Universidade Federal de Minas Gerais

**Exercício 4**

**Guilherme Vinícius Amorim**

**2017089081**

Exercício 1:

Inicialmente, deve-se amostrar duas distribuições normais no espaço R2, ou seja, duas distribuições com duas variáveis cada. As distribuição são caracterizadas como 𝒩(2*,* 2*, 𝜎*2) e 𝒩(4*,* 4*, 𝜎*2):

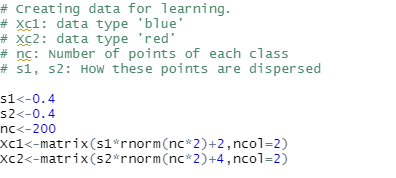


Figura 1: Rotina em R que implementa as duas classes com distribuições normais.

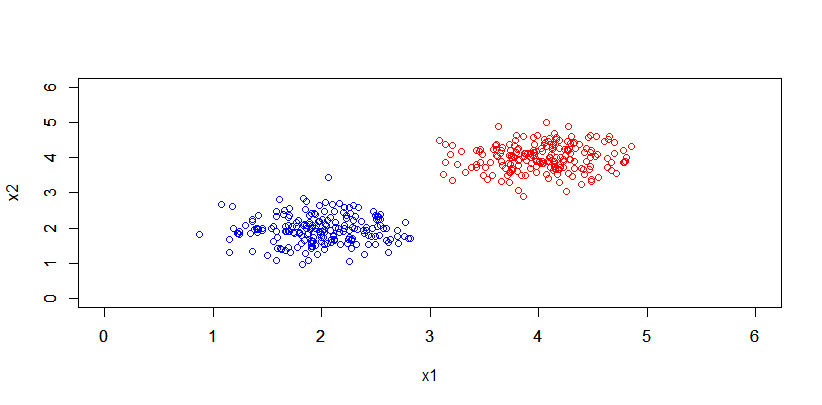


Figura 2: Classes um e dois dispersas no espaço R2.

Após amostrarmos as classes um e dois no espaço R2, a rotina do Perceptron foi utilizada para se encontrar o vetor **w** de parâmetros e, por conseguinte, a equação da reta que permite a classificação das classes. A rotina do Perceptron foi utilizada com os seguintes parâmetros:

Portanto, os parâmetros do vetor **w**, assim como os coeficientes da equação de reta que separa as duas classes são:

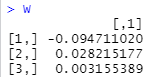


Figura 3: Vetor de parâmetros retornado pela rotina do Perceptron.

Segue abaixo o gráfico que mostra as classes um e dois no espaço R2, assim como a reta obtida pela rotina do Percepton:

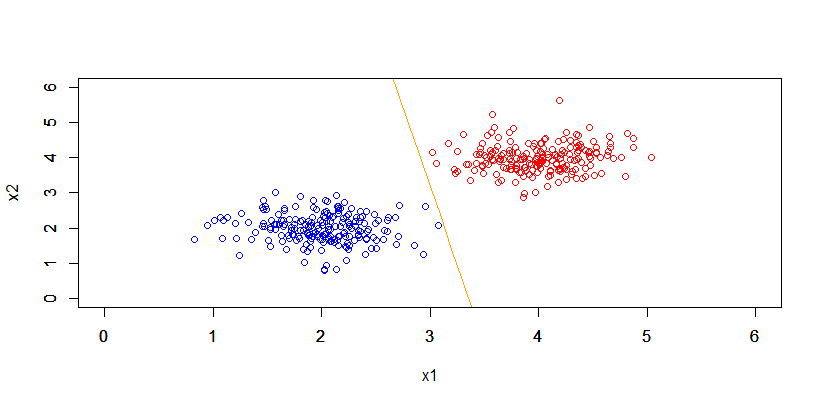


Figura 4: Classes um e dois e reta obtida pela rotina do Perceptron.

Além disso, para uma melhor ilustração de como a rotina do Perceptron consegue separar as classes um e dois, um *grid* foi criado de forma tal que, a partir de um gráfico 3D, separa-se possíveis novas entradas a partir da resposta linear obtida pelo Perceptron:

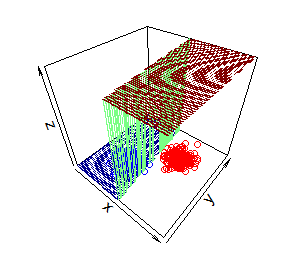


Figura 5: *Grid* com a separação entre classes um e dois.

Exercício 2:

Nesta segunda atividade o aluno deverá criar um conjunto de amostras de cada uma das duas distribuições do Exercício 1, ou seja, 200 amostras da classe 1 e 200 amostras da classe 2:

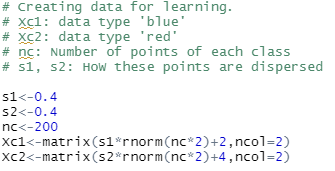


Figura 6: Rotina em R que gera 200 amostras da classe 1 e 200 amostras da classe 2.

O aluno deverá utilizar essas amostras para criar dois conjuntos **balanceados**, um chamado de conjunto de treinamento que será usado para achar os pesos *𝑤* e outro chamado de teste que servirá para avaliar a performance do seu separador dado pelos pesos encontrados no treinamento. O conjunto de treinamento irá conter 70% das amostras e o de teste 30%.

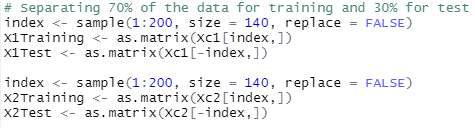


Figura 7: Rotina em R que separa 70% dos dados para o treinamento e 30% dos dados para teste.

Após a separação dos dois conjuntos o aluno usará o conjunto de treinamento para encontrar os pesos do Perceptron e utilizará o conjunto de teste para avaliar a performance do Perceptron simples. Apresente a acurácia e a matriz de confusão.

O conjunto de dados de treinamento, que possui 140 dados da classe um e 140 dados da classe dois, foi utilizado para o treinamento do Perceptron. Os seguintes parâmetros do Perceptron foram utilizados:

Assim, o vetor **w** obtido foi:

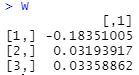


Figura 8: Vetor de parâmetros **w** obtido pela rotina de treinamento.

Utilizando, agora, o conjunto de dados de teste, que possui 60 dados da classe um e 60 dados da classe 2, a acurácia encontrada, o erro médio quadrático final, assim como o erro percentual foram:

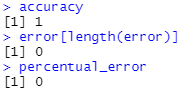


Figura 9: Acurácia, Erro quadrático médio e Erro Percentual obtidos na rotina de treinamento.

Segue abaixo um gráfico que acompanha a evolução do erro médio quadrático durante as épocas do treinamento:

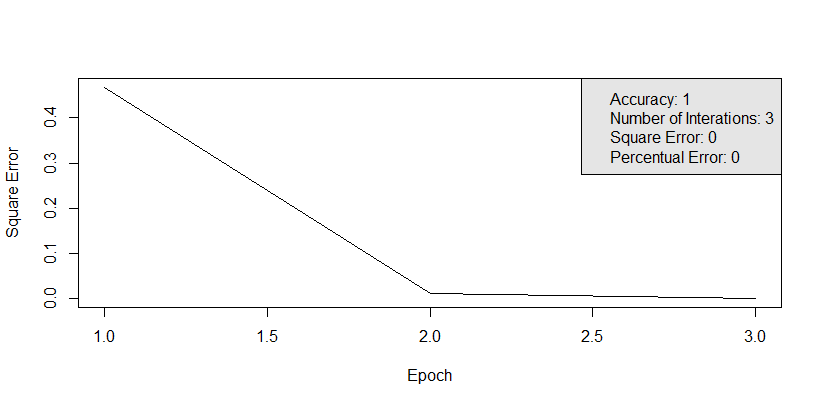


Figura 10: Gráfico que relaciona erro médio quadrático com a respectiva época durante o treinamento do Perceptron.

Exercício 3:

No Exercício 3 iremos trabalhar com uma base de dados conhecida como Iris (comando: *𝑑𝑎𝑡𝑎*(”*𝑖𝑟𝑖𝑠*”)). Essa base de dados possui 150 amostras e 4 características, sendo 50 para cada uma das três espécies de plantas que constitui a base. Nesta atividade o aluno irá realizar o treinamento do *Perceptron* para separar a espécie 1 (50 primeiras amostras) das outras duas espécies e avaliar o desempenho do mesmo. Com isso a espécie 1 será a Classe 1 e o conjunto das espécies 2 e 3 será a Classe 2. O aluno deverá então:

1. Importar as funções *yperceptron* e *trainperceptron* desenvolvida por ele em sala de aula.
2. Carregar os dados da Iris e armazená-los, sendo que a Classe 1 será composta das 50 primeiras amostras e a Classe 2 das 100 amostras posteriores as 50 primeiras, como descrito na introdução do problema.

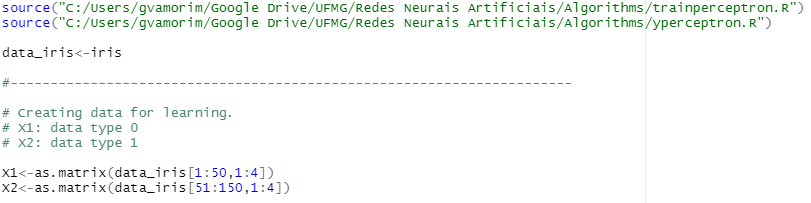


Figura 11: Importação das funções *yperceptron* e *trainperceptron* e separação entre as classes um e dois.

1. Rotular as amostras da Classe 1 com o valor de 0 e as amostras da Classe 2 com o valor 1.



Figura 12: Rotulando a Classe um com o valor de 0 e a Classe dois com o valor de 1

1. Selecionar aleatoriamente 70% das amostras para o conjunto de treinamento e 30% para o conjunto de teste, para cada uma das duas classes.

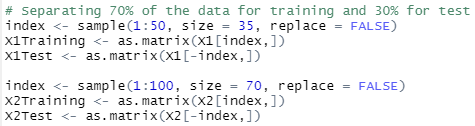


Figura 13: Separando 70% das amostras para treinamento e 30% das amostras para o conjunto de teste.

1. Utilizar as amostras de treinamento para fazer o treinamento do *Perceptron* utilizando a função *trainperceptron*.

A rotina do Perceptron foi feita com os dados de treinamento e com os seguintes parâmetros:

1. Extrair o vetor de pesos da função *trainperceptron*.

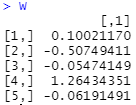


Figura 14: Vetor de pesos encontrado pela rotina de treinamento.

1. Concatenar as amostras de teste e seus respectivos *y* e dar entrada na função *yperceptron* (a função *yperceptron* não recebe o y), utilizando o vetor de peso extraído.

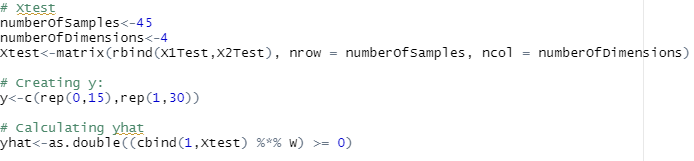


Figura 15: Calculando o dos dados separados para teste.

1. Calcular o erro percentual. (O erro é dado pelo número de amostras de teste classificadas de forma errada).
2. Imprimir a matriz de confusão.

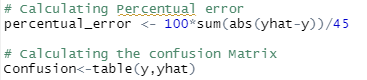


Figura 16: Calculando erro percentual e a matriz de Confusão.

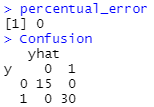


Figura 17: Erro percentual e a matriz de Confusão encontrados.

1. Crie um loop para repetir 100 vezes os itens 4-8, armazenando o valor do erro percentual do item 8. Plote o erro percentual em função do número de iteração e imprima o valor da variância do erro.

A partir da repetição dos itens 4-8 100 vezes em um loop e armazenando o erro percentual em um vetor, segue abaixo o gráfico relacionando erro percentual e o número de iteração, assim como o valor da variância do erro foi obtido:

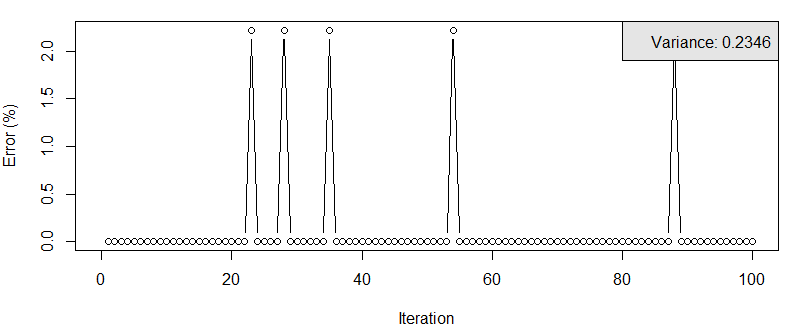


Figura 18: Gráfico relacionando o erro percentual por interação e o valor da variância desse erro.

.