter cuidado na escolha de λ . Altos valores de λ causam alta variância e valores baixos, alto bias. Note que, para esse nosso exemplo, se aumentarmos o valor de λ , a decision boundary muda.