Resolução de problemas em equipe 25

Estudantes: \_\_\_\_\_\_\_\_\_Gustavo Hammerschmidt\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A função nbclassifier recebe como parâmetros os dados observados (X, Y,) e o dado a ser classificado, (x), e um vetor com as classes (y). X é a matriz com 14 linhas e 4 colunas, onde cada linha corresponde a uma amostra das características, e as colunas os valores de cada uma das 4 características. Y é um vetor com 14 linhas e uma coluna. Cada linha é o valor da decisão (sim = 1, não = 2).

No exemplo, os valores de X e Y são obtidos da seguinte tabela de observações:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Idade (*X*1)** | **Renda (*X*2)** | **Estudante (*X*3)** | **Crédito (*X*4)** | **Compra computador (***Y***)** |
| < 30 (1) | Alta (3) | Não (2) | Bom (1) | Não (2) |
| < 30 (1) | Alta (3) | Não (2) | Bom (1) | Não (2) |
| 30 - 40 (2) | Alta (3) | Não (2) | Bom (1) | Sim (1) |
| > 40 (3) | Média (2) | Não (2) | Bom (1) | Sim (1) |
| > 40 (3) | Baixa (1) | Sim (1) | Bom (1) | Sim (1) |
| > 40 (3) | Baixa (1) | Sim (1) | Excelente (2) | Não (2) |
| 30 - 40 (2) | Baixa (1) | Sim (1) | Excelente (2) | Sim (1) |
| < 30 (1) | Média (2) | Não (2) | Bom (1) | Não (2) |
| < 30 (1) | Baixa (1) | Sim (1) | Bom (1) | Sim (1) |
| > 40 (3) | Média (2) | Sim (1) | Bom (1) | Sim (1) |
| < 30 (1) | Média (2) | Sim (1) | Excelente (2) | Sim (1) |
| 30 - 40 (2) | Média (2) | Não (2) | Excelente (2) | Sim (1) |
| 30 - 40 (2) | Alta (3) | Sim (1) | Bom (1) | Sim (1) |
| > 40 (3) | Média (2) | Não (2) | Excelente (2) | Não (2) |

As variáveis X e Y podem ser carregadas em memória usando o script xy.m. Basta digitar xy na linha de comando do MatLab.

>> xy

Você pode observar a matiz X digitando X na linha de comando

>> X

X =

1 3 2 1

1. 3 2 1
2. 3 2 1
3. 2 2 1

3 1 1 1

3 1 1 2

2 1 1 2

1 2 2 1

1 1 1 1

3 2 1 1

1. 2 1 2
2. 2 2 2
3. 3 1 1
4. 2 2 2

Você pode também observar o vetor Y da mesma maneira.

Para realizar a classificação conforme o exemplo 1.1, é preciso definir o vetor x para ser classificado. Na sequência vamos usar dois vetores diferentes para a classificação O primeiro vamos chamar de vetor x1, e colocá-lo na memória com o seguinte comando:

x1 = [1 2 1 1]

Observe no exemplo 1.1 que os valores dos elementos de x1 são aqueles escolhidos como os valores do vetor x (teórico), que no exemplo tem componentes x1 = 1, x2 = 2, x3 = 1, x3 = 1.

Use a função nbclassifier para obter o resultado do exemplo 1.1. Essa função recebe como argumento a matriz X, o vetor Y, que já foram carregados, o vetor x1, que você quer classificar. O último argumento é outro vetor (y) com os valores associados às classes (Sim = 1, Não = 2). A função retorna um vetor com os resultados dos cálculos da classificação, e a classe prevista. >> nbclassifier(X, Y, x1, [1,2]) % o vetor [1, 2] é o último argumento da função (y) ans =

0.028219 0.010286 ans = 1

>>

1. Repetir agora os mesmos procedimentos para o exemplo 1.2.

Colocar aqui o comando do MatLab e o resultado.

>> X2 = [3,2,2,1] ; % Os Valores Obtidos para esse vetor estão nos slides: TBL 25 - RA3

>> nbclassifier(X,Y,X2,[1,2])

ans =

0.021164 0.027429

ans = 2

1. Usar a função PXdadoY para encontrar os valores das probabilidades condicionais dos exemplos 1.1 e 1.2. Veja os argumentos da função nos comentários.

Exemplo 1.1:

>> PXdadoY(X,Y,[1,2,1,1], [1,2])

ans =

0.043896 0.028800

Exemplo 1.2:

>> PXdadoY(X,Y,[3,2,2,1], [1,2])

ans =

0.032922 0.076800

1. Use também a função PXdadoY1. Descreva aqui o resultado que ela fornece.

Exemplo 1.1:

>> PXdadoY1(X,Y,[1,2,1,1],[1,2])

ans =

0.22222 0.44444 0.66667 0.66667

0.60000 0.40000 0.20000 0.60000

Exemplo 1.2:

>> PXdadoY1(X,Y,[3,2,2,1],[1,2])

ans =

0.33333 0.44444 0.33333 0.66667

0.40000 0.40000 0.80000 0.60000

Fazer o upload desse arquivo no Blackboard.