Dimensionality Reduction and Feature Selection

Joshua Kibuye

2022-04-04

Carrefour Kenya Marketing Strategies

1. Defining the Question

a) Specifying the Data Analytic Question.

What are most relevant marketing strategies that will result in the highest no. sales at Carrefour Kenya.

b) Defining the Metric for Success

c) Understanding the context

You are a Data analyst at Carrefour Kenya and are currently undertaking a project that will inform the marketing department on the most relevant marketing strategies that will result in the highest no. of sales (total price including tax). Your project has been divided into four parts where you'll explore a recent marketing dataset by performing various unsupervised learning techniques and later providing recommendations based on your insights.

d) Recording the Experimental Design

- Effectively cleaning our dataset.
- Performing extensive exploratory data analysis where applicable.
- Applying Dimensionality Reduction.
- Selecting our features.
- Applying Association rules.
- Detecting anomalies in our data.

e) Data Relevance

2. Data Understanding

```r
#Loading our dataset
df <- read.csv('http://bit.ly/CarreFourDataset')
```

```r
#Looking at the top of the dataset
head(df)
```

```
##    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
#Looking at the tail of the dataset
tail(df)
```

```
##       Invoice.ID Branch Customer.type Gender           Product.line Unit.price
## 995  652-49-6720      C        Member Female Electronic accessories      60.95
## 996  233-67-5758      C        Normal   Male      Health and beauty      40.35
## 997  303-96-2227      B        Normal Female     Home and lifestyle      97.38
## 998  727-02-1313      A        Member   Male     Food and beverages      31.84
## 999  347-56-2442      A        Normal   Male     Home and lifestyle      65.82
## 1000 849-09-3807      A        Member Female    Fashion accessories      88.34
##      Quantity     Tax      Date  Time Payment   cogs gross.margin.percentage
## 995         1  3.0475 2/18/2019 11:40 Ewallet  60.95                4.761905
## 996         1  2.0175 1/29/2019 13:46 Ewallet  40.35                4.761905
## 997        10 48.6900  3/2/2019 17:16 Ewallet 973.80                4.761905
## 998         1  1.5920  2/9/2019 13:22    Cash  31.84                4.761905
## 999         1  3.2910 2/22/2019 15:33    Cash  65.82                4.761905
## 1000        7 30.9190 2/18/2019 13:28    Cash 618.38                4.761905
##      gross.income Rating     Total
## 995        3.0475    5.9   63.9975
## 996        2.0175    6.2   42.3675
## 997       48.6900    4.4 1022.4900
## 998        1.5920    7.7   33.4320
## 999        3.2910    4.1   69.1110
## 1000      30.9190    6.6  649.2990
```

```r
#Looking at the summary of the dataset
summary(df)
```

```
##   Invoice.ID           Branch          Customer.type        Gender         
##  Length:1000        Length:1000        Length:1000        Length:1000       
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##  Product.line         Unit.price       Quantity          Tax         
##  Length:1000        Min.   :10.08   Min.   : 1.00   Min.   : 0.5085  
##  Class :character   1st Qu.:32.88   1st Qu.: 3.00   1st Qu.: 5.9249  
##  Mode  :character   Median :55.23   Median : 5.00   Median :12.0880  
##                     Mean   :55.67   Mean   : 5.51   Mean   :15.3794  
##                     3rd Qu.:77.94   3rd Qu.: 8.00   3rd Qu.:22.4453  
##                     Max.   :99.96   Max.   :10.00   Max.   :49.6500  
##      Date               Time             Payment               cogs       
##  Length:1000        Length:1000        Length:1000        Min.   : 10.17  
##  Class :character   Class :character   Class :character   1st Qu.:118.50  
##  Mode  :character   Mode  :character   Mode  :character   Median :241.76  
##                                                           Mean   :307.59  
##                                                           3rd Qu.:448.90  
##                                                           Max.   :993.00  
##  gross.margin.percentage  gross.income        Rating          Total        
##  Min.   :4.762           Min.   : 0.5085   Min.   : 4.000   Min.   :  10.68  
##  1st Qu.:4.762           1st Qu.: 5.9249   1st Qu.: 5.500   1st Qu.: 124.42  
##  Median :4.762           Median :12.0880   Median : 7.000   Median : 253.85  
##  Mean   :4.762           Mean   :15.3794   Mean   : 6.973   Mean   : 322.97  
##  3rd Qu.:4.762           3rd Qu.:22.4453   3rd Qu.: 8.500   3rd Qu.: 471.35  
##  Max.   :4.762           Max.   :49.6500   Max.   :10.000   Max.   :1042.65
```

```r
#Getting the shape of the dataset
dim(df)
```

```
## [1] 1000   16
```

There are 1,000 records and 16 variables. ## 3. Data Cleaning

```r
#Checking for missing data
sum(is.null(df))
```

```
## [1] 0
```

There are no missing values in the dataset.

```r
#Checking for duplicates
sum(duplicated(df))
```

```
## [1] 0
```

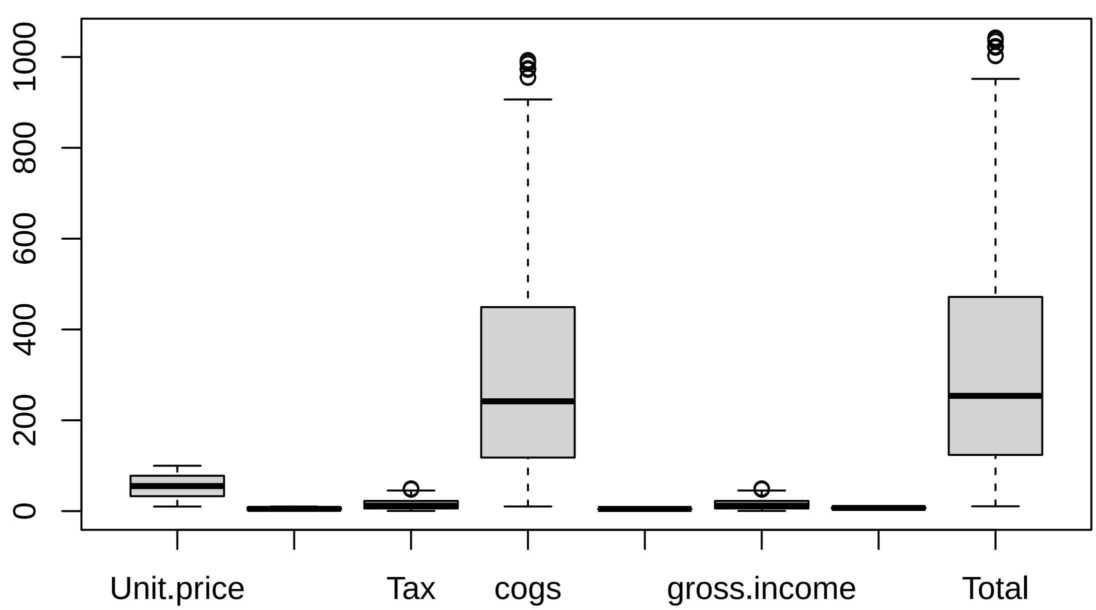
There are no duplicates in the dataset.

```r
#Defining numerical columns
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## Warning: package 'ggplot2' was built under R version 4.1.3

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

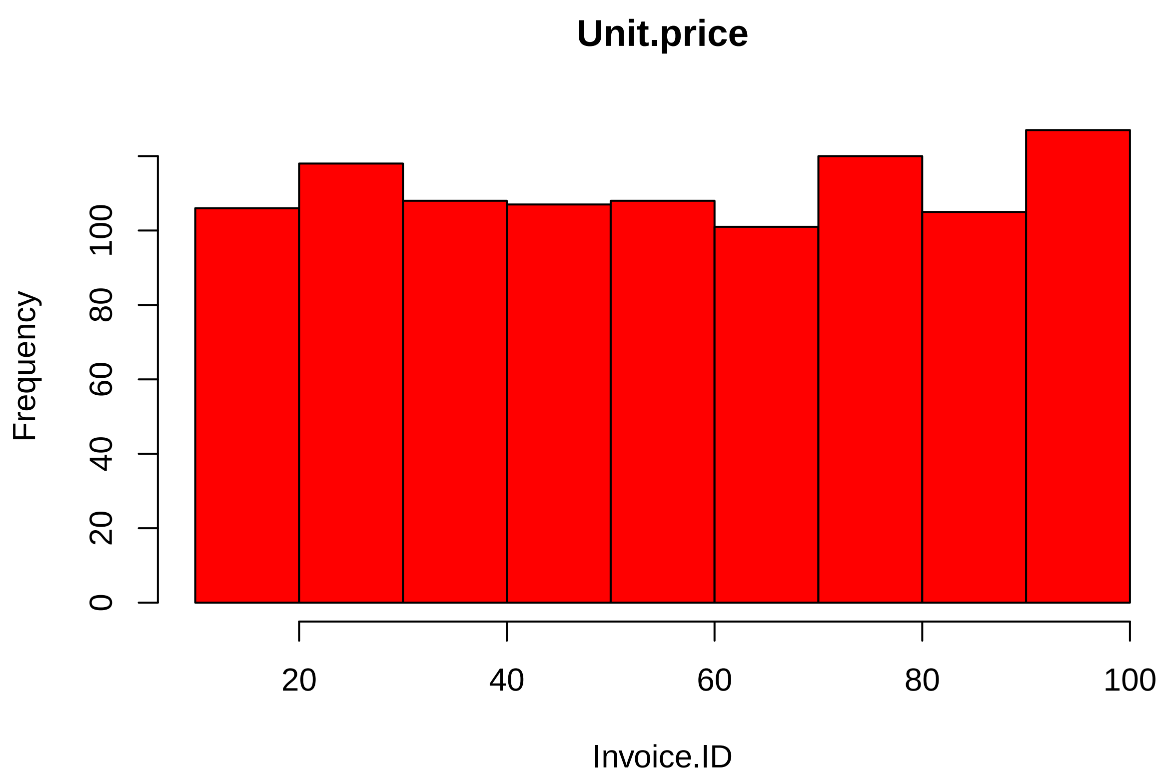
```r
numeric <- df%>%select_if(is.numeric)
```

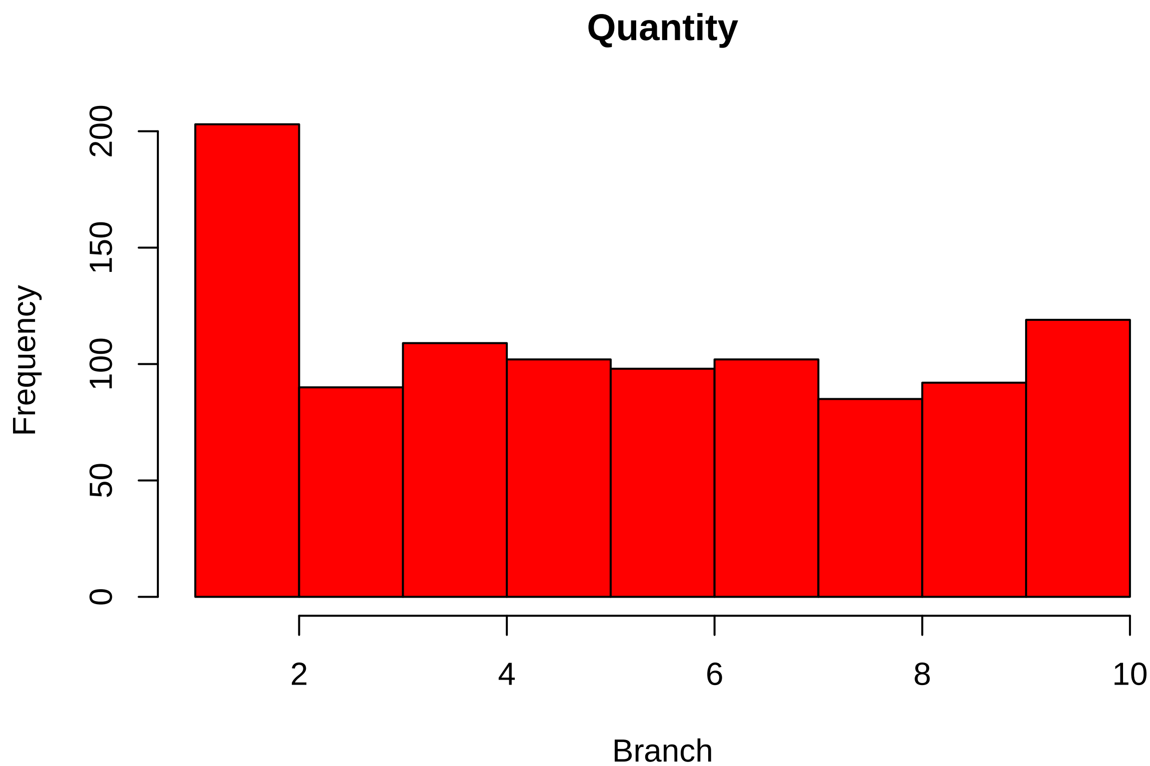
```r
#Checking for outliers in the numerical dataset
boxplot(numeric)
```

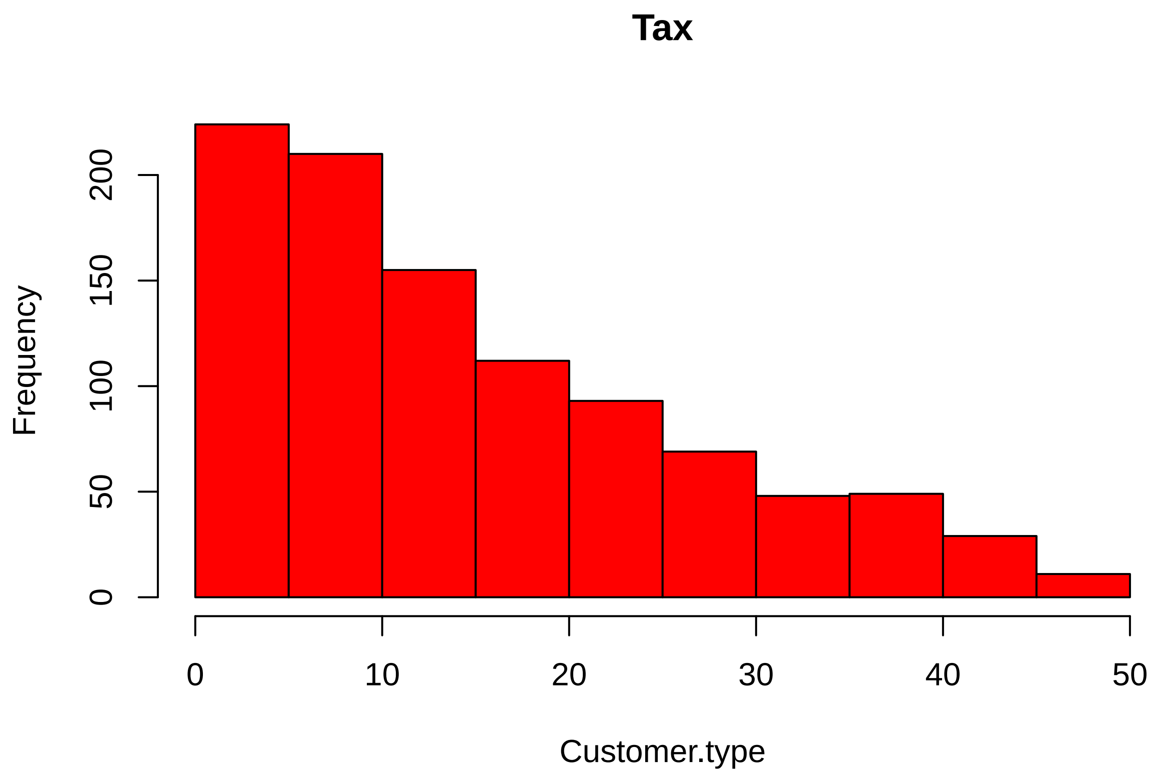


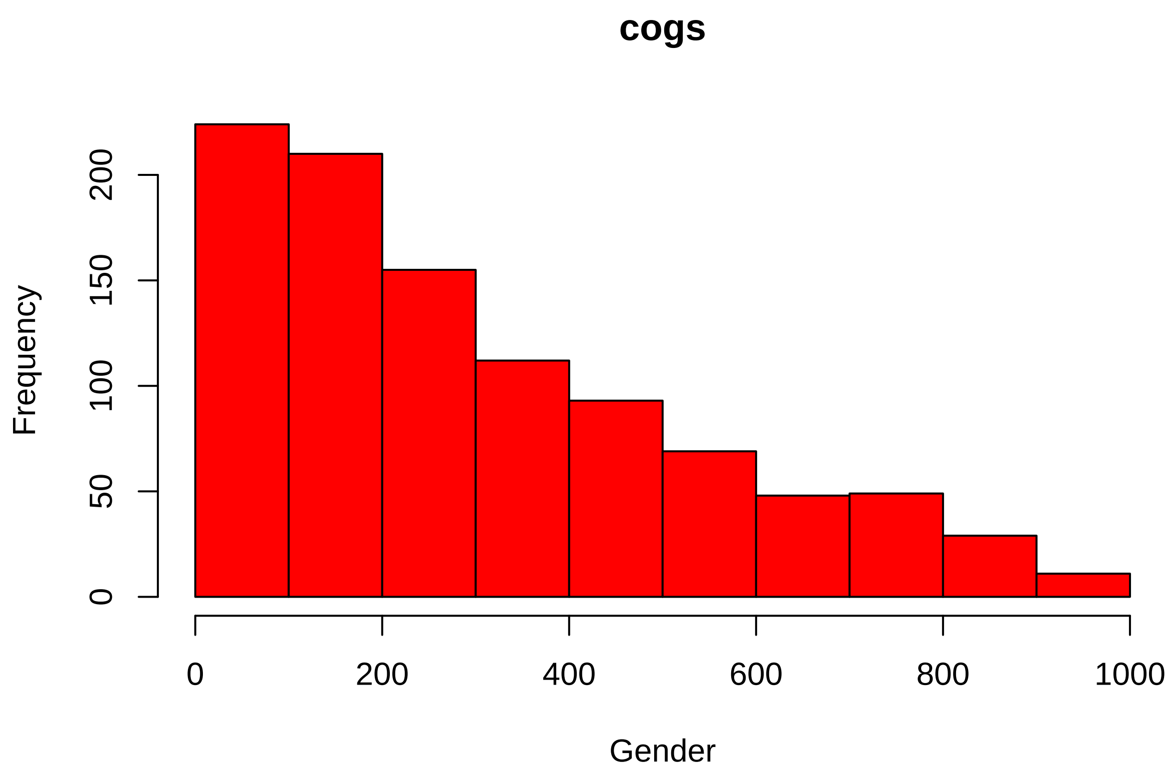
There is presence of some outliers in the dataset. However, we don't drop them as they are true values. ## 4. Exploratory Data Analysis ### 4.1 Univariate Analysis

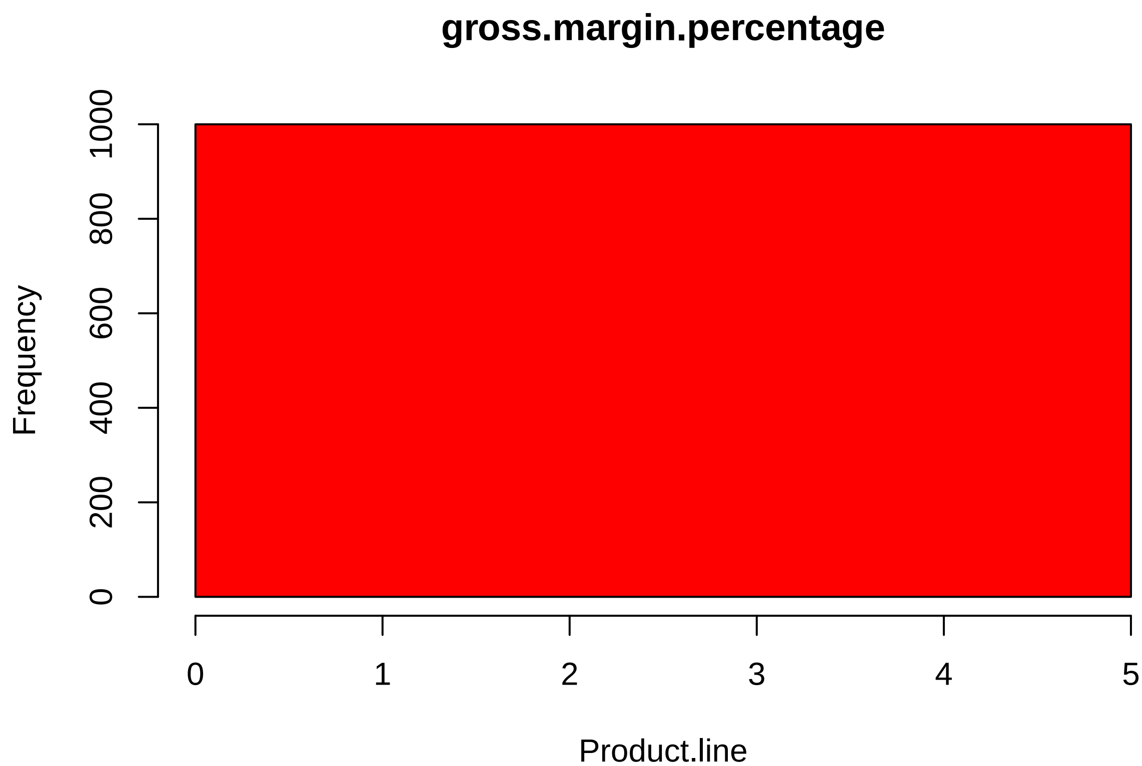
```r
#Histogram of numerical columns
for(i in 1:8) {
    hist(numeric[,i], main=names(numeric)[i], xlab=names(df)[i],col = "red")}
```

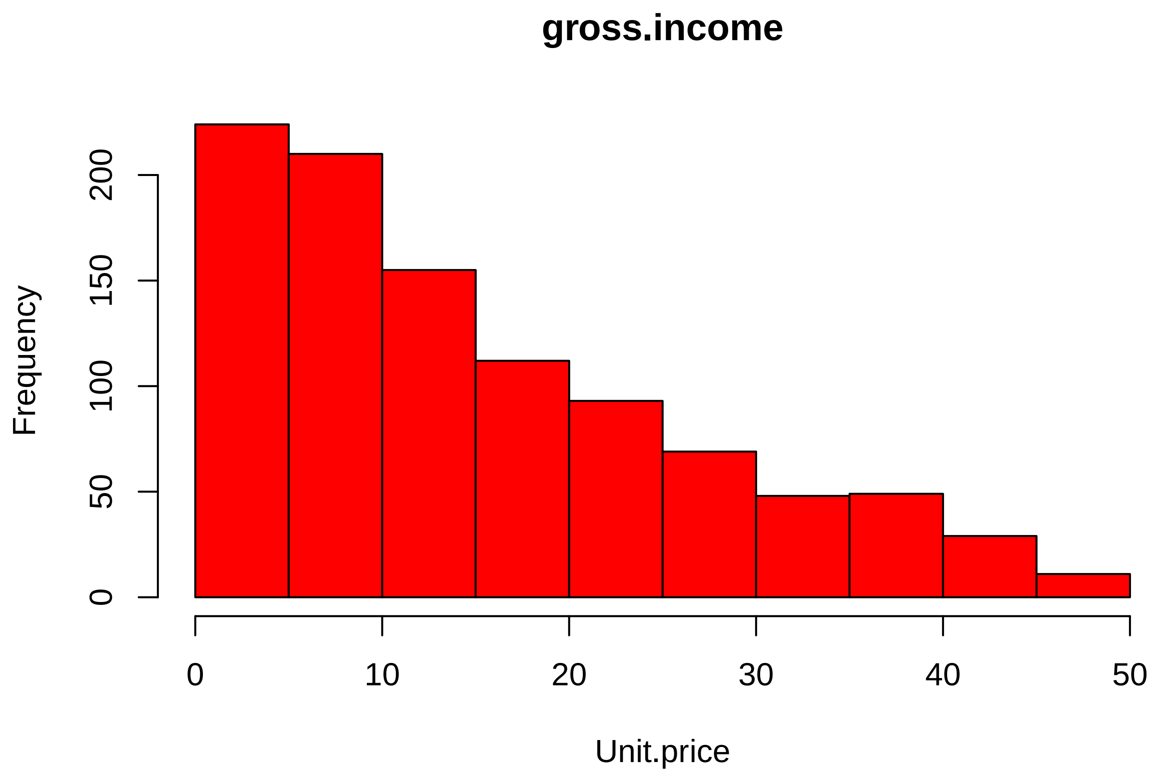


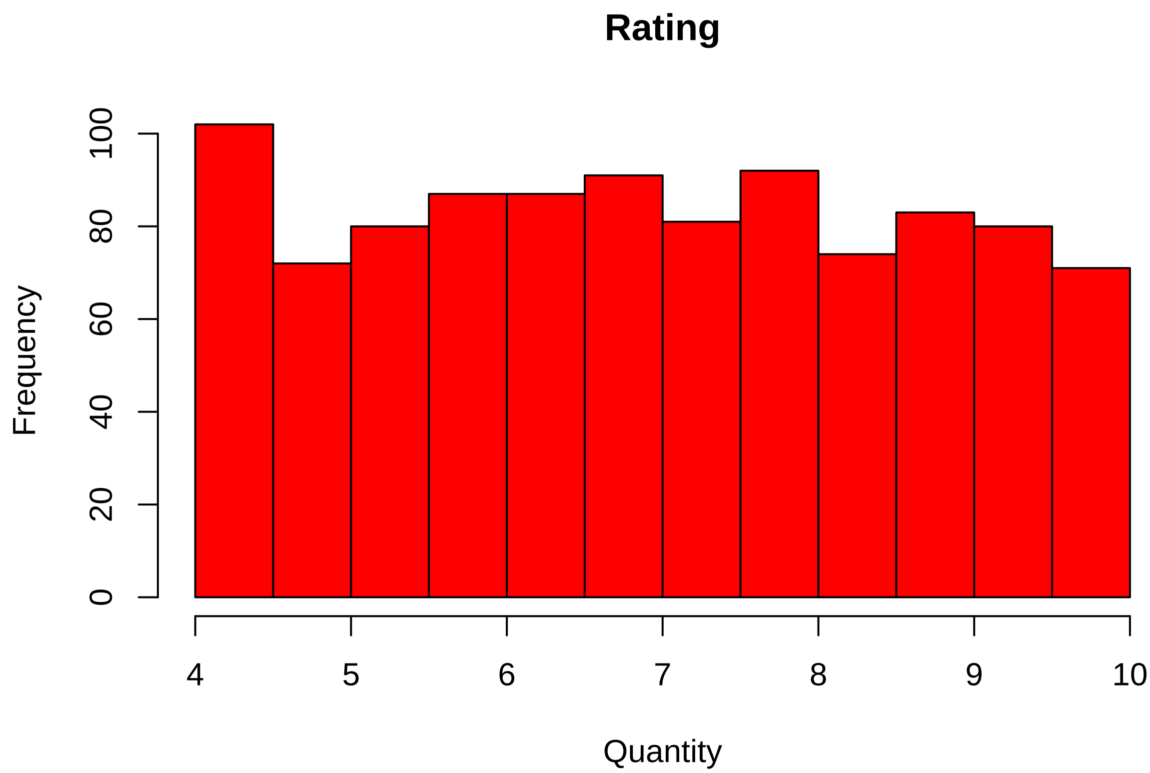

Quantity
Frequency
0
50
100
150
200
2
4
6
8
10
Branch

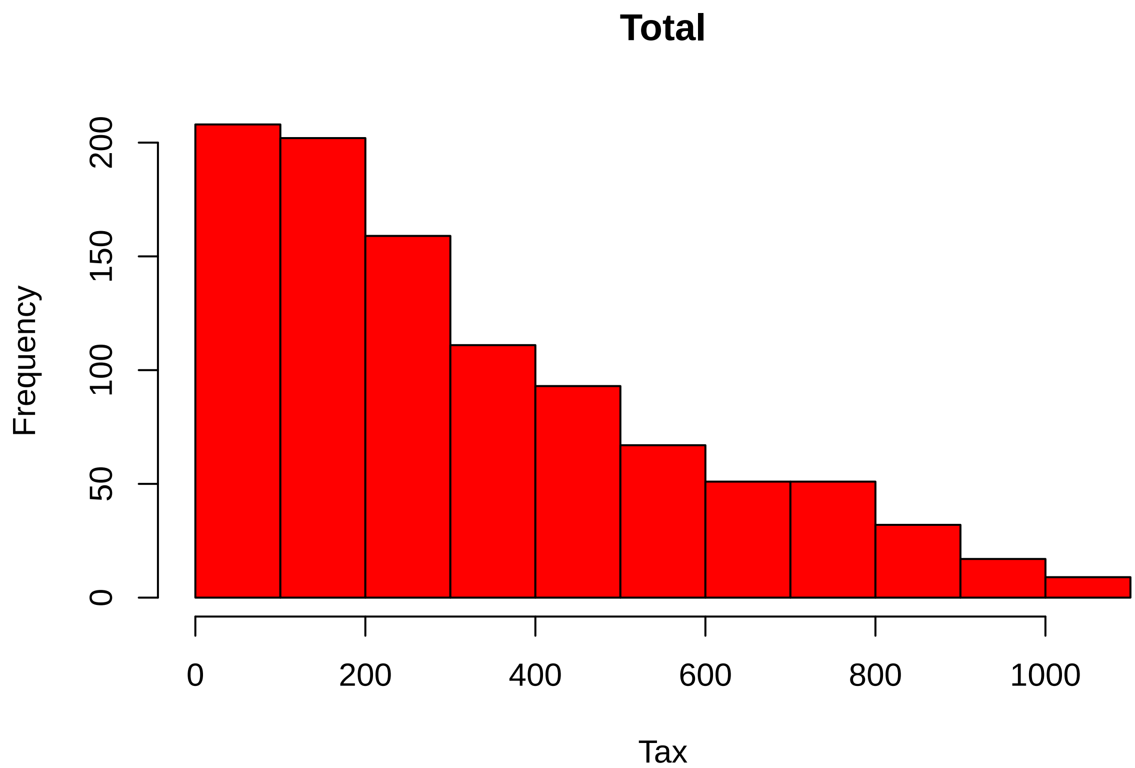

Tax
Frequency
0
50
100
150
200
0
10
20
30
40
50
Customer.type


cogs
Frequency
0
50
100
150
200
0
200
400
600
800
1000
Gender


gross.margin.percentage
Frequency
0
200
400
600
800
1000
0
1
2
3
4
5
Product.line


gross.income
Frequency
0
50
100
150
200
0
10
20
30
40
50
Unit.price


Rating
Frequency
0
20
40
60
80
100
4
5
6
7
8
9
10
Quantity




```
#Getting a statistical summary of numerical columns
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.1.3
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

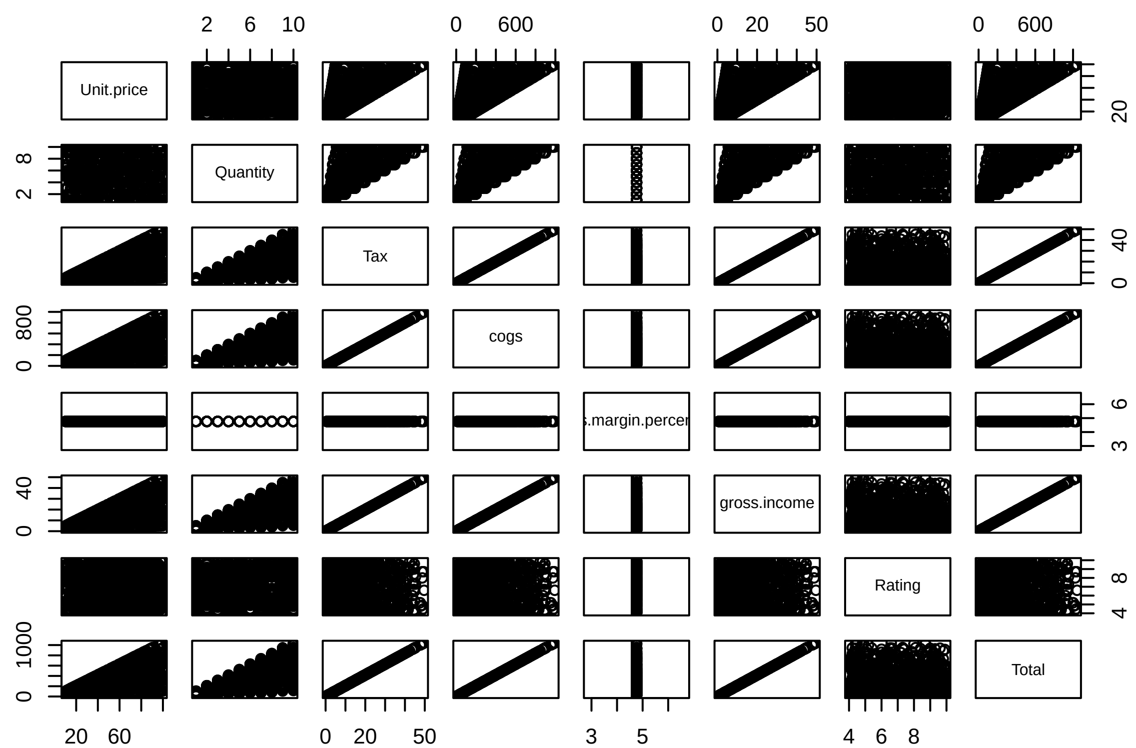
```
describe(numeric)
```

```
##                         vars    n   mean     sd median trimmed    mad   min
## Unit.price                 1 1000  55.67  26.49  55.23   55.62  33.37 10.08
## Quantity                   2 1000   5.51   2.92   5.00    5.51   2.97  1.00
## Tax                        3 1000  15.38  11.71  12.09   14.00  11.13  0.51
## cogs                       4 1000 307.59 234.18 241.76  279.91 222.65 10.17
## gross.margin.percentage    5 1000   4.76   0.00   4.76    4.76   0.00  4.76
## gross.income               6 1000  15.38  11.71  12.09   14.00  11.13  0.51
## Rating                     7 1000   6.97   1.72   7.00    6.97   2.22  4.00
## Total                      8 1000 322.97 245.89 253.85  293.91 233.78 10.68
##                             max   range skew kurtosis   se
```

```
## Unit.price                  99.96   89.88 0.01    -1.22 0.84
## Quantity                    10.00    9.00 0.01    -1.22 0.09
## Tax                         49.65   49.14 0.89    -0.09 0.37
## cogs                       993.00  982.83 0.89    -0.09 7.41
## gross.margin.percentage      4.76    0.00  NaN      NaN 0.00
## gross.income                49.65   49.14 0.89    -0.09 0.37
## Rating                      10.00    6.00 0.01    -1.16 0.05
## Total                     1042.65 1031.97 0.89    -0.09 7.78
```

Statistical information is stored in a dataframe ## 4.2 Bivariate Analysis

```
#Pairplot of numerical columns
plot(numeric)
```




```
# calculate correlations
library(corrplot)
```

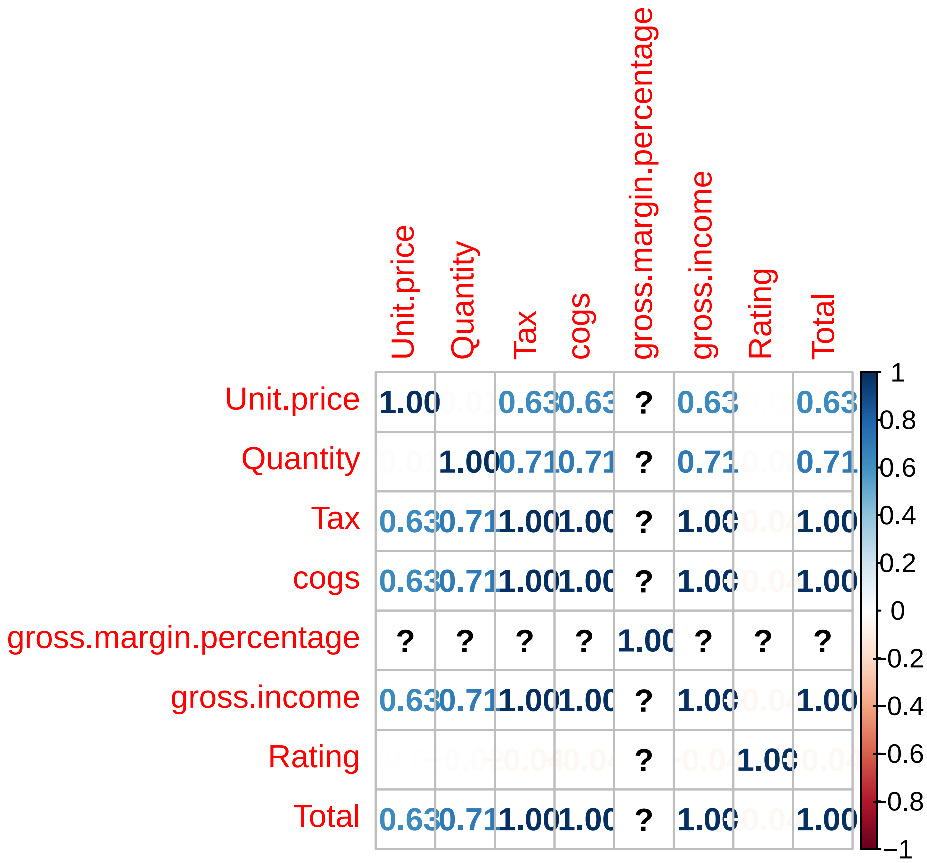
```
## Warning: package 'corrplot' was built under R version 4.1.3
```

```
## corrplot 0.92 loaded
```

```
correlations <- cor(numeric)
```

```
## Warning in cor(numeric): the standard deviation is zero
```

```r
# create correlation plot
corrplot(correlations, method="number")
```



5. Dimensionality Reduction (PCA)

```r
num_var <- df[ , which(apply(df, 2, var) != 0)]
```

```
## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion

## Warning in FUN(newX[, i], ...): NAs introduced by coercion
```

```r
head(num_var)
```

```
##   Unit.price Quantity     Tax   cogs gross.income Rating    Total
## 1      74.69        7 26.1415 522.83      26.1415    9.1 548.9715
## 2      15.28        5  3.8200  76.40       3.8200    9.6  80.2200
## 3      46.33        7 16.2155 324.31      16.2155    7.4 340.5255
## 4      58.22        8 23.2880 465.76      23.2880    8.4 489.0480
## 5      86.31        7 30.2085 604.17      30.2085    5.3 634.3785
## 6      85.39        7 29.8865 597.73      29.8865    4.1 627.6165
```

There is no zero variance.

```r
# Previewing our PCAs
num_var.pca <- prcomp(num_var, center = TRUE, scale. = TRUE)
summary(num_var.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4       PC5       PC6
## Standard deviation     2.2185 1.0002 0.9939 0.30001 2.981e-16 1.493e-16
## Proportion of Variance 0.7031 0.1429 0.1411 0.01286 0.000e+00 0.000e+00
## Cumulative Proportion  0.7031 0.8460 0.9871 1.00000 1.000e+00 1.000e+00
##                              PC7
## Standard deviation     9.831e-17
## Proportion of Variance 0.000e+00
## Cumulative Proportion  1.000e+00
```

We obtain seven principle components. The first principle component explains 70% of the variance.

```r
# Calling str() to have a look at your PCA object
str(num_var.pca)
```

```
## List of 5
##  $ sdev    : num [1:7] 2.22 1.00 9.94e-01 3.00e-01 2.98e-16 ...
##  $ rotation: num [1:7, 1:7] -0.292 -0.325 -0.45 -0.45 -0.45 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:7] "Unit.price" "Quantity" "Tax" "cogs" ...
##   .. ..$ : chr [1:7] "PC1" "PC2" "PC3" "PC4" ...
##  $ center  : Named num [1:7] 55.67 5.51 15.38 307.59 15.38 ...
##   ..- attr(*, "names")= chr [1:7] "Unit.price" "Quantity" "Tax" "cogs" ...
##  $ scale   : Named num [1:7] 26.49 2.92 11.71 234.18 11.71 ...
##   ..- attr(*, "names")= chr [1:7] "Unit.price" "Quantity" "Tax" "cogs" ...
##  $ x       : num [1:1000, 1:7] -2.005 2.306 -0.186 -1.504 -2.8 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:7] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"
```

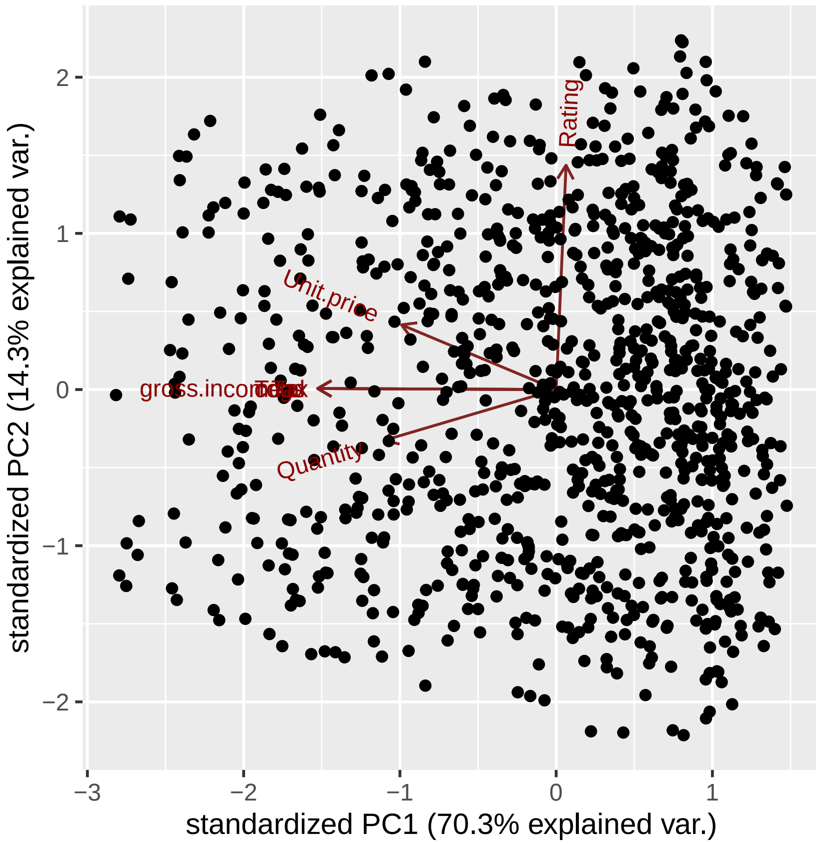
```r
#Visualizing the results
library(ggbiplot)
```

```
## Loading required package: plyr
```

```
## Warning: package 'plyr' was built under R version 4.1.3
```

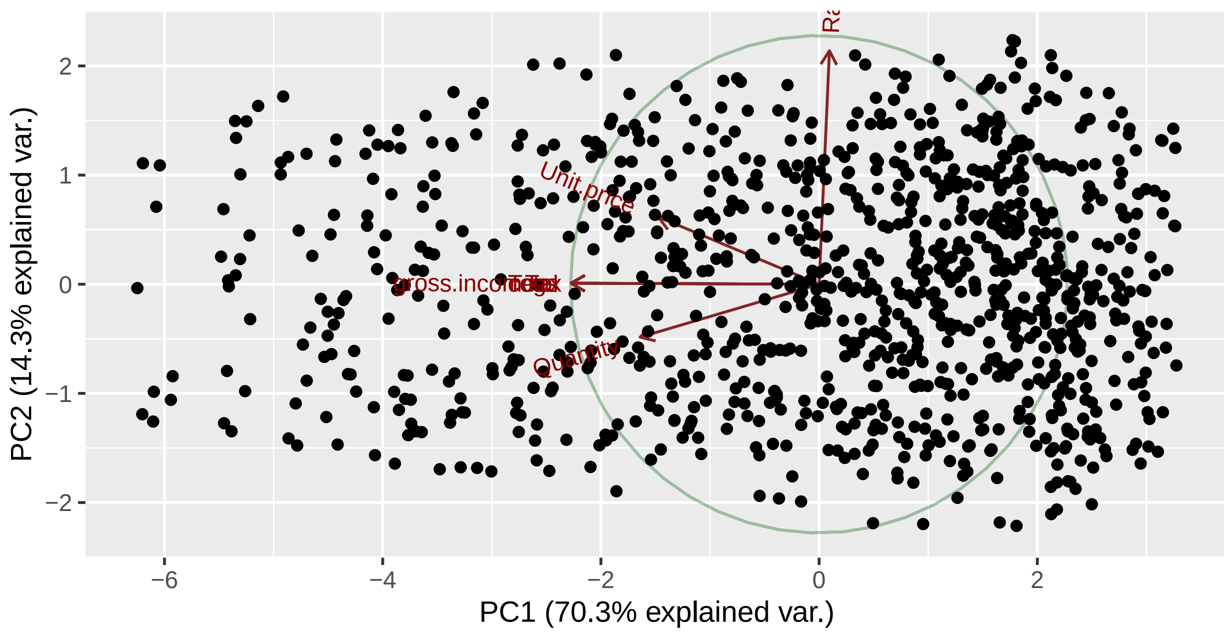
```
## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following object is masked from 'package:purrr':
##
##     compact

## Loading required package: scales

##
## Attaching package: 'scales'

## The following objects are masked from 'package:psych':
##
##     alpha, rescale

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

## Loading required package: grid
```

```
ggbiplot(num_var.pca)
```



We have a good plot but more details can be obtained.

```
#Getting more detailed information on the dataset
ggbiplot(num_var.pca, obs.scale = 1, var.scale = 1,
  groups = num_var.pca$Total, ellipse = TRUE, circle = TRUE,ellipse.prob = 0.68) +
  scale_color_discrete(name = '') +
  theme(legend.direction = 'horizontal', legend.position = 'top')
```



7. Feature Selection

```r
#Removing variables with a standard deviation of 0
sdf <- numeric %>% select(-gross.margin.percentage)
```

The dataframe will be used for analysis.

```r
#Creating correlation matrix
cor <- cor(sdf)
```

```r
#Now to deal with highly correlated variables
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.3
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
high_corr <- findCorrelation(cor, cutoff=0.75)
names(sdf[,high_corr])
```

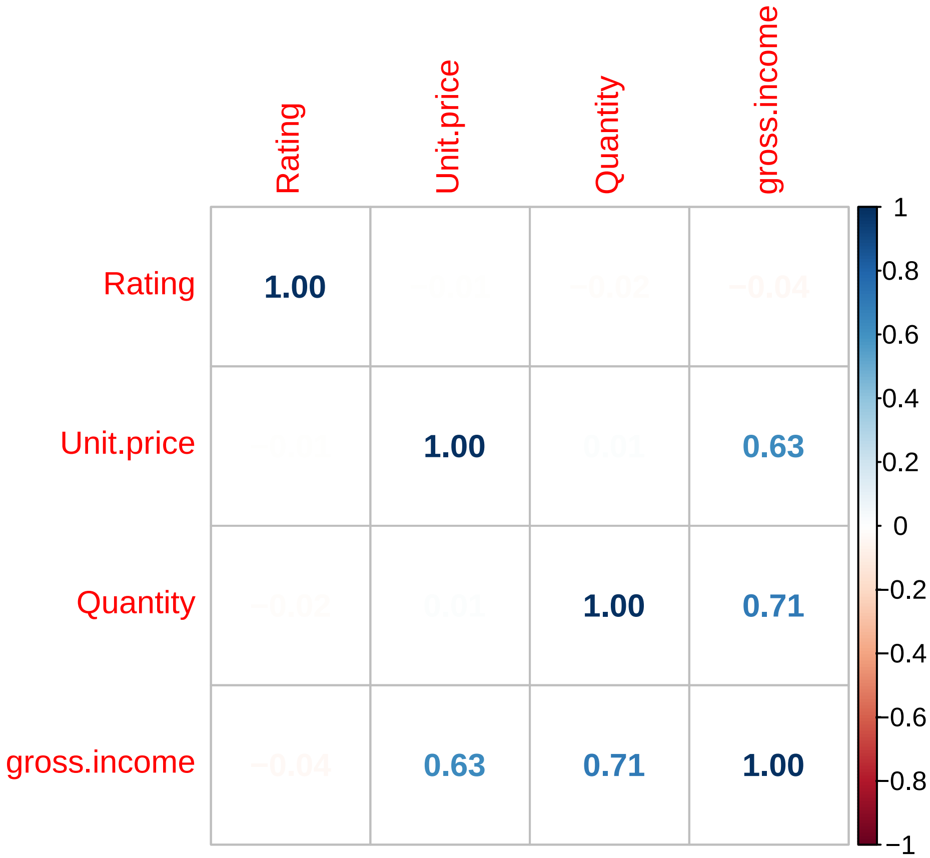
```
## [1] "cogs"  "Total" "Tax"
```

Cogs, Total, and Tax are highly correlated.

```r
#Dropping the highly correlated functions
df1 <- sdf[-high_corr]
```

```r
#Comparing correlation before and after dropping elements
corrplot(cor, order = "hclust", method = "number")
```


```r
corrplot(cor(df1), order = "hclust", method = "number")
```



The correlation is much better than in the original.

Conclusion

We find that the most important features are rating, unit price, quantity, and gross income.