Resumo da atividade da disciplina exploração e mineração de dados (Técnicas)

Prof. Dr. Baldoino Fonseca.

Mestrando: Randerson Douglas R. Santos

Universidade Federal de Alagoas (UFAL)
Instituto de Computação (IC)

rdrs@ic.ufal.br

1. Regressão Linear

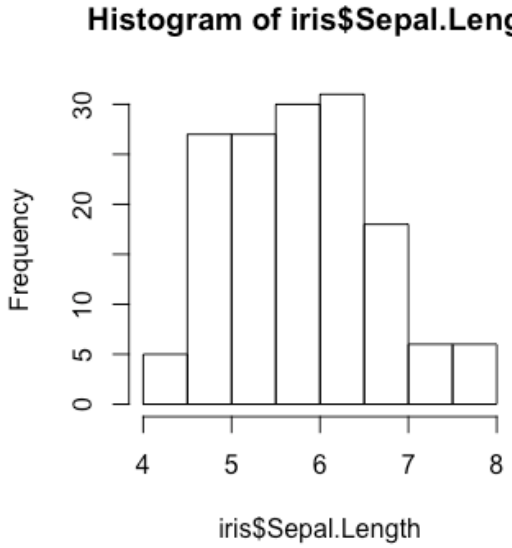
- A base utilizada para a aplicação desta técnica foi a IRIS baixada da plataforma kaggle.

- Código gerado:
  data("iris")
  str(iris)

  'data.frame': 150 obs. of 5 variables:
  $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
  $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
  $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
  $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
  $ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

  summary(iris$Sepal.Length)
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  4.300000 5.100000 5.800000 5.843333 6.400000 7.900000

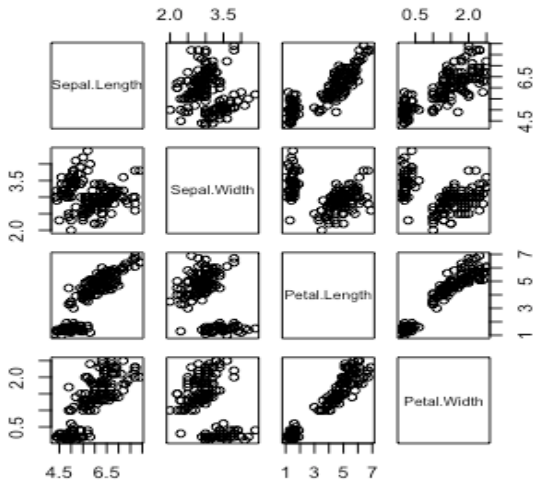
  hist(iris$Sepal.Length)




```
cor(iris[c("Sepal.Length", "Sepal.Width", "Petal.Length",
"Petal.Width")])
```

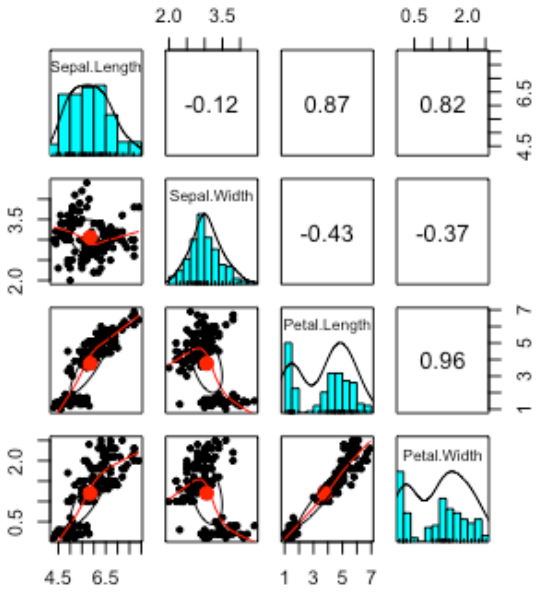
```
              Sepal.Length  Sepal.Width  Petal.Length  Petal.Width
Sepal.Length  1.0000000000 -0.1175697841  0.8717537759  0.8179411263
Sepal.Width  -0.1175697841  1.0000000000 -0.4284401043 -0.3661259325
Petal.Length  0.8717537759 -0.4284401043  1.0000000000  0.9628654314
Petal.Width   0.8179411263 -0.3661259325  0.9628654314  1.0000000000
```

```
pairs(iris[c("Sepal.Length", "Sepal.Width", "Petal.Length",
"Petal.Width")])
```




```
library(psych)
library(stats)
```

```
pairs.panels(iris[c("Sepal.Length", "Sepal.Width", "Petal.Length",
"Petal.Width")])
```




```
ins_model <- lm(Sepal.Length ~ ., data = iris)
ins_model
```

```
Call:
lm(formula = Sepal.Length ~ ., data = iris)

Coefficients:
       (Intercept)        Sepal.Width       Petal.Length        Petal.Width  
         2.1712663           0.4958889           0.8292439          -0.3151552  
Speciesversicolor   Speciesvirginica  
        -0.7235620          -1.0234978  
```

```
summary(ins_model)
```

```
Call:
lm(formula = Sepal.Length ~ ., data = iris)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.79423 -0.21874  0.00899  0.20255  0.73104 

Coefficients:
```

```
                   Estimate   Std. Error      t value             Pr(>|t|)
(Intercept)        2.17126629  0.27979415  7.76023    0.0000000000014295 ***
Sepal.Width        0.49588894  0.08606992  5.76147     0.0000000486751587
***
Petal.Length       0.82924391  0.06852765 12.10087  < 0.000000000000000222
***
Petal.Width       -0.31515517  0.15119575 -2.08442              0.0388883 *
Speciesversicolor -0.72356196  0.24016894 -3.01272              0.0030596 **
Speciesvirginica  -1.02349781  0.33372630 -3.06688              0.0025843 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3068261 on 144 degrees of freedom
Multiple R-squared:  0.8673123,       Adjusted R-squared:  0.862705
F-statistic:  188.251 on 5 and 144 DF,  p-value: < 0.00000000000000022204
```

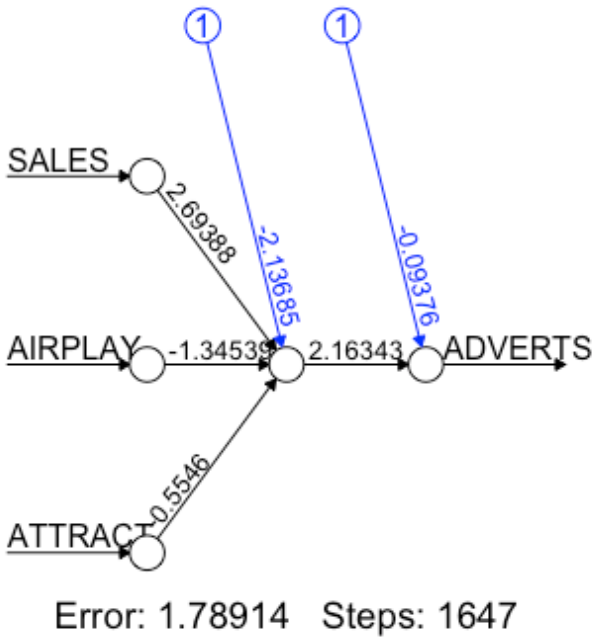
2. Redes Neurais
   - A base utilizada para a aplicação desta técnica foi a Record2 encontra-se localmente, trata de uma base referente a uma produtora musical.

   - Código gerado:

```
library(haven)
Record2 <- read_sav("Desktop/Mestrado PPGMCC 2017:2/2
Estatística/Chapters_01_08/Chapter 05/Record2.sav")
View(Record2)

normalize <- function(x) {
  return((x - min(x)) / (max(x) - min(x)))
}
record_norm <- as.data.frame(lapply(Record2, normalize))
summary(record_norm$SALES)
```

```
      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
0.0000000 0.3642857 0.5428571  0.5234286  0.6857143 1.0000000
```

```
record_train <- record_norm[1:150, ]
record_test <- record_norm[151:200, ]

library(neuralnet)

record_model <- neuralnet(ADVERTS ~ SALES + AIRPLAY
                          + ATTRACT,
                          data = record_train)
plot(record_model)
```



```
record_test
model_results <- compute(record_model, record_test[1:3])
predicted_adverts <- model_results$net.result
cor(predicted_adverts, record_test$ADVERTS)
```

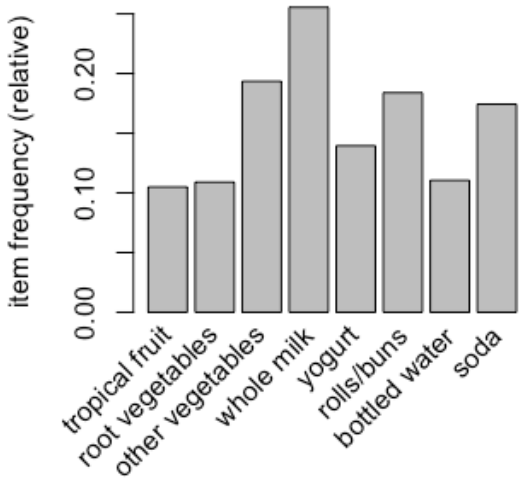
```
[1,] 0.8463515737
```

O que indica um forte relação, pois o valor está muito próximo de 1.

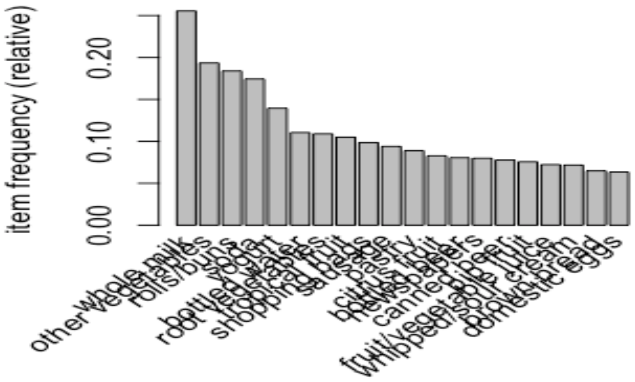
3. Regras de Associação
    - Quanto ao método regra de associação fiquei confuso quanto a escolher um dataset e fiz utilizado mesmo do livro que é o dataset groceryrules que já vem no pacote arules.

    - Código gerado:

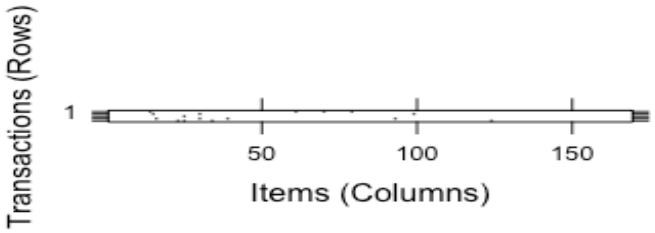
```
install.packages("arules")
library(arules)
data("Groceries")
Groceries
summary(Groceries)
View(Groceries)

inspect(Groceries[1:5])
itemFrequency(Groceries[, 1:3])

itemFrequencyPlot(Groceries, support = 0.1)
```



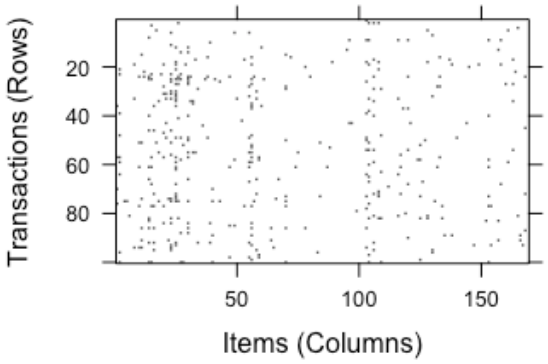
itemFrequencyPlot(Groceries, topN = 20)



image(Groceries[1:5])



```
image(sample(Groceries, 100))
```




```
groceryrules <- apriori(Groceries, parameter = list(support =
                                        0.006, confidence = 0.25,
minlen = 2))
groceryrules

summary(groceryrules)
```

```
set of 463 rules

rule length distribution (lhs + rhs):sizes
  2   3   4
150 297  16

    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
2.000000 2.000000 3.000000 2.710583 3.000000 4.000000

summary of quality measures:
    support              confidence            lift              count
 Min.   :0.006100661   Min.   :0.2500000   Min.   :0.9932367   Min.   : 60.0000
 1st Qu.:0.007117438   1st Qu.:0.2970711   1st Qu.:1.6229230   1st Qu.: 70.0000
 Median :0.008744281   Median :0.3553719   Median :1.9332351   Median : 86.0000
 Mean   :0.011539429   Mean   :0.3785573   Mean   :2.0350922   Mean   :113.4903
 3rd Qu.:0.012302999   3rd Qu.:0.4494849   3rd Qu.:2.3564791   3rd Qu.:121.0000
 Max.   :0.074834774   Max.   :0.6600000   Max.   :3.9564774   Max.   :736.0000

mining info:
      data ntransactions support confidence
 Groceries          9835   0.006       0.25
```

```
inspect(groceryrules[1:3])

inspect(sort(groceryrules, by = "lift")[1:5])
berryrules <- subset(groceryrules, items %in% "berries")
inspect(berryrules)

write(groceryrules, file = "groceryrules.csv",
      sep = ",", quote = TRUE, row.names = FALSE)
```

```
groceryrules_df <- as(groceryrules, "data.frame")
str(groceryrules_df)

data.frame':     463 obs. of  5 variables:
$ rules      : Factor w/ 463 levels "{baking powder} => {other
vegetables}  ",..: 237 204 128 127 129 238 317 21 89 90 ...
 $ support   : num  0.00691 0.0061 0.00702 0.00773 0.00773 ...
 $ confidence: num  0.4 0.405 0.431 0.475 0.475 ...
 $ lift      : num  1.57 1.59 3.96 2.45 1.86 ...
 $ count     : num  68 60 69 76 76 69 70 67 63 88 ...
```