Feature Selection

Kennedy Muriuki

22/09/2020

Feature Selection in R

```r
# loading the dataset
data<- read.csv(file.choose(), header = T, stringsAsFactors = T)
head(data)
```

```
##    Invoice.ID Branch Customer.type Gender           Product.line Unit.price
## 1 750-67-8428      A        Member Female      Health and beauty      74.69
## 2 226-31-3081      C        Normal Female Electronic accessories      15.28
## 3 631-41-3108      A        Normal   Male     Home and lifestyle      46.33
## 4 123-19-1176      A        Member   Male      Health and beauty      58.22
## 5 373-73-7910      A        Normal   Male      Sports and travel      86.31
## 6 699-14-3026      C        Normal   Male Electronic accessories      85.39
##   Quantity     Tax      Date  Time     Payment   cogs gross.margin.percentage
## 1        7 26.1415  1/5/2019 13:08     Ewallet 522.83                4.761905
## 2        5  3.8200  3/8/2019 10:29        Cash  76.40                4.761905
## 3        7 16.2155  3/3/2019 13:23 Credit card 324.31                4.761905
## 4        8 23.2880 1/27/2019 20:33     Ewallet 465.76                4.761905
## 5        7 30.2085  2/8/2019 10:37     Ewallet 604.17                4.761905
## 6        7 29.8865 3/25/2019 18:30     Ewallet 597.73                4.761905
##   gross.income Rating    Total
## 1      26.1415    9.1 548.9715
## 2       3.8200    9.6  80.2200
## 3      16.2155    7.4 340.5255
## 4      23.2880    8.4 489.0480
## 5      30.2085    5.3 634.3785
## 6      29.8865    4.1 627.6165
```

```r
# checking the dataset
str(data)
```

```
## 'data.frame':    1000 obs. of  16 variables:
##  $ Invoice.ID             : Factor w/ 1000 levels "101-17-6199",..: 815 143 654 19 340 734 316 265 7
##  $ Branch                 : Factor w/ 3 levels "A","B","C": 1 3 1 1 1 3 1 3 1 2 ...
##  $ Customer.type          : Factor w/ 2 levels "Member","Normal": 1 2 2 1 2 2 1 2 1 1 ...
##  $ Gender                 : Factor w/ 2 levels "Female","Male": 1 1 2 2 2 2 1 1 1 1 ...
##  $ Product.line           : Factor w/ 6 levels "Electronic accessories",..: 4 1 5 4 6 1 1 5 4 3 ...
##  $ Unit.price             : num  74.7 15.3 46.3 58.2 86.3 ...
##  $ Quantity               : int  7 5 7 8 7 7 6 10 2 3 ...
```

```
##  $ Tax                    : num  26.14 3.82 16.22 23.29 30.21 ...
##  $ Date                   : Factor w/ 89 levels "1/1/2019","1/10/2019",..: 27 88 82 20 58 77 49 48 2
##  $ Time                   : Factor w/ 506 levels "10:00","10:01",..: 147 24 156 486 30 394 215 78 342
##  $ Payment                : Factor w/ 3 levels "Cash","Credit card",..: 3 1 2 3 3 3 3 3 2 2 ...
##  $ cogs                   : num  522.8 76.4 324.3 465.8 604.2 ...
##  $ gross.margin.percentage: num  4.76 4.76 4.76 4.76 4.76 ...
##  $ gross.income           : num  26.14 3.82 16.22 23.29 30.21 ...
##  $ Rating                 : num  9.1 9.6 7.4 8.4 5.3 4.1 5.8 8 7.2 5.9 ...
##  $ Total                  : num  549 80.2 340.5 489 634.4 ...
```

```r
# changing the date and time column into date and time
Dates <- format(as.POSIXct(strptime(data$Date, "%d/%m/%y"), format="%d/%m/%y"))
Times <- format(as.POSIXct(strptime(data$Time, "%H:%M"), format="%H:%M"))
head(Times)
```

```
## [1] "2020-09-22 13:08:00" "2020-09-22 10:29:00" "2020-09-22 13:23:00"
## [4] "2020-09-22 20:33:00" "2020-09-22 10:37:00" "2020-09-22 18:30:00"
```

```r
# separating the classes
library(tidyr)

data <- separate(data, "Date", c("Day","Month","Year"))

data <- separate(data, "Time", c("Hour","Minute"))
head(data)
```

```
##    Invoice.ID Branch Customer.type Gender           Product.line Unit.price
## 1 750-67-8428      A        Member Female      Health and beauty      74.69
## 2 226-31-3081      C        Normal Female Electronic accessories      15.28
## 3 631-41-3108      A        Normal   Male     Home and lifestyle      46.33
## 4 123-19-1176      A        Member   Male      Health and beauty      58.22
## 5 373-73-7910      A        Normal   Male      Sports and travel      86.31
## 6 699-14-3026      C        Normal   Male Electronic accessories      85.39
##   Quantity     Tax Day Month Year Hour Minute     Payment   cogs
## 1        7 26.1415   1     5 2019   13     08     Ewallet 522.83
## 2        5  3.8200   3     8 2019   10     29        Cash  76.40
## 3        7 16.2155   3     3 2019   13     23 Credit card 324.31
## 4        8 23.2880   1    27 2019   20     33     Ewallet 465.76
## 5        7 30.2085   2     8 2019   10     37     Ewallet 604.17
## 6        7 29.8865   3    25 2019   18     30     Ewallet 597.73
##   gross.margin.percentage gross.income Rating    Total
## 1                4.761905      26.1415    9.1 548.9715
## 2                4.761905       3.8200    9.6  80.2200
## 3                4.761905      16.2155    7.4 340.5255
## 4                4.761905      23.2880    8.4 489.0480
## 5                4.761905      30.2085    5.3 634.3785
## 6                4.761905      29.8865    4.1 627.6165
```

```r
# Label encoding the categorical column Gender
data$Gender <- ifelse(data$Gender == "Male",1,2)
table(data$Gender)
```

```
##
```

```
##   1   2
## 499 501
```

```r
# label encoding the customer type column
data$Customer.type <- ifelse(data$Customer.type == "Member",1,2)
table(data$Customer.type)
```

```
##
##   1   2
## 501 499
```

```r
# label encoding the payment column
data$Payment <- as.numeric(data$Payment)
table(data$Payment)
```

```
##
##   1   2   3
## 344 311 345
```

```r
# label encoding the product line column
data$Product.line <- as.numeric(data$Product.line)
table(data$Product.line)
```

```
##
##   1   2   3   4   5   6
## 170 178 174 152 160 166
```

```r
# label encoding the branch column
data$Branch <- as.numeric(data$Branch)
table(data$Branch)
```

```
##
##   1   2   3
## 340 332 328
```

```r
# changing the new columns into factors
data$Day <- as.numeric(data$Day)
data$Month <- as.numeric(data$Month)
data$Year <- as.numeric(data$Year)
data$Hour <- as.numeric(data$Hour)
data$Minute <- as.numeric(data$Minute)
#data$Branch <- as.factor(data$Branch)
#data$Customer.type <- as.factor(data$Customer.type)
#data$Gender <- as.factor(data$Gender)
#data$Product.line <- as.factor(data$Product.line)
#data$Payment <- as.factor(data$Payment)
```

```r
# removing the columns that are not needed
data$Invoice.ID<-NULL
data$Year<-NULL
data$gross.margin.percentage<-NULL
```

```r
# loading the required package
library(caret)
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

To conduct feature selection, I will use a filter method to filter out variables that have high pairwise correlation.

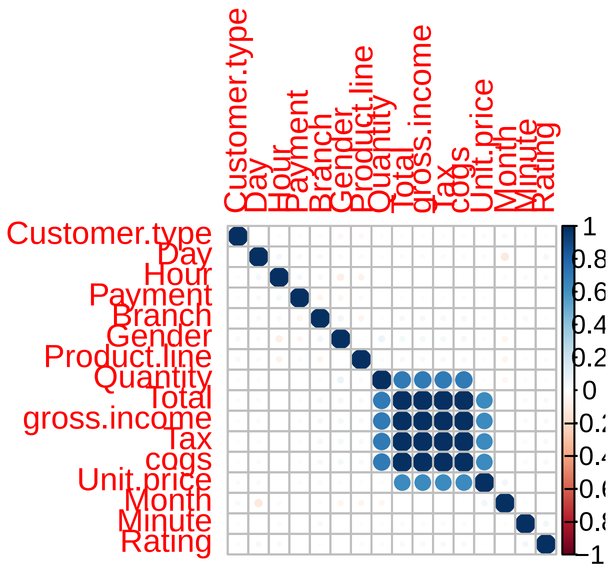
```r
head(data)
```

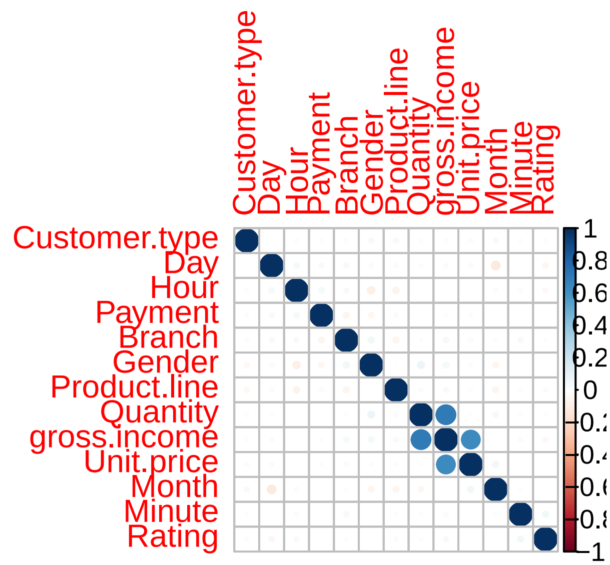
```
##   Branch Customer.type Gender Product.line Unit.price Quantity     Tax Day
## 1      1             1      2            4      74.69        7 26.1415   1
## 2      3             2      2            1      15.28        5  3.8200   3
## 3      1             2      1            5      46.33        7 16.2155   3
## 4      1             1      1            4      58.22        8 23.2880   1
## 5      1             2      1            6      86.31        7 30.2085   2
## 6      3             2      1            1      85.39        7 29.8865   3
##   Month Hour Minute Payment   cogs gross.income Rating    Total
## 1     5   13      8       3 522.83      26.1415    9.1 548.9715
## 2     8   10     29       1  76.40       3.8200    9.6  80.2200
## 3     3   13     23       2 324.31      16.2155    7.4 340.5255
## 4    27   20     33       3 465.76      23.2880    8.4 489.0480
## 5     8   10     37       3 604.17      30.2085    5.3 634.3785
## 6    25   18     30       3 597.73      29.8865    4.1 627.6165
```

```r
# generate the correlation matrix
correlation <- cor(data)

# obtaining the variables that are highly correlated using a cutoff of 75%
most_corr <- findCorrelation(correlation, cutoff=0.75)

# obtaining the names the higly correlated variables
names(data[,most_corr])
```

```
## [1] "cogs"  "Total" "Tax"
```

```r
# removing the highly correlated values
data2<-data[-most_corr]

# Performing our graphical comparison
# ---
#
par(mfrow = c(1, 2))
corrplot(correlation, order = "hclust")
corrplot(cor(data2), order = "hclust")
```





The most correlated variables are cogs total and tax. These variables distort the data therefore they should be filtered out. After filtering them out there is less correlation.