

Universidade Federal de Minas Gerais

Nome: Guilherme Vinícius Amorim

Matrícula: 2017089081

Data: 10/2020

Exercício 8

O objetivo do exercício desta semana é combinar os conceitos aprendidos na Unidade 2 e construir uma rede neural que soma elementos das redes RBF e das redes ELM. As bases de dados a serem estudadas são as mesmas do exercício 6:

- *BreastCancer (diagnostic)*
- *Statlog (Heart)*

Os mesmos cuidados para separação de conjunto de treinamento e teste, já mencionados no enunciado do exercício 6, devem ser tomados, bem como deve ser dada atenção ao escalonamento dos dados (entre [0; 1] ou [-1; 1]). Segue abaixo as rotinas que carregam essas bases de dados e fazem esse ajuste nos valores de entradas:

```
### Data

data("BreastCancer")
BreastCancer$Class<-1*(BreastCancer$Class=='benign')
for(i in 1:length(BreastCancer$Class)){
  if(BreastCancer[[11]][i] == 0){
    BreastCancer[[11]][i] <- -1
  }
}
# Removing NA data
BreastCancer<-BreastCancer[complete.cases(BreastCancer),]
dt<-data.matrix(BreastCancer, rownames.force = NA)

X<-as.matrix(dt[1:dim(dt)[1], 2:10])
Y<-as.matrix(dt[1:dim(dt)[1], 11])

for(i in 1:dim(X)[2]){
  max<-max(X[,i])
  min<-min(X[,i])
  for(j in 1:dim(X)[1]){
    X[j,i]<-(X[j,i]-min)/(max-min)
  }
}

#-----

XY <- splitForTrainingAndTest(X, Y, ratio = 0.3)
xtrain <- XY$inputsTrain
ytrain <- XY$targetsTrain
xtest <- XY$inputsTest
ytest <- XY$targetsTest
```

Figura 1: Rotina para carregamento e ajuste dos dados da base de dados *BreastCancer*.

```

### Data

data(heart)

# Removing NA data
heart<-heart[complete.cases(heart),]
dt<-data.matrix(heart, rownames.force = NA)

X<-as.matrix(dt[1:dim(dt)[1], 1:12])
Y<-as.matrix(dt[1:dim(dt)[1], 13])

for(i in 1:length(Y)){
  if(Y[i] == 2){
    Y[i] <- -1
  }
}

#-----

XY <- splitForTrainingAndTest(X, Y, ratio = 0.3)
xtrain <- as.matrix(XY$inputsTrain)
ytrain <- as.matrix(XY$targetsTrain)
xtest <- as.matrix(XY$inputsTest)
ytest <- as.matrix(XY$targetsTest)

for(i in 1:dim(X)[2]){
  max<-max(X[,i])
  min<-min(X[,i])
  for(j in 1:dim(X)[1]){
    X[j,i]<-(X[j,i]-min)/(max-min)
  }
}

```

Figura 2: Rotina para carregamento e ajuste dos dados da base de dados *Heart*.

Com os dados carregados e ajustados, os algoritmos das redes ELM e RBF foram combinados de forma que uma rede RBF foi construída com centros e raios atribuídos de forma aleatória aos neurônios. Segue abaixo a alteração feita na rotina aprendida em sala de aula, permitindo, agora, uma distribuição aleatória dos centros:

```

xcenters <- matrix(0, nrow = p, ncol = n)
for (i in seq(1, p, 1)) {
  xcenters[i,] <- xin[runif(1, 1, N),]
}
xcluster <- matrix(0, ncol = 1, nrow = N)
for (i in seq(1, N, 1)) {
  xcluster[i,] <- sample(1:p, size = 1)
}

# Armazena vetores de centros das funções
m <- xcenters

```

Figura 3: Distribuição aleatória dos centros na rede RBF.

A partir da rotina para treinamento ter sido modificada, o algoritmo foi utilizado para números de centros entre 1 e 20, tanto para a base *heart* quanto para a base

BreastCancer. Segue abaixo as rotinas utilizadas para o treinamento da rede, assim como os resultados obtidos:

```
# RBF network with randomly assigned centers to neurons
print("Error using randomly assigned centers ")
for (p in 1:20){
  modRBF <- treinaRBF(xtrain, ytrain, p)
  yhat <- YRBF(xtest, modRBF)
  yhat <- sign(yhat)

  # Error calc
  # Calc Error of Test data
  error<-sum((ytest-yhat)^2)/(4*nrow(yhat))
  vr <- var(ytest-yhat)
  cat("Error : ", error,"+/-", vr, "using ", p, " neurons", "\n")
}
```

Figura 4: Rotina de treinamento de ambas as bases de dados.

```
Error : 0.09756098 +/- 0.3538977 using 1 neurons
Error : 0.1317073 +/- 0.4596844 using 2 neurons
Error : 0.1219512 +/- 0.4304161 using 3 neurons
Error : 0.102439 +/- 0.3695839 using 4 neurons
Error : 0.1414634 +/- 0.4881875 using 5 neurons
Error : 0.1317073 +/- 0.4596844 using 6 neurons
Error : 0.1219512 +/- 0.4304161 using 7 neurons
Error : 0.1512195 +/- 0.5274032 using 8 neurons
Error : 0.1219512 +/- 0.4304161 using 9 neurons
Error : 0.1170732 +/- 0.415495 using 10 neurons
Error : 0.1170732 +/- 0.415495 using 11 neurons
Error : 0.1121951 +/- 0.4003826 using 12 neurons
Error : 0.08780488 +/- 0.3219512 using 13 neurons
Error : 0.1317073 +/- 0.4596844 using 14 neurons
Error : 0.07804878 +/- 0.2892396 using 15 neurons
Error : 0.06341463 +/- 0.2387374 using 16 neurons
Error : 0.09268293 +/- 0.3380201 using 17 neurons
Error : 0.1317073 +/- 0.4596844 using 18 neurons
Error : 0.1560976 +/- 0.5295074 using 19 neurons
Error : 0.1170732 +/- 0.415495 using 20 neurons
```

Figura 5: Erro médio quadrático +/- variância para a base de dados *BreastCancer*.

```
Error : 0.4320988 +/- 0.9938272 using 1 neurons
Error : 0.4320988 +/- 0.9938272 using 2 neurons
Error : 0.4320988 +/- 0.9938272 using 3 neurons
Error : 0.4320988 +/- 0.9938272 using 4 neurons
Error : 0.4320988 +/- 0.9938272 using 5 neurons
Error : 0.4320988 +/- 0.9938272 using 6 neurons
Error : 0.4320988 +/- 0.9938272 using 7 neurons
Error : 0.4320988 +/- 0.9938272 using 8 neurons
Error : 0.4320988 +/- 0.9938272 using 9 neurons
Error : 0.4320988 +/- 0.9938272 using 10 neurons
Error : 0.4320988 +/- 0.9938272 using 11 neurons
Error : 0.4320988 +/- 0.9938272 using 12 neurons
Error : 0.4320988 +/- 0.9938272 using 13 neurons
Error : 0.4320988 +/- 0.9938272 using 14 neurons
Error : 0.4320988 +/- 0.9938272 using 15 neurons
Error : 0.4320988 +/- 0.9938272 using 16 neurons
Error : 0.4320988 +/- 0.9938272 using 17 neurons
Error : 0.4320988 +/- 0.9938272 using 18 neurons
```

Figura 6: Erro médio quadrático +/- variância para a base de dados *Heart*.

Comparando os resultados atuais alcançados para a base *BreastCancer* com os resultados dos algoritmos de Perceptron e ELM feitos no exercício 6, tem-se um resultado não suficiente. Mas também vale ressaltar que para a base ELM do exercício 6 um número maior de neurônios foi utilizado:

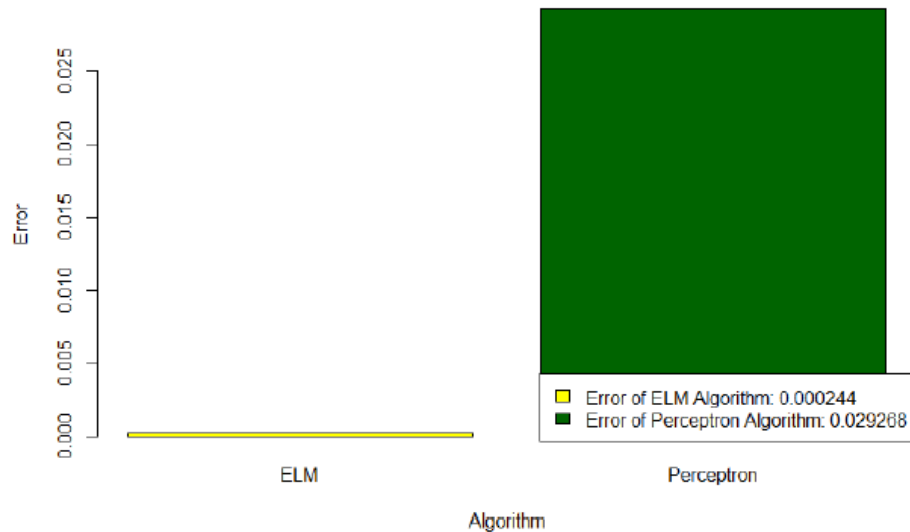


Figura 7: Resultados alcançados para a base de dados *BreastCancer* no exercício 6.

Comparando os resultados atuais para a base *Heart* com os resultados alcançados no exercício 6, observa-se também que os resultados foram não satisfatórios:

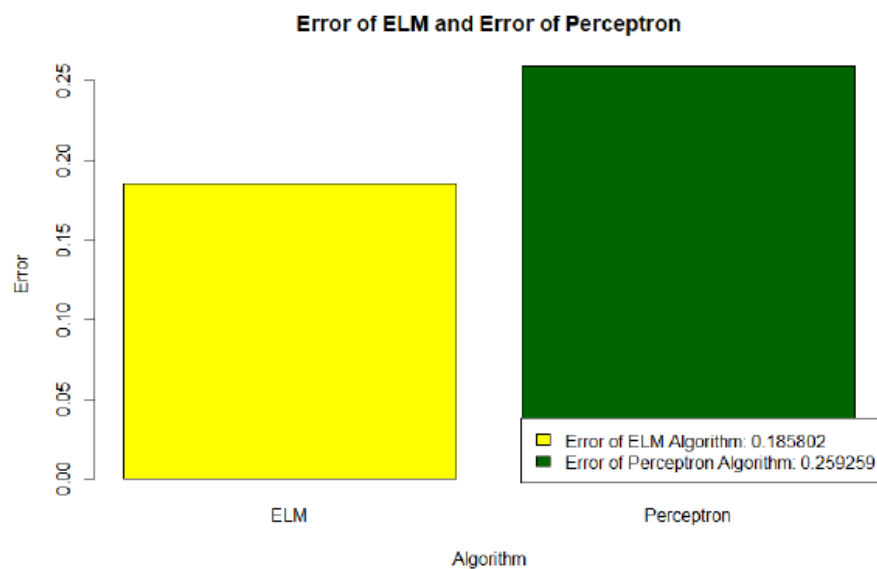


Figura 8: Resultados alcançados para a base de dados *Heart* no exercício 6.

Ademais, treinou-se também ambas as bases de dados com uma rede RBF, mas agora com os centros e raios ajustados com o k-medias.

```
xclust<-kmeans(xin,p)

# Armazena vetores de centros das funções
m<-as.matrix(xclust$centers)
```

Figura 9: Escolha dos centros da rede RBF com o k-médias.

Segue abaixo os resultados alcançados para ambas as bases de dados:

```
Error : 0.09756098 +/- 0.3538977 using 1 neurons
Error : 0.1317073 +/- 0.4596844 using 2 neurons
Error : 0.1219512 +/- 0.4304161 using 3 neurons
Error : 0.1073171 +/- 0.3850789 using 4 neurons
Error : 0.1121951 +/- 0.4003826 using 5 neurons
Error : 0.1219512 +/- 0.4304161 using 6 neurons
Error : 0.1268293 +/- 0.4451459 using 7 neurons
Error : 0.102439 +/- 0.3695839 using 8 neurons
Error : 0.1219512 +/- 0.4304161 using 9 neurons
Error : 0.1121951 +/- 0.4003826 using 10 neurons
Error : 0.09756098 +/- 0.3538977 using 11 neurons
Error : 0.1219512 +/- 0.4304161 using 12 neurons
Error : 0.1463415 +/- 0.5021521 using 13 neurons
Error : 0.09756098 +/- 0.3538977 using 14 neurons
Error : 0.1317073 +/- 0.4596844 using 15 neurons
Error : 0.1512195 +/- 0.5159254 using 16 neurons
Error : 0.1219512 +/- 0.4304161 using 17 neurons
Error : 0.1463415 +/- 0.5021521 using 18 neurons
Error : 0.1121951 +/- 0.4003826 using 19 neurons
Error : 0.1902439 +/- 0.6192253 using 20 neurons
```

Figura 10: Erro médio quadrático +/- variância para a base de dados *BreastCancer*.

```
Error : 0.4320988 +/- 0.9938272 using 1 neurons
Error : 0.4320988 +/- 0.9938272 using 2 neurons
Error : 0.4320988 +/- 0.9938272 using 3 neurons
Error : 0.4320988 +/- 0.9938272 using 4 neurons
Error : 0.4320988 +/- 0.9938272 using 5 neurons
Error : 0.4320988 +/- 0.9938272 using 6 neurons
Error : 0.4320988 +/- 0.9938272 using 7 neurons
Error : 0.4320988 +/- 0.9938272 using 8 neurons
Error : 0.4320988 +/- 0.9938272 using 9 neurons
Error : 0.4320988 +/- 0.9938272 using 10 neurons
Error : 0.4320988 +/- 0.9938272 using 11 neurons
Error : 0.4320988 +/- 0.9938272 using 12 neurons
Error : 0.4320988 +/- 0.9938272 using 13 neurons
Error : 0.4320988 +/- 0.9938272 using 14 neurons
Error : 0.4320988 +/- 0.9938272 using 15 neurons
Error : 0.4320988 +/- 0.9938272 using 16 neurons
Error : 0.4320988 +/- 0.9938272 using 17 neurons
Error : 0.4320988 +/- 0.9938272 using 18 neurons
Error : 0.5679012 +/- 0.9938272 using 19 neurons
```

Figura 11: Erro médio quadrático +/- variância para a base de dados *Heart*.

Observa-se, portanto, que os resultados obtidos utilizando o k-médias como maneira de se escolher os centros e os raios da rede RBF se mostram também inferiores aos resultados obtidos no exercício 6 para as redes Perceptron e ELM.

Contudo, vale também ressaltar, que a diferença de resultado entre as formas de se escolher os centros das redes RBFs não interferem consideravelmente nos resultados do treinamento. Inclusive, para alguns casos, os resultados para centros gerados aleatoriamente foram melhores que os resultados de centros gerados pelo k-médias.