

Exercício 4

A.P. Braga

Janeiro de 2021

TREINAMENTO PERCEPTRON SIMPLES

Esta atividade abordará o treinamento do perceptron simples.

EXERCÍCIO 1

Inicialmente, devem-se amostrar duas distribuições Normais no espaço R^2 , ou seja, duas distribuições com duas variáveis cada (Ex: x_1 e x_2). As distribuições são caracterizadas como $\mathcal{N}(2, 2, \sigma^2)$ e $\mathcal{N}(4, 4, \sigma^2)$, como pode ser visualizado na Fig. 1.

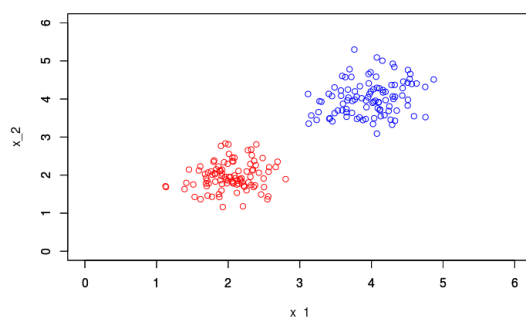


Figure 1: Dados amostrados de duas distribuições Normais com médias $m1 = (2; 2)^T$ e $m2 = (4; 4)^T$ e coeficiente de correlação nulo

Nesta atividade o aluno irá fazer o treinamento do perceptron com o objetivo de encontrar o vetor de pesos w e encontrar a de separação como mostra a Figura 3.

```
rm(list=ls())
library('plot3D')
source('trainperceptron.R')
source('yperceptron.R')
s1<-0.4
s2<-0.4|
nc<-200
xc1<-matrix(rnorm(nc*2),ncol=2)*s1 + t(matrix((c(2,2)),ncol=nc,nrow=2))
xc2<-matrix(rnorm(nc*2),ncol=2)*s2 + t(matrix((c(4,4)),ncol=nc,nrow=2))
plot(xc1[,1],xc1[,2],col = 'red', xlim = c(0,6),ylim = c(0,6),xlab = 'x_1',ylab='x_2')
par(new=T)
plot(xc2[,1],xc2[,2],col = 'blue', xlim = c(0,6),ylim = c(0,6),xlab = '',ylab='')

x1_reta<-seq(6/100,6,6/100)
x2_reta<- -x1_reta+6
par(new=T)
plot(x1_reta, x2_reta , type = 'l',col = 'orange', xlim = c(0,6),ylim = c(0,6),xlab = '',ylab='')
```

Aqui você deve chamar a função trainperceptron que irá retornar o vetor de pesos w

```
seqi<-seq(0,6,0.1)
seqj<-seq(0,6,0.1)
M <- matrix(0,nrow=length(seqi),ncol=length(seqj))

ci<-0
for (i in seqi){
  ci<-ci+1
  cj<-0
  for(j in seqj)
  {
    cj<-cj+1
    x<-c(i,j)
    M[ci,cj]<- yperceptron(x,w,1)
  }
}

plot(xc1[,1],xc1[,2],col = 'red', xlim = c(0,6),ylim = c(0,6),xlab = 'x_1',ylab='x_2')
par(new=T)
plot(xc2[,1],xc2[,2],col = 'blue', xlim = c(0,6),ylim = c(0,6),xlab = '',ylab='')
par(new=T)
contour(seqi,seqj,M, xlim = c(0,6),ylim = c(0,6),xlab = '',ylab='')

persp3D(seqi,seqj,M,counter=T,theta = 55, phi = 30, r = 40, d = 0.1, expand = 0.5,
  ltheta = 90, lphi = 180, shade = 0.4, ticktype = "detailed", nticks=5)
```

Figure 2: Código em R para plotar a superfície de resposta do Perceptron

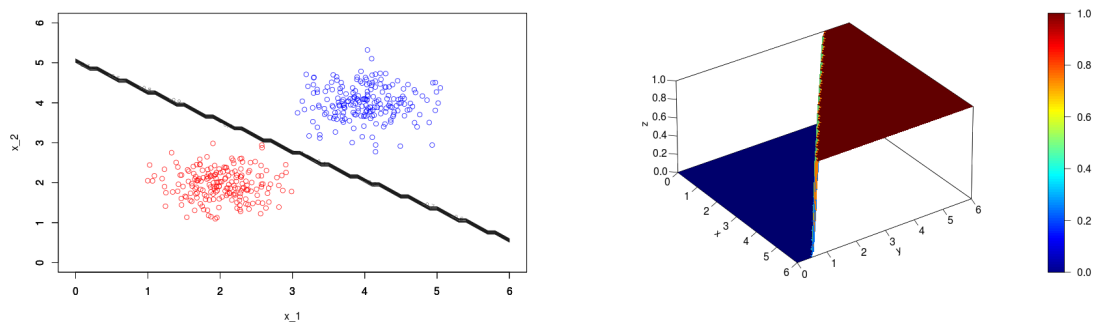


Figure 3: Resolução Ex1

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 divididas em conjunto de treinamento e teste. O conjunto de treinamento irá conter 70% das amostras e o de teste 30%, amostradas de forma aleatória do conjunto total de 400 amostras.

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 o seu desempenho. Apresente a acurácia e a matriz de confusão.

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

4. Selecionar aleatoriamente 70% das amostras para o conjunto de treinamento e 30% para o conjunto de teste, para cada uma das duas classes.
5. Utilizar as amostras de treinamento para fazer o treinamento do *Perceptron* utilizando a função *trainperceptron*.
6. Extrair o vetor de pesos da função *trainperceptron*.
7. Concatenar as amostras de teste e seus respectivos rótulos y e dar entrada na função *yperceptron* (a função *yperceptron* não recebe o y), utilizando o vetor de peso extraído.
8. Calcular o erro percentual. (O erro é dado pelo número de amostras de teste classificadas de forma errada)
9. Imprimir a matriz de confusão
10. 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 (como na Figura) e imprima o valor da variância do erro.

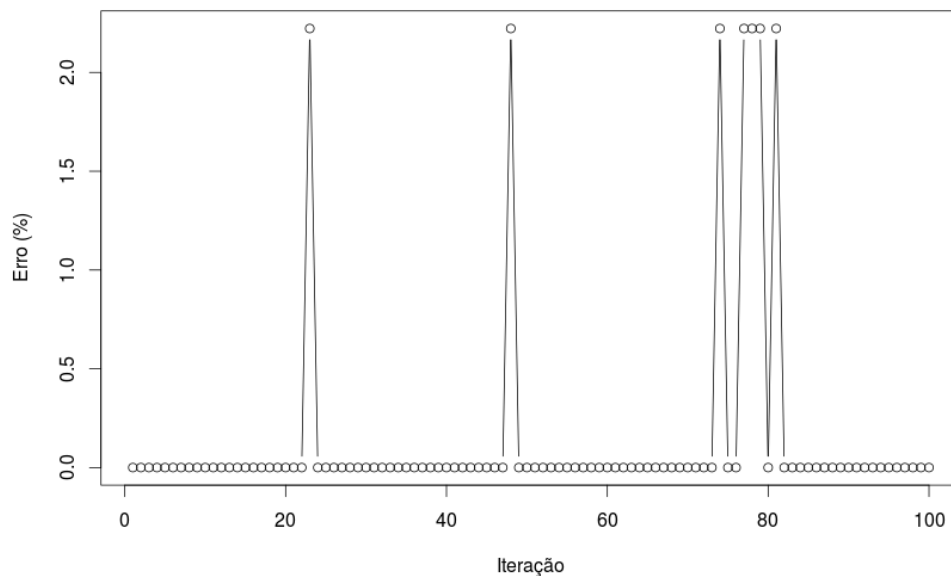


Figure 4: Exemplo de gráfico para o Exercício 3, ítem 10

EXERCÍCIO 4

Deve ser realizado o treinamento de uma rede Perceptron para a base Breast Cancer (disponível no pacote MLbench e no repositório UCI). O procedimento deve ser composto de:

1. Carregar a base e preprocessar (por exemplo, remover dados faltantes e normalizar os dados, caso necessário)
2. Separar a base de dados entre treinamento e teste
3. Com os dados de treinamento, treinar um modelo Perceptron, conforme consta nas notas de aula
4. Realizar a previsão para os dados de teste
5. Comparar o valor obtido pelo perceptron com os valores reais de teste (calcular a acurácia)
6. repetir 30 vezes as etapas a partir do item 2 e calcular média e desvio padrão dos valores de acurácia