Independent_Project_Week_14

Jonah okiru

2022-06-10

1. Problem definition

a). Specifying the question.

To identify the most relevant marketing strategies that will result in the highest number of sales.

b). The metric of success.

To come up with an t-NSE model and Filter methond model that will able to map high dimension dataset into low dimension dataset to aid in the visualization of the data.

c). The context

As the data analyst at carrefour Kenya i undertake the project that will help to come up with the sales strategies that will help Carrefour to increase its sales.

d). The Experimental design.

-Loading the data.

-Check the data.

-Clean the data.

-univariate analysis.

-Bivariate analysis.

- Conclusion.

-Recommendation

-Challenge the solution

-Follow up question

2). Data sourcing

The data fro the project was provided the intitution.

3. Load the data

```
df = read.csv("D:/R studio/week3 R/Supermarket_Dataset.csv")
```

4. Check for the data

```
#Load the data,table library
library("data.table")
#Preview the first 4 records of the data
head(df, 4)
```

```
##    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
##   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
##   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
```

:

```
#Check the number of columns in the dataset
ncol(df)
```

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

The dataset has 16 columns

```
#Check for the number of rows in the dataset
nrow(df)
```

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

The dataset has 1000 records

```r
#Check the datatype of each column
str(df)
```

```
## 'data.frame':    1000 obs. of  16 variables:
##  $ Invoice.ID             : chr  "750-67-8428" "226-31-3081" "631-41-3108" "123-19-1176" ...
##  $ Branch                 : chr  "A" "C" "A" "A" ...
##  $ Customer.type          : chr  "Member" "Normal" "Normal" "Member" ...
##  $ Gender                 : chr  "Female" "Female" "Male" "Male" ...
##  $ Product.line           : chr  "Health and beauty" "Electronic accessories" "Home and lifestyle" "
##  $ 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                   : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ Time                   : chr  "13:08" "10:29" "13:23" "20:33" ...
##  $ Payment                : chr  "Ewallet" "Cash" "Credit card" "Ewallet" ...
##  $ 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 ...
```

The dataset column is of the following datatypes; 8 columns are of string, 7 columns are of numerical datatypes and 1 column is of interger datatype

```r
#Check for the missing values in the dataset
colSums(is.na(df))
```

```
##              Invoice.ID                  Branch           Customer.type
##                       0                       0                       0
##                  Gender            Product.line              Unit.price
##                       0                       0                       0
##                Quantity                     Tax                    Date
##                       0                       0                       0
##                    Time                 Payment                    cogs
##                       0                       0                       0
## gross.margin.percentage            gross.income                  Rating
##                       0                       0                       0
##                   Total
##                       0
```

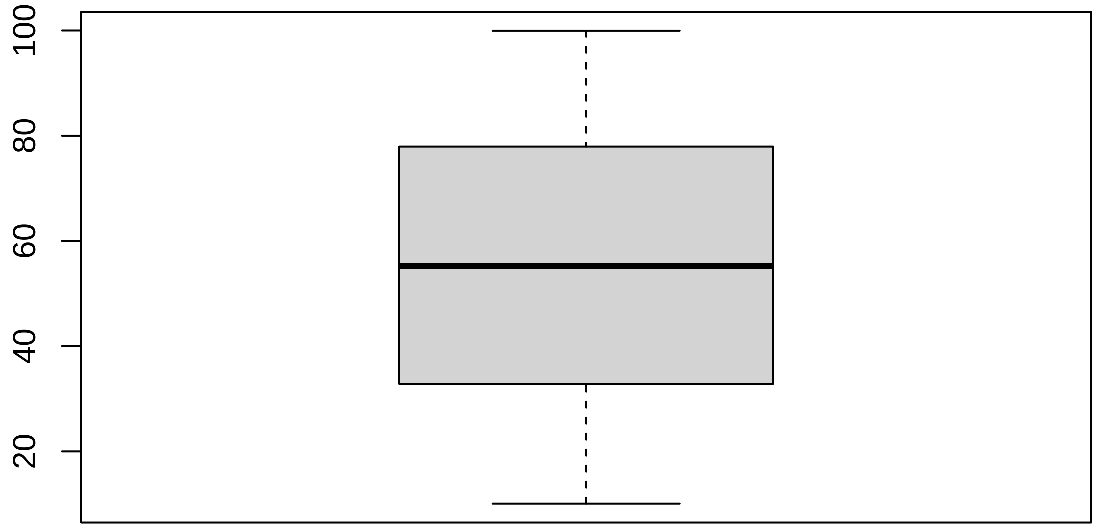
The dataset has no missing values

```r
#Check for the duplicates in the dataset
duplicates <- df[duplicated(df), ]
#Number of duplicate records
nrow(duplicates)
```

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

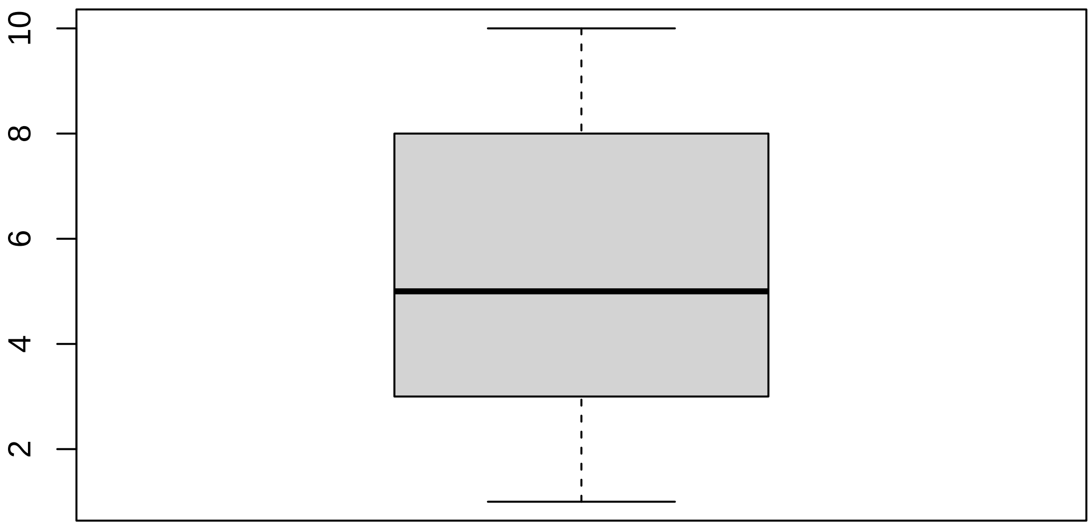
There is no duplicate records in the dataset

```r
#Check for the outliers in the Unit.price column
boxplot(df$Unit.price)
```



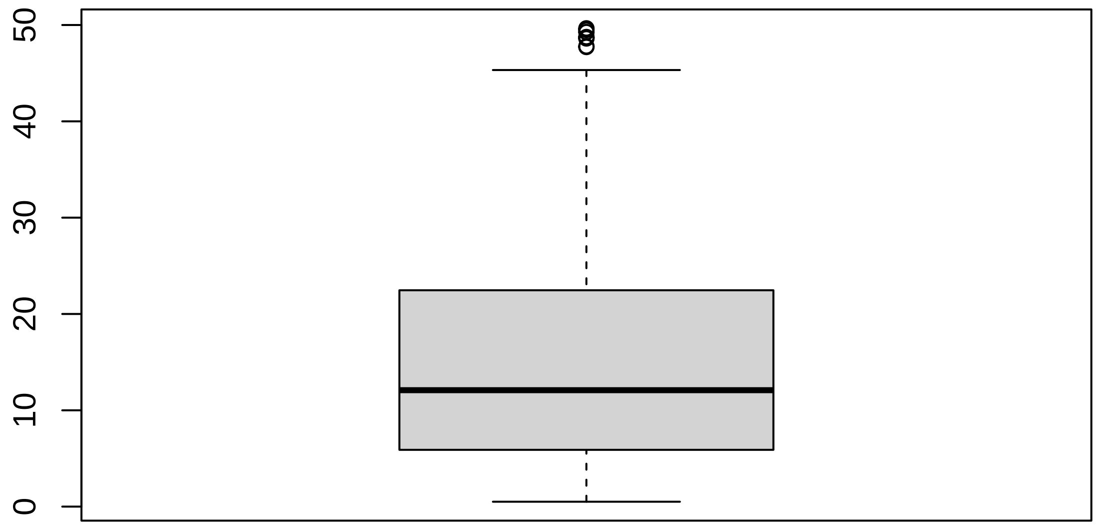
There is no outlier in the Unit.price column.

```r
#Check for the outliers in the Quantity column
boxplot(df$Quantity)
```



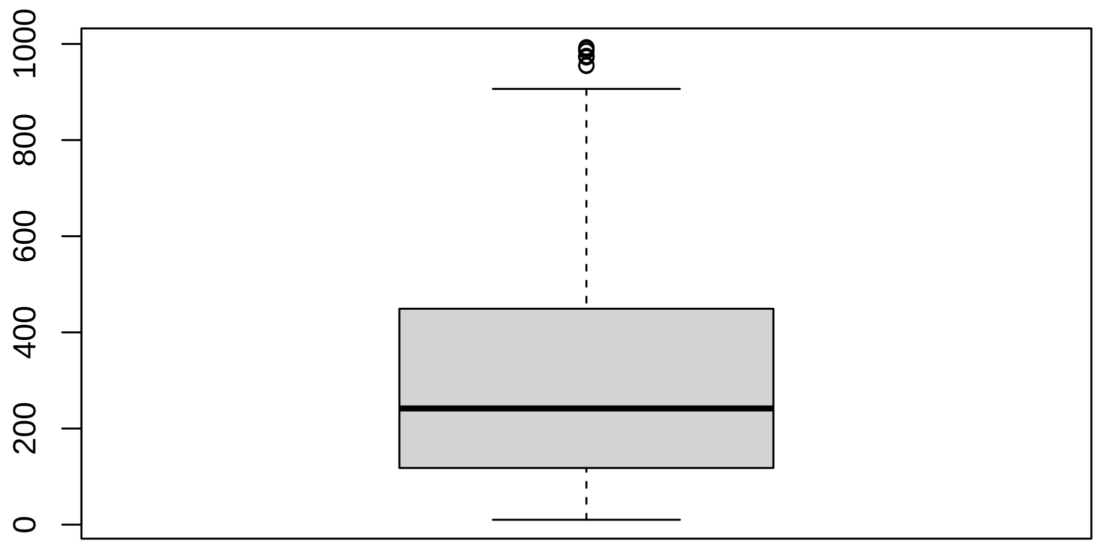
There is no outliers in the column of quantity

```
#Check for the outlier in the Tax column
boxplot(df$Tax)
```



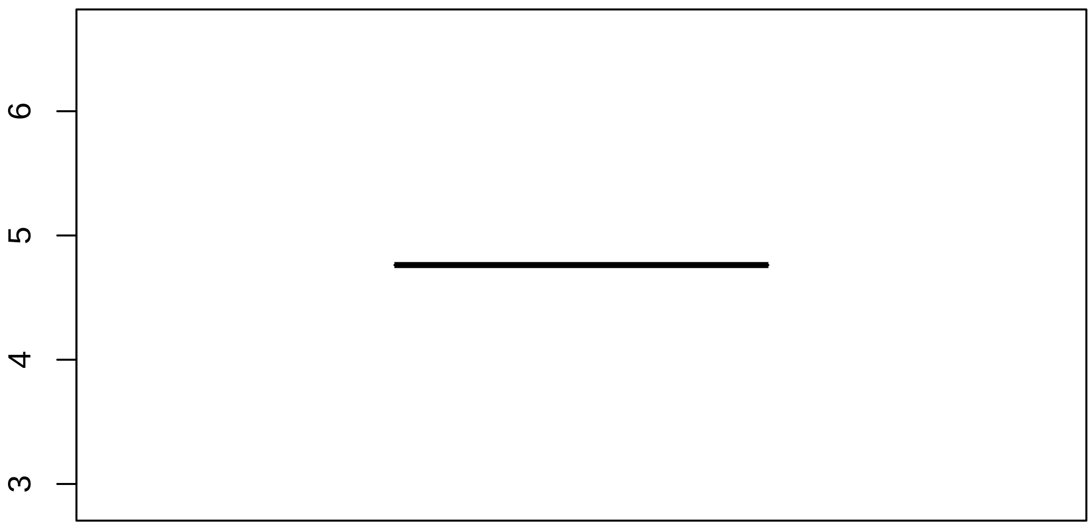
There exists outlier in the Tax column.

```
#Check for the outliers in the cogs column
boxplot(df$cogs)
```



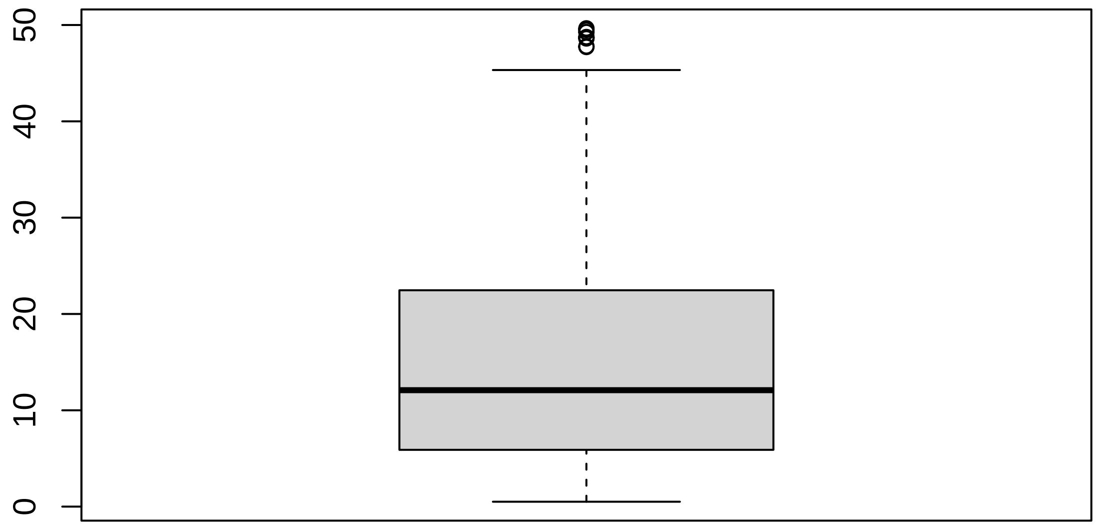
There exists an outlier in the column of cogs

```r
#Check for the existence of outliers in the column of gross.margin.percentage
boxplot(df$gross.margin.percentage)
```



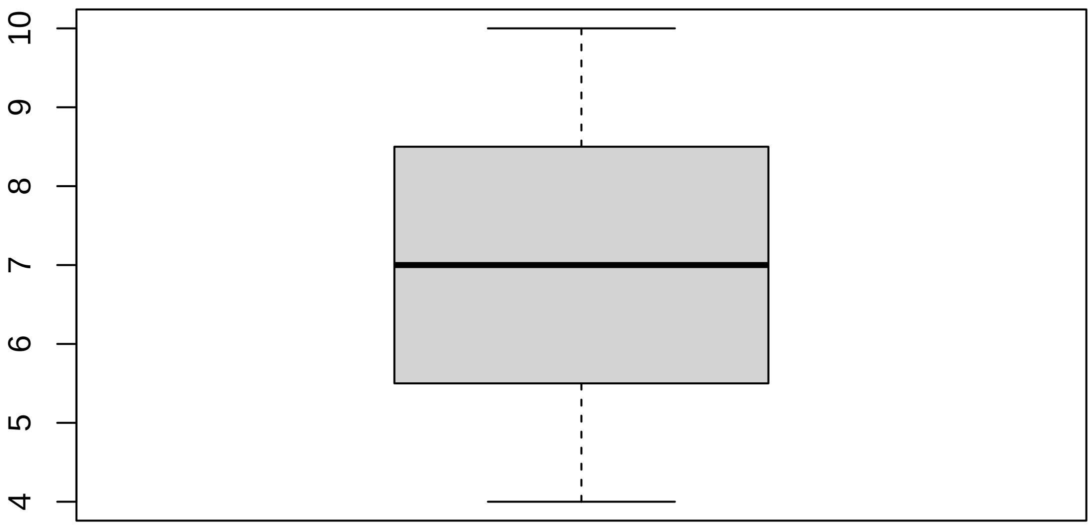
There's no outliers in the column of gross.margin.percentage.

```
#Check for the outliers in the column of gross.income
boxplot(df$gross.income)
```



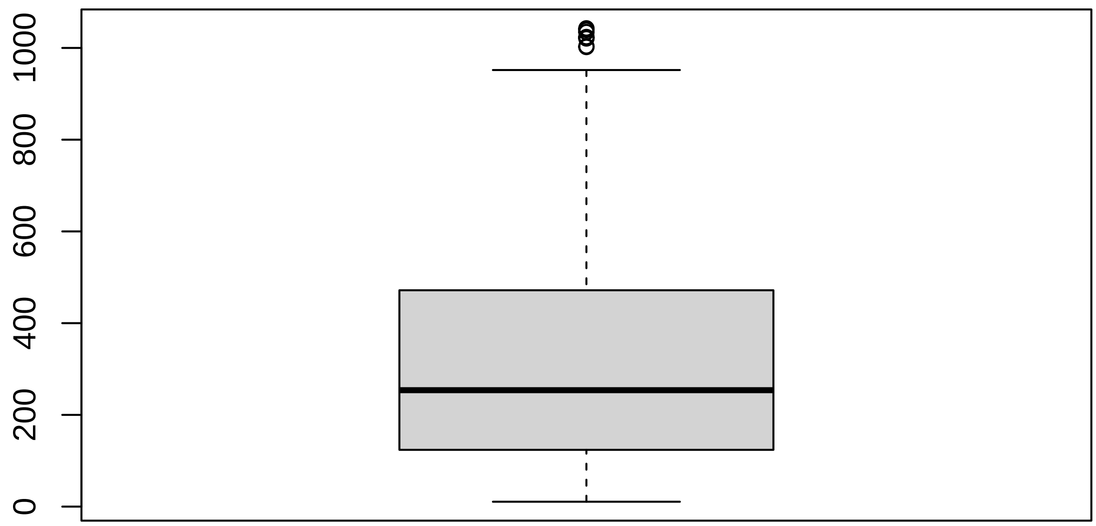
There exists the outliers in the column of gross.income

```
#Check for the outliers in the column of Rating
boxplot(df$Rating)
```



There is no outliers in the column of rating.

```
#Check for the existence of outliers in the column of Total
boxplot(df$Total)
```



There's outliers in the column of total.

4. Data cleaning

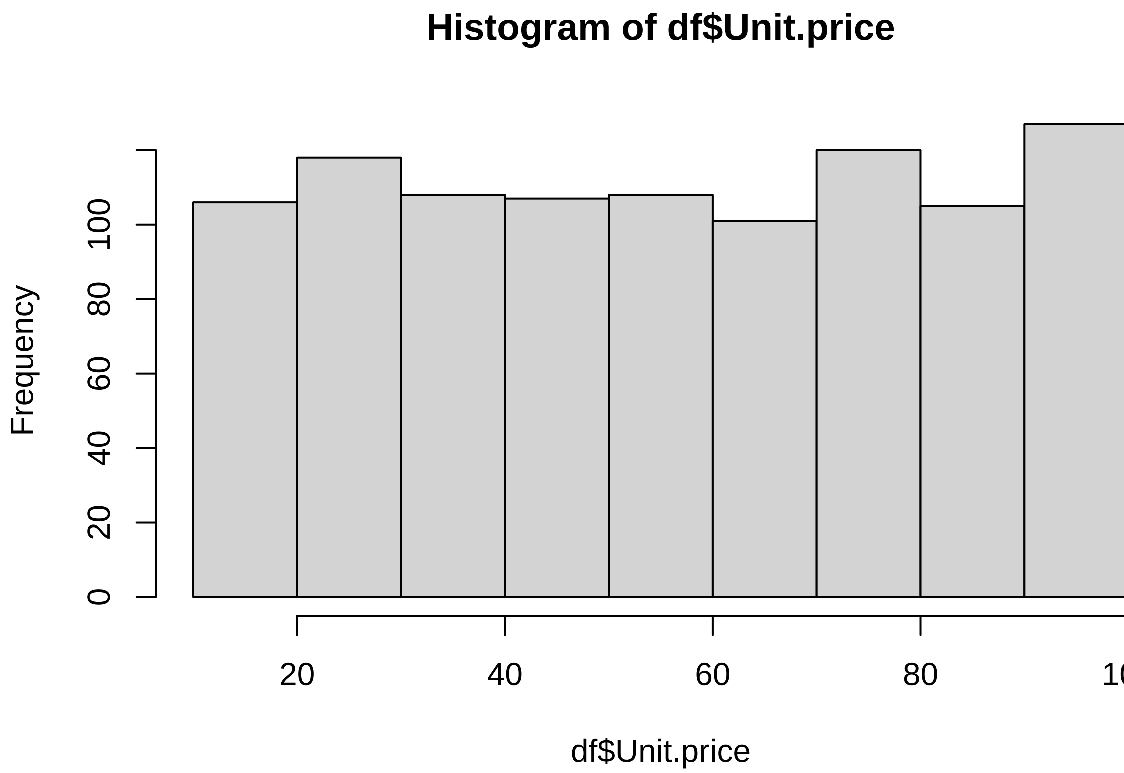
```
#Dealing with the outliers.
#The outliers will not be dropped
```

The outliers in the above dataset will be detained due to the following reasons; 1. In the tax column the outliers will be detained because the tax paid by different customers are not the due to difference in the quantity of items purchased by the customers. 2. In the gross income column the outliers will be detained due to difference gross income among customers 3. total column the outliers will be detained due difference in the purchase power of the customers.

5.Exploratory data analysis.

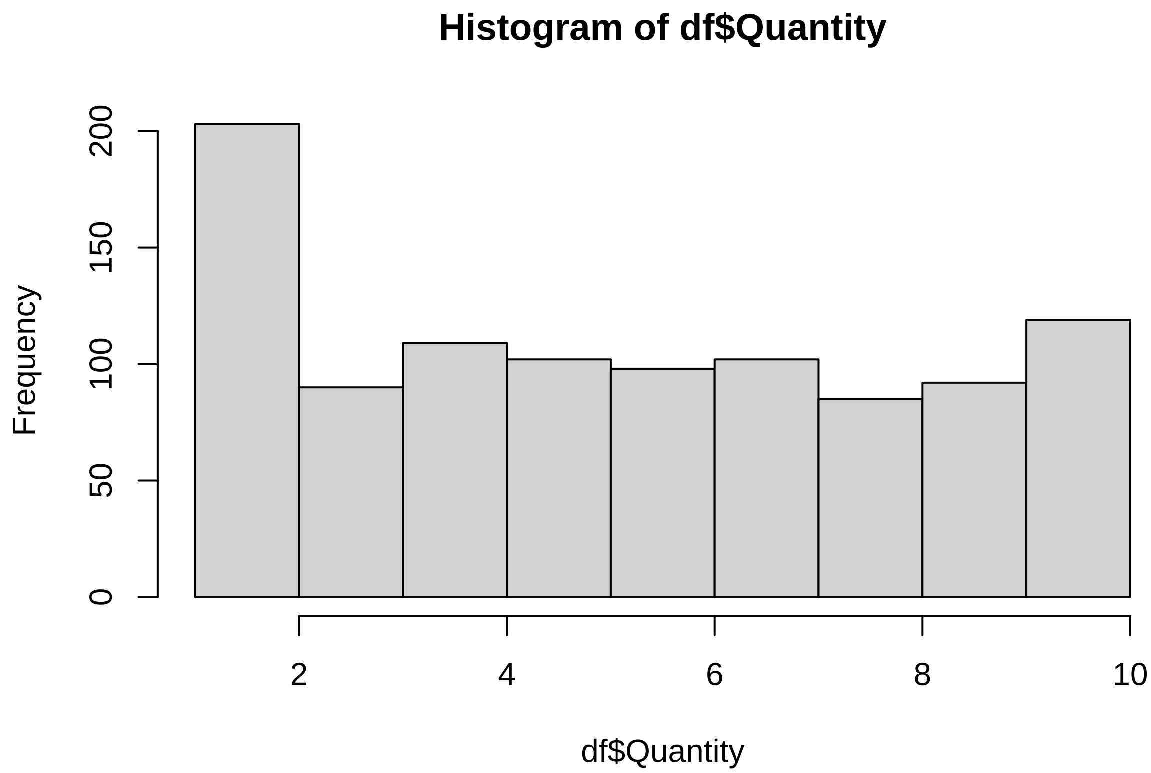
a). Univariate

```
#Histogram of Unit.price
hist(df$Unit.price)
```



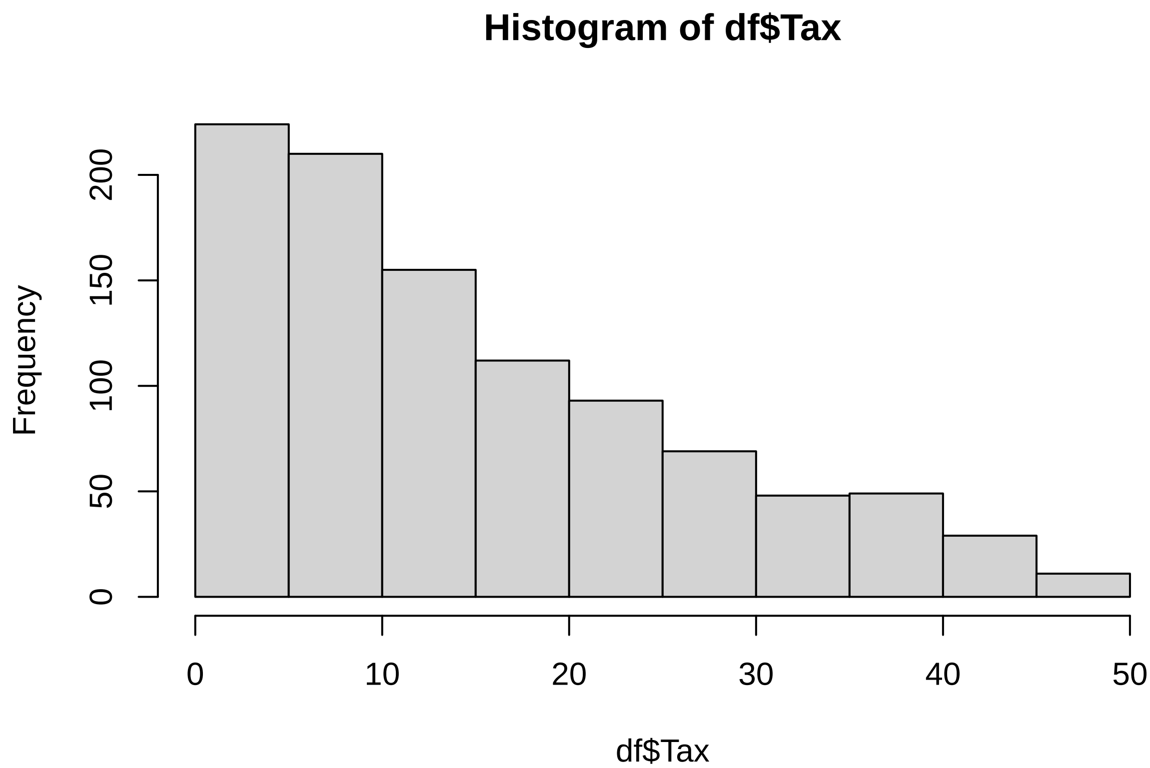
(i) Numerical analysis
The unit price is normally distributed. The unit price with the most frequency is the price of 20 to 30, 70 to 80 and 90 to 100.

```
#The histogram of Quantity
hist(df$Quantity)
```



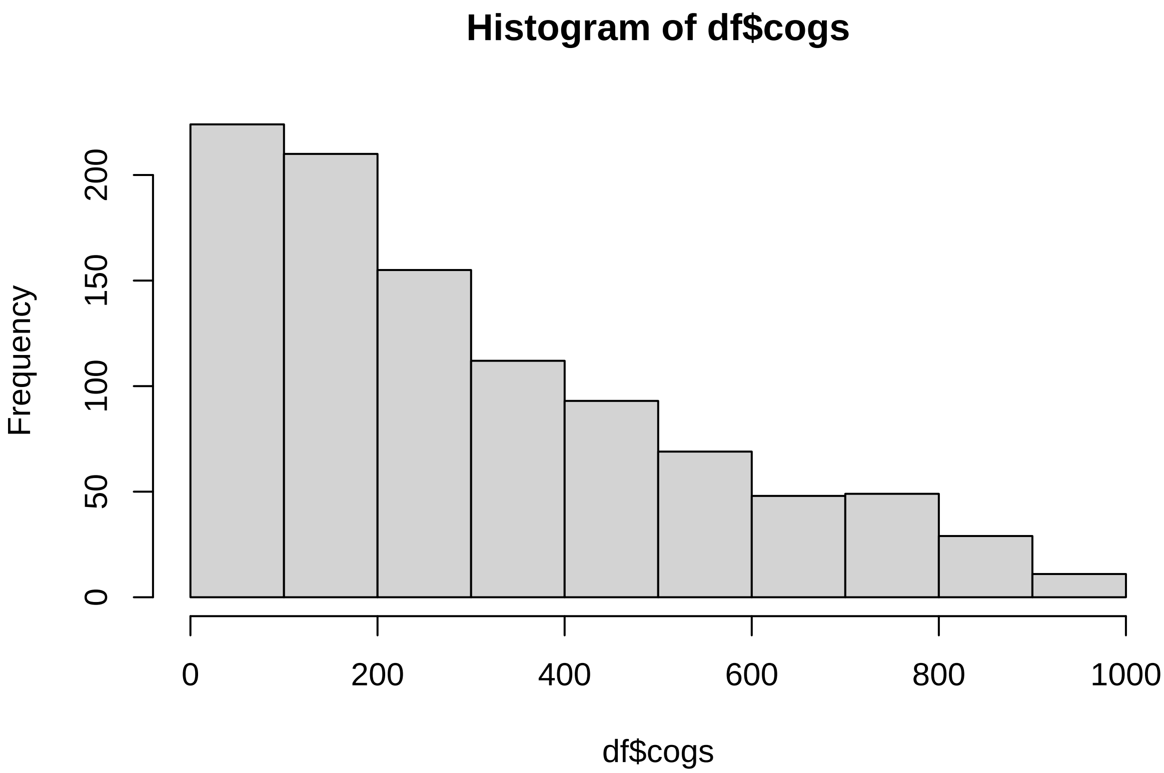
The quantity purchased is normally distributed. The quantity of the product that was most purchased by the customers is between o to 2 followed by 9 to 10.

```
#The histogram of Tax
hist(df$Tax)
```



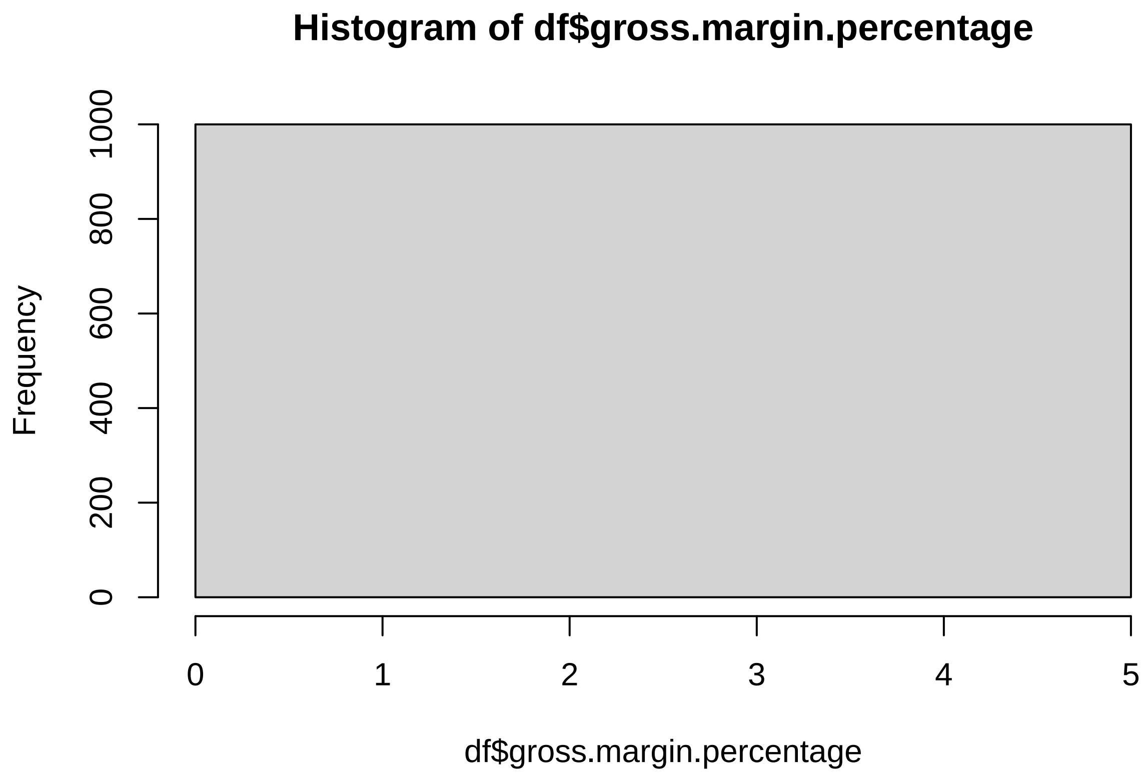
The tax that was is skewed to the right. The tax that was paid by most customers is 0 0 to 10

```r
#The histogram of cogs
hist(df$cogs)
```



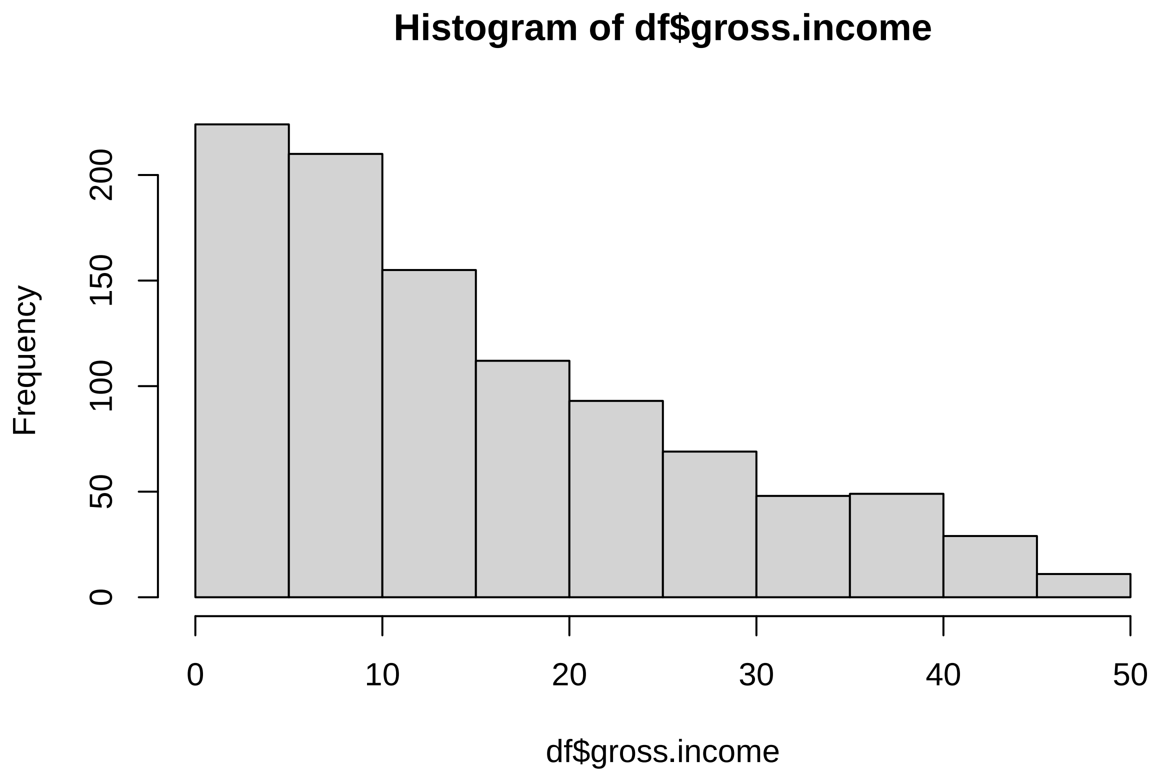
The cogs that was is skewed to the right. The cogs of most customers is between 0 to 200

```
#The histogram of gross.margin.percentage
hist(df$gross.margin.percentage)
```



The gross margin was constant across all the products.

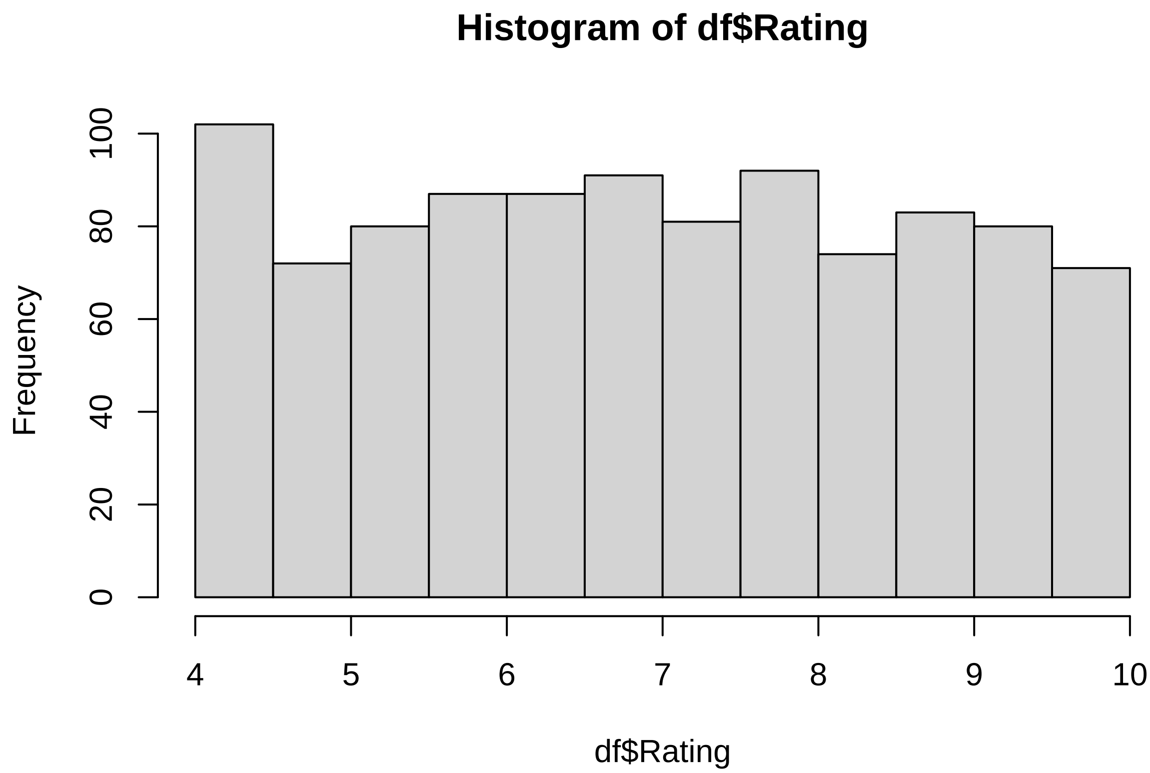
```
#The histogram of gross.income
hist(df$gross.income)
```



1.The gross income is skewed to the right

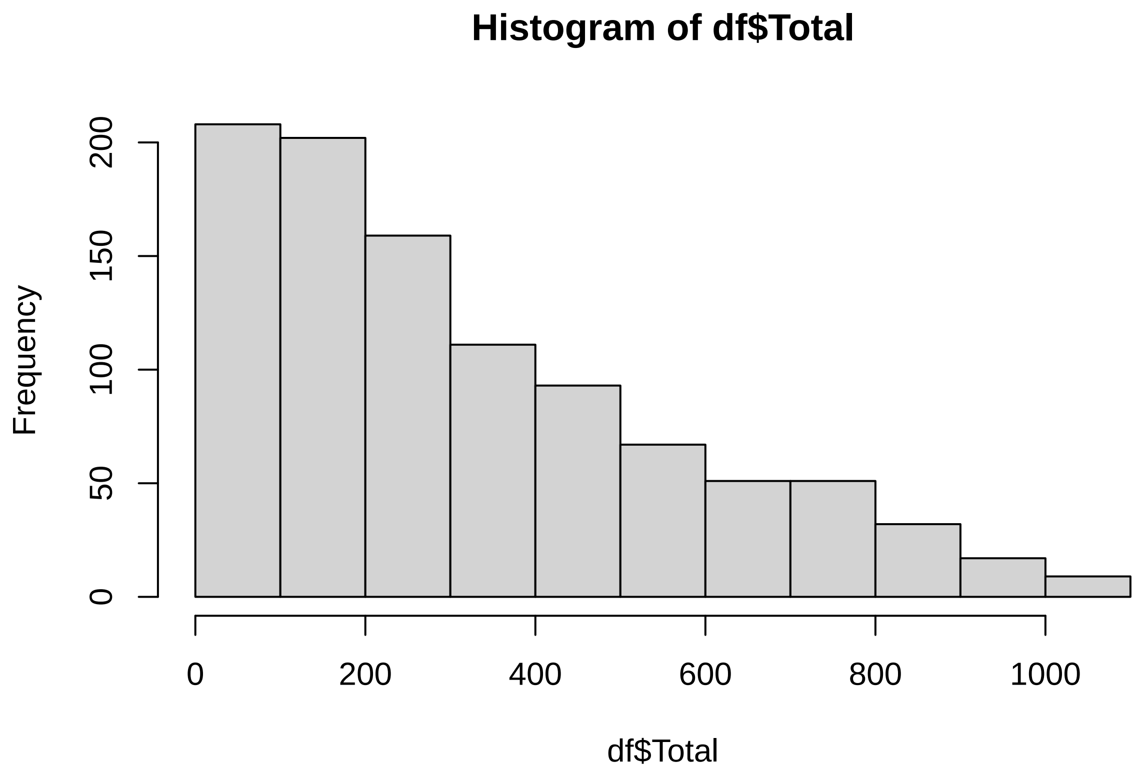
2.Most gross income is between 0 to 10.

```
#The histogram of Rating
hist(df$Rating)
```



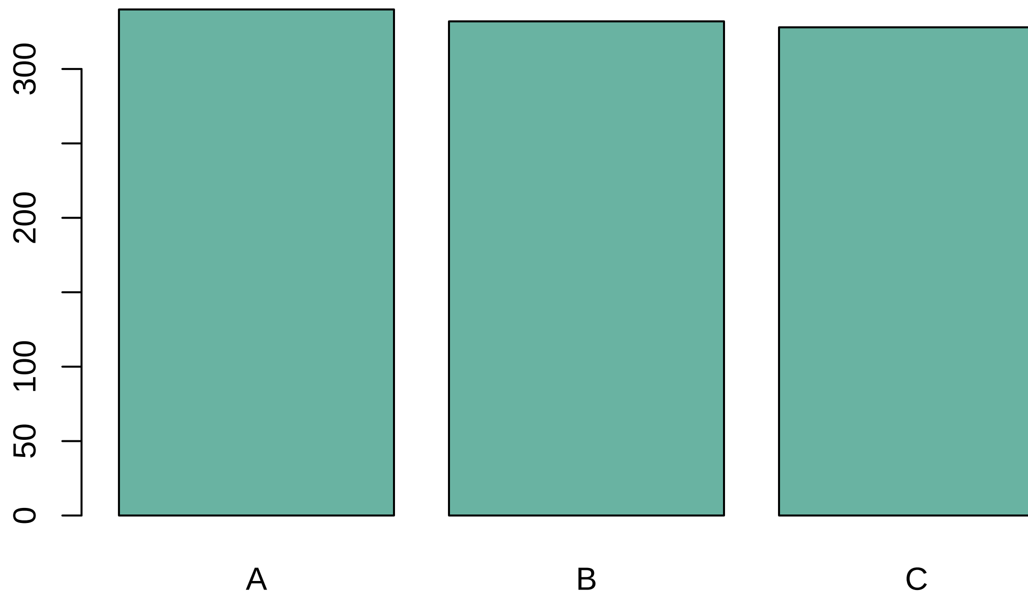
1. The ratings is normally distributed.

```
#The histogram  of Total
hist(df$Total)
```



The total is skewed to the right.

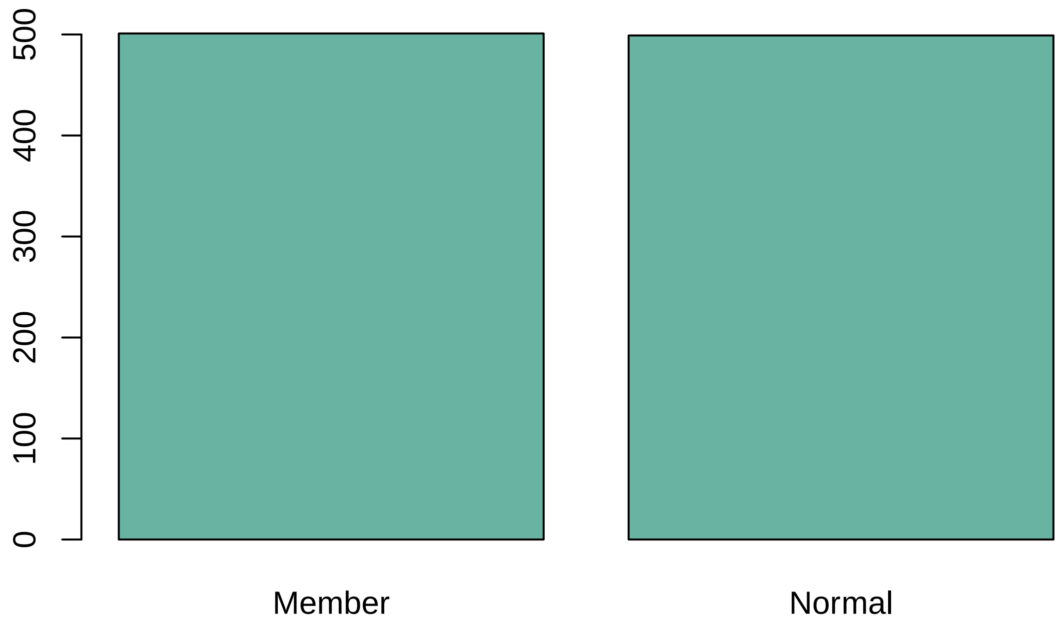
```
#Bar plot of Branch
freq <- table(df$Branch)
barplot(height=freq, names = df$name, col = "#69b3a2")
```



(ii) categorical analysis.

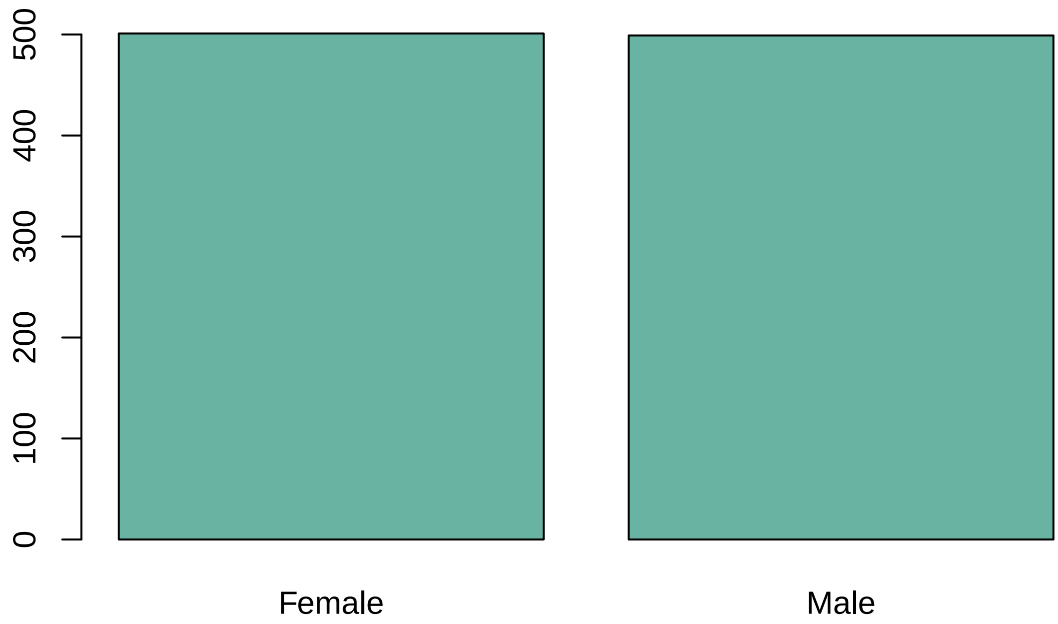
All the branches had the same branches has the same number of records in the dataset.

```r
#Bar plot of Customer.type
freq <- table(df$Customer.type)
barplot(height=freq, names = df$name, col = "#69b3a2")
```



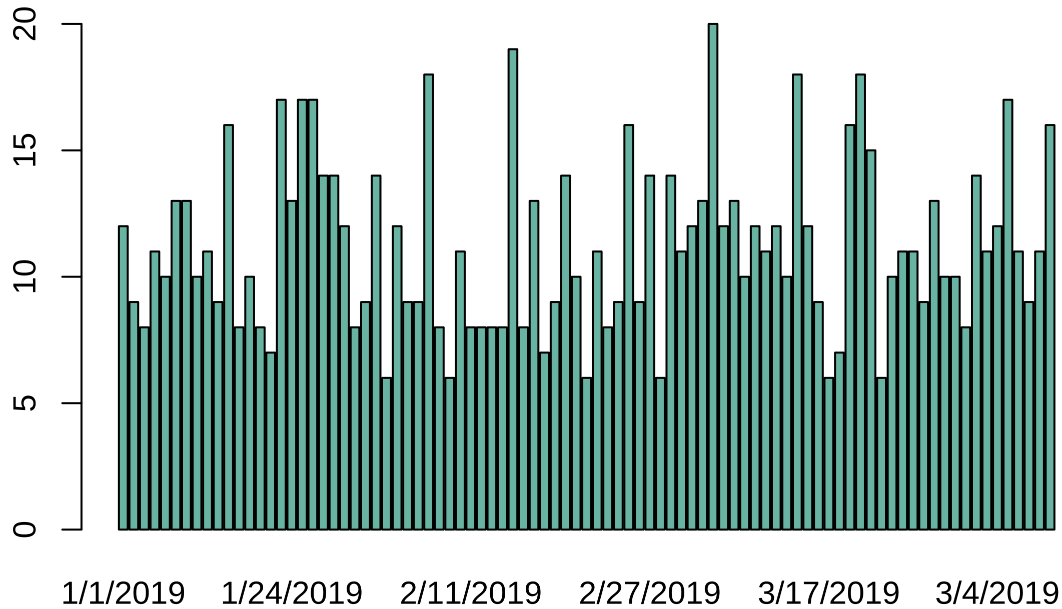
In the dataset the customer types are of member and normal have the same records.

```r
#Bar plot of Gender
freq <- table(df$Gender)
barplot(height=freq, names = df$name, col = "#69b3a2")
```

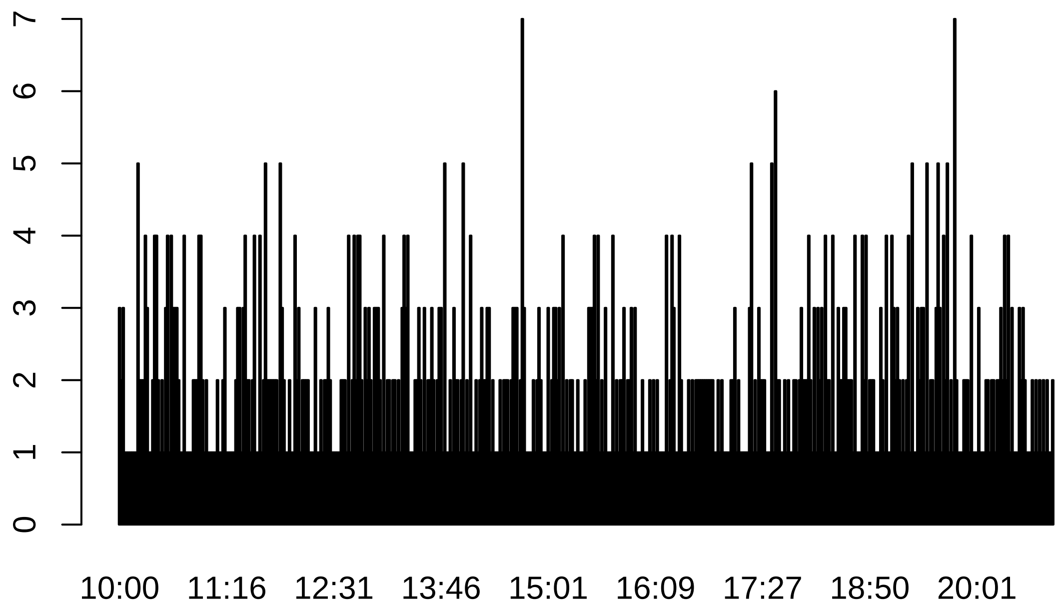


Both genders has the same number of records in the dataset.

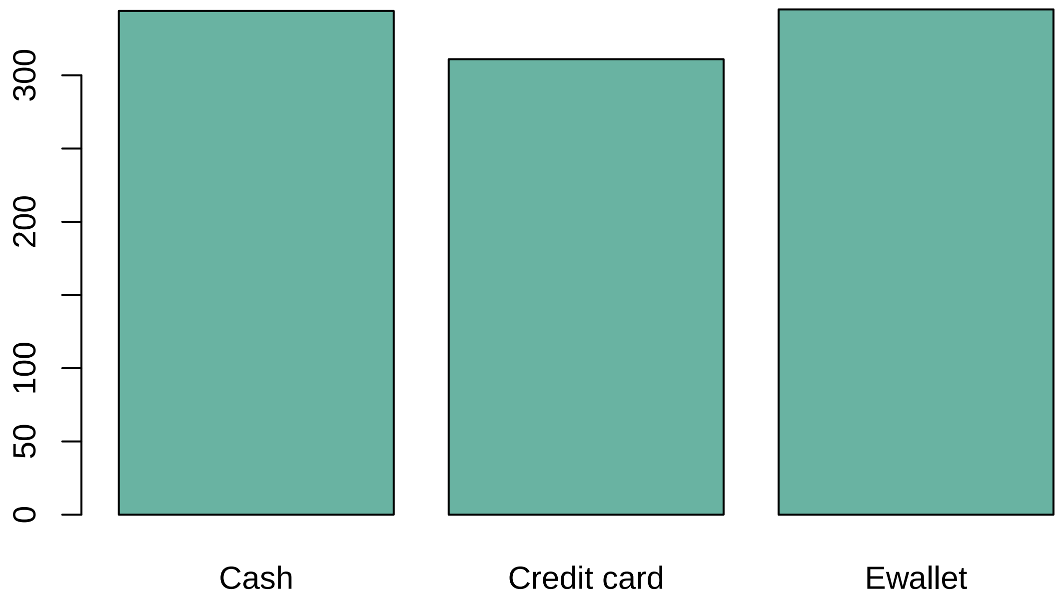
```
#Bar plot of Date
freq <- table(df$Date)
barplot(height=freq, names = df$name, col = "#69b3a2")
```




```r
#Bar plot of Time
freq <- table(df$Time)
barplot(height=freq, names = df$name, col = "#69b3a2")
```



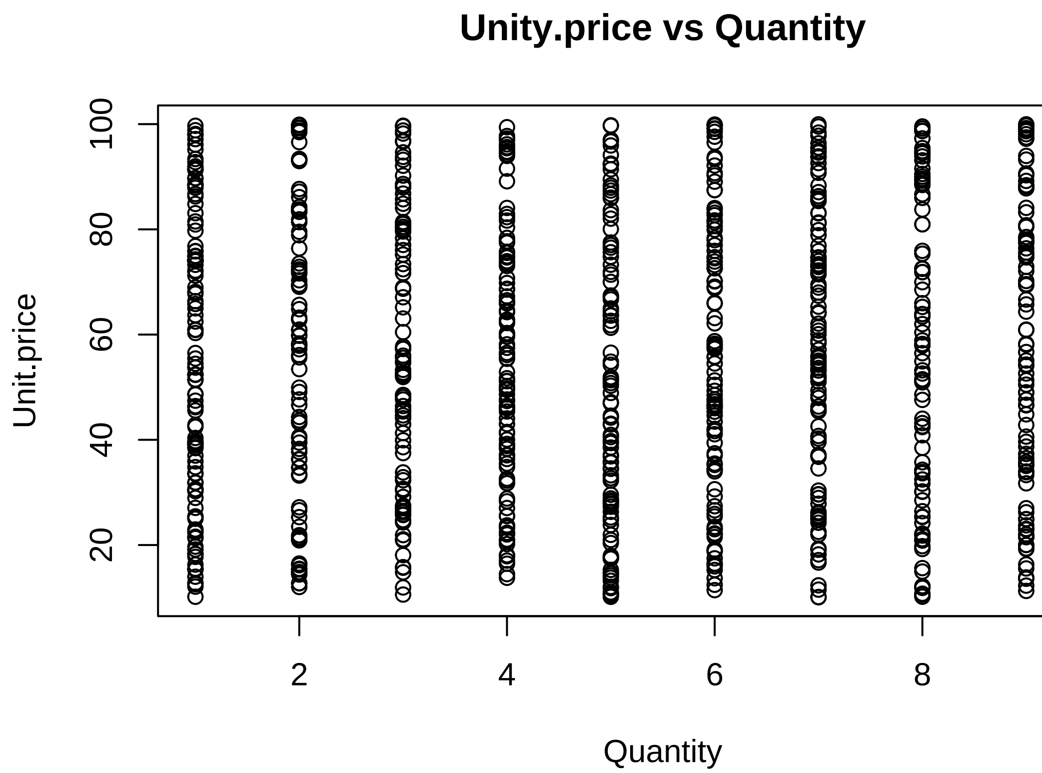

```
#Bar plot of Payment
freq <- table(df$Payment)
barplot(height=freq, names = df$name, col = "#69b3a2")
```



All three payment method is preferred by the customers but they slightly prefer the method of cash and Wallet.

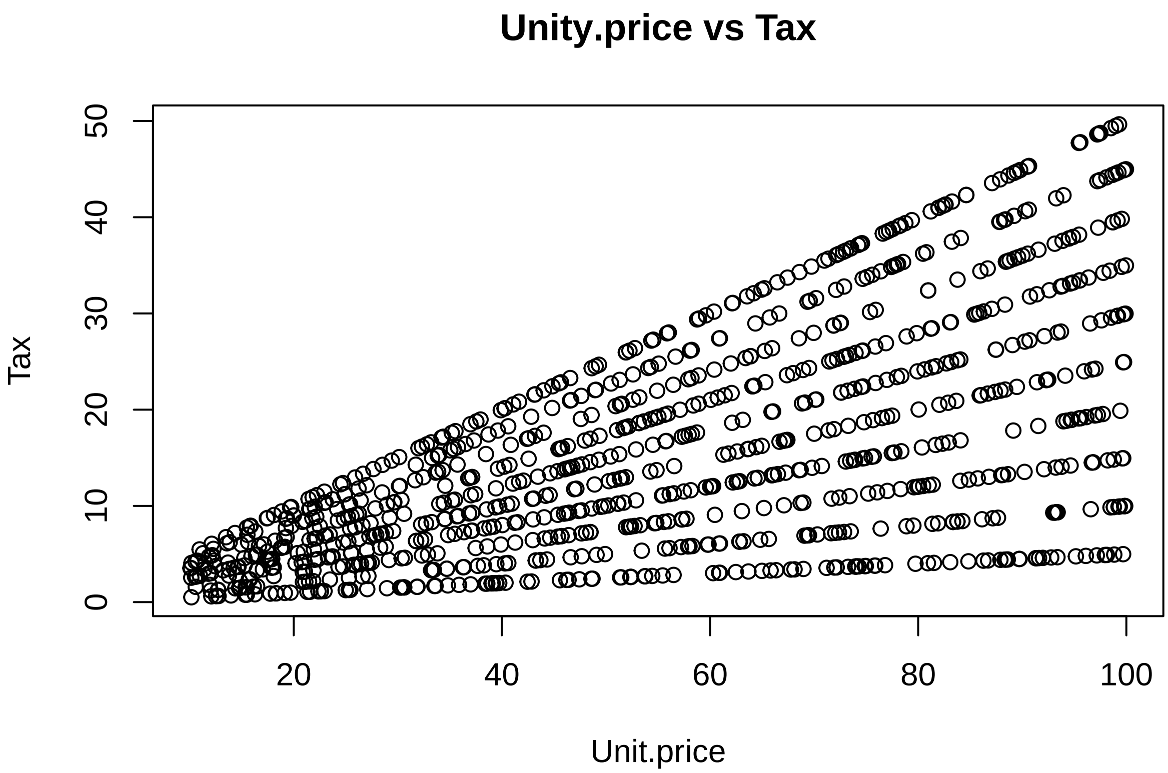
b). Bivariate analysis.

```r
#Scatter plot of Unity.price vs Quantity
plot(df$Quantity, df$Unit.price, xlab = ("Quantity"),
     ylab = ("Unit.price"), main = "Unity.price vs Quantity")
```



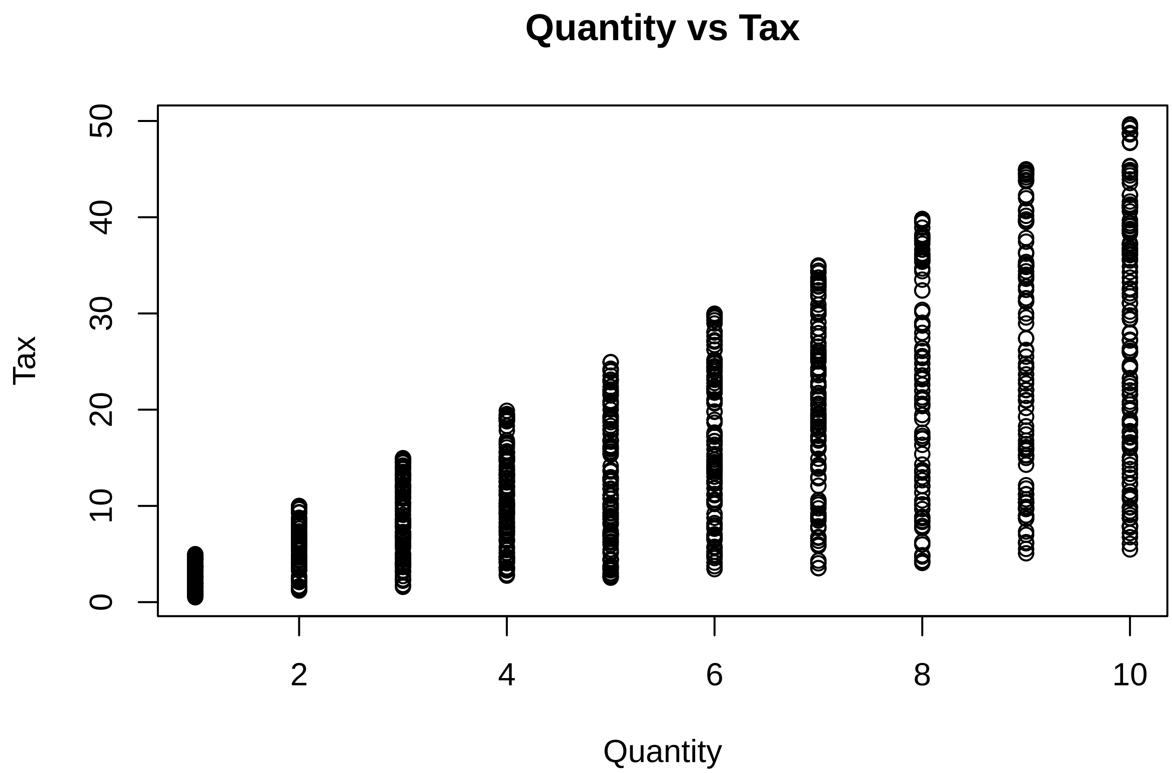
(i). Numerical vs Numerical
There's no correlation between quantity and unit,price

```
#Scatter plot of Unity.price vs Tax
plot(df$Unit.price, df$Tax, xlab = ("Unit.price"),
     ylab = ("Tax"), main = "Unity.price vs Tax")
```



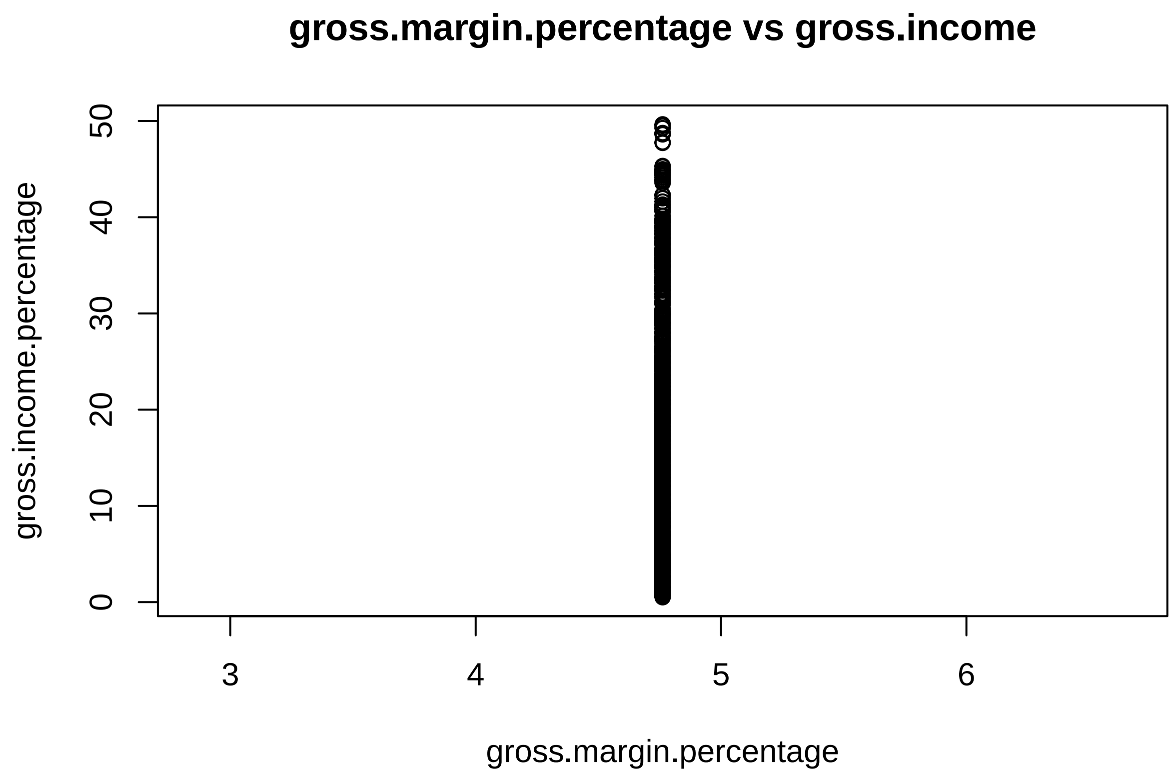
The correlation between unit price and tax is a positive correlation. The tax increases with the increase in unit price.

```
#Scatter plot of Quantity vs Tax
plot(df$Quantity, df$Tax, xlab = ("Quantity"),
     ylab = ("Tax"), main = "Quantity vs Tax")
```



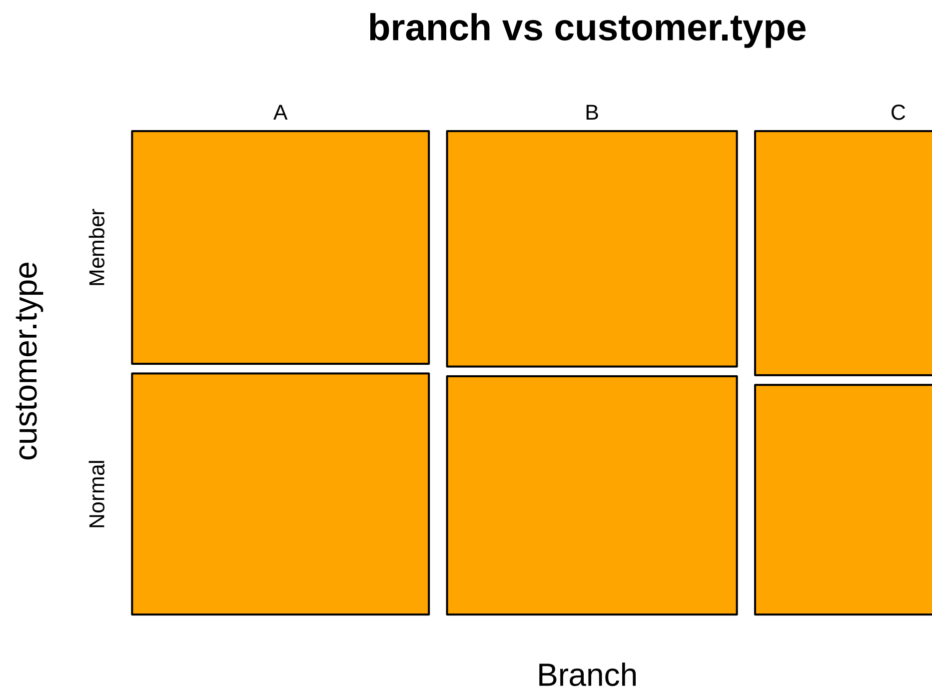
The quantity and tax have a positive correlation. The tax increases as the quantity increases.

```
#Scatter plot of Gross.income vs Gross.margin.income
plot(df$gross.margin.percentage, df$gross.income, xlab = ("gross.margin.percentage"),
     ylab = ("gross.income.percentage"), main = "gross.margin.percentage vs gross.income")
```



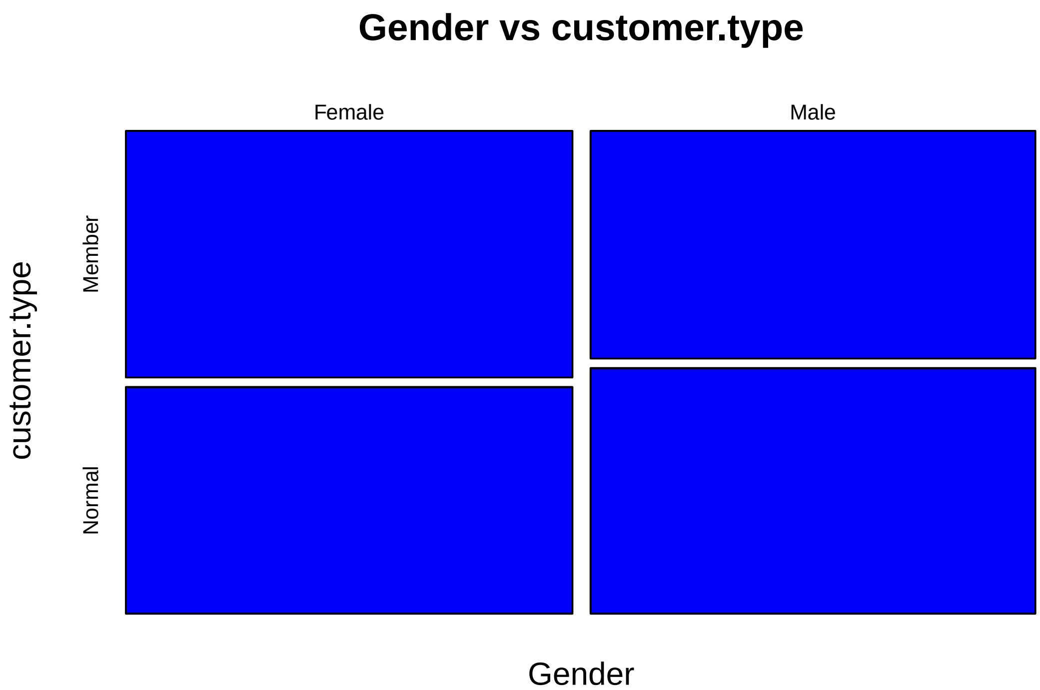
There is no correlation between the gross.margin.percentage and the gross income percentage.

```r
#Mosaic plot of branch vs customer.type
counts <- table(df$Branch, df$Customer.type)
#Create a mosaic plot
mosaicplot(counts, xlab="Branch", ylab="customer.type", main= "branch vs customer.type",
           col ="orange")
```



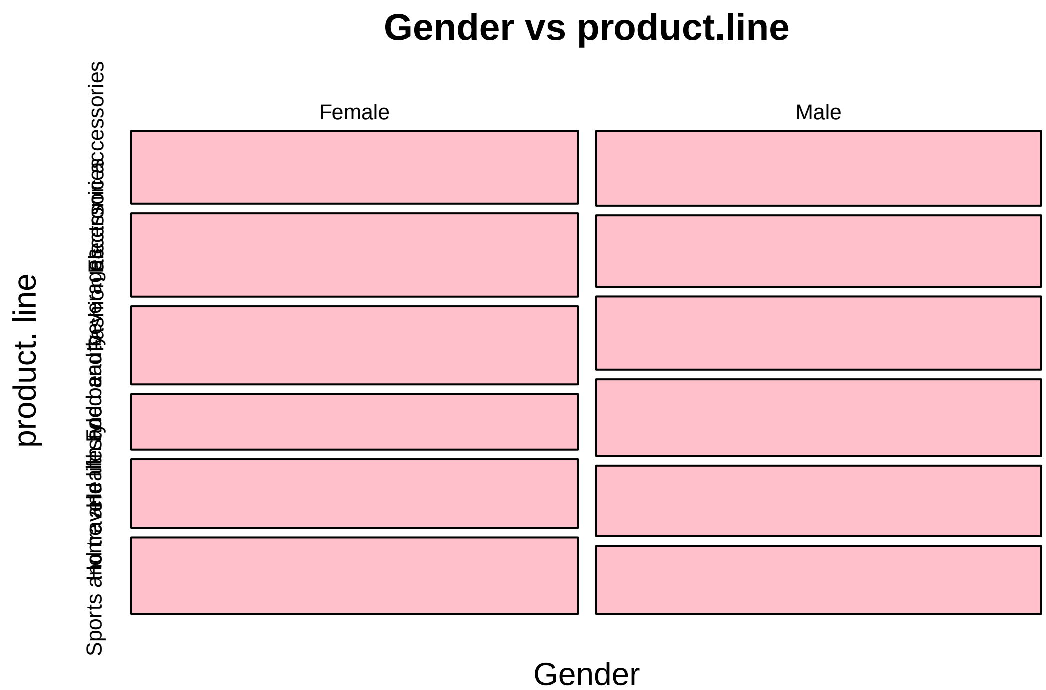
(ii). Categorical vs Categorical
All the branches have the same customer type.

```
#Mosaic plot of Gender vs customer.type
counts <- table(df$Gender, df$Customer.type)
#Create a mosaic plot
mosaicplot(counts, xlab="Gender", ylab="customer.type", main= "Gender vs customer.type",
           col ="blue")
```



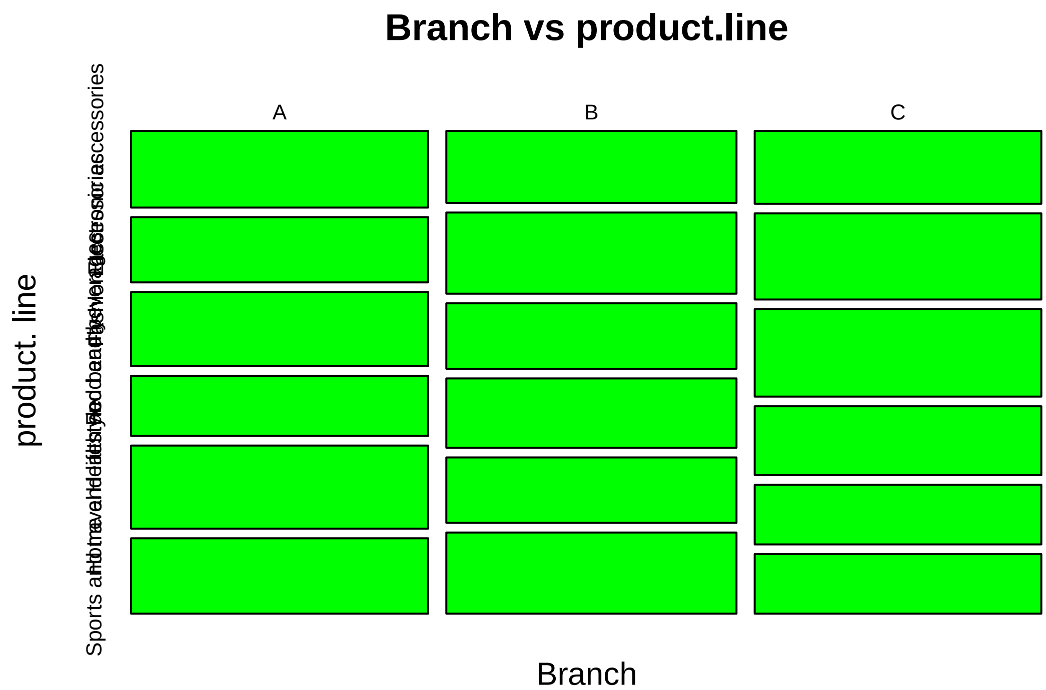
Most of the customer types who are members are female.

```
#Mosaic plot of Gender vs product. line
counts <- table(df$Gender, df$Product.line)
#Create a mosaic plot
mosaicplot(counts, xlab="Gender", ylab="product. line", main= "Gender vs product.line",
           col ="pink")
```



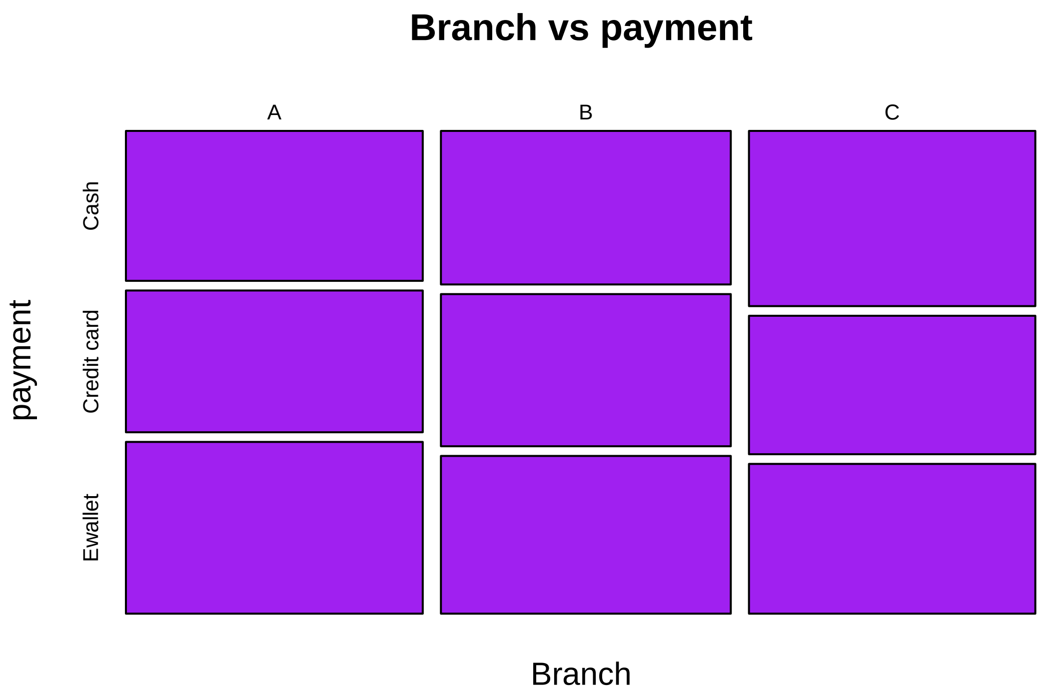
There was no particular product being favoured by certain gender.

```
#Mosaic plot of Branch vs product. line
counts <- table(df$Branch, df$Product.line)
#Create a mosaic plot
mosaicplot(counts, xlab="Branch", ylab="product. line", main= "Branch vs product.line",
           col ="green")
```



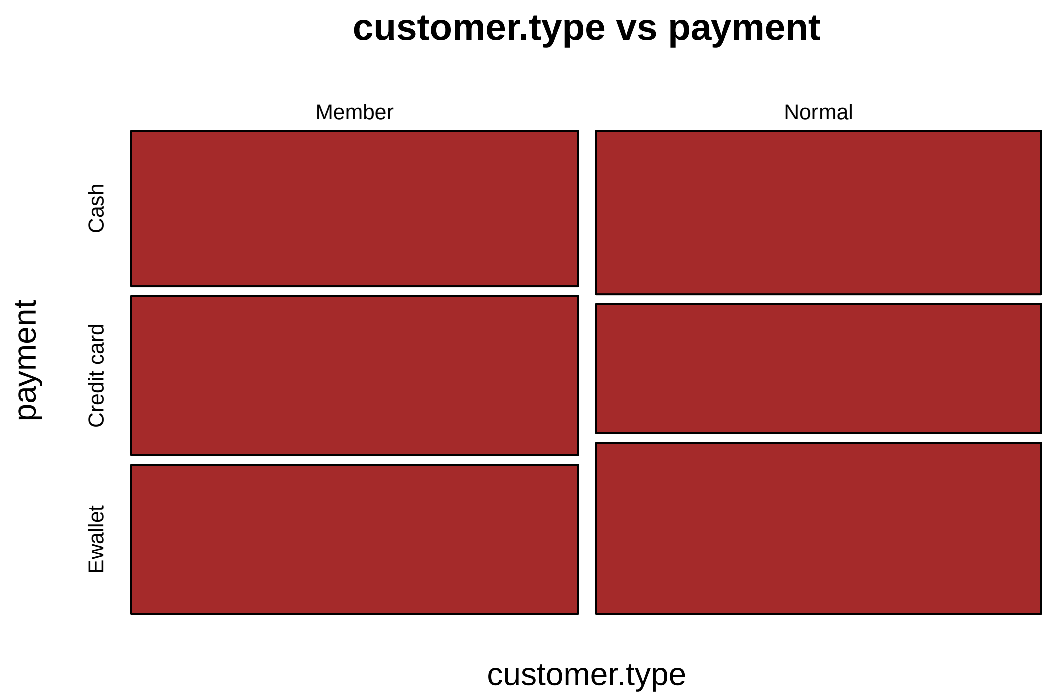
There was no specific product line being preferred by particular branch.

```r
#Mosaic plot of Branch vs payment
counts <- table(df$Branch, df$Payment)
#Create a mosaic plot
mosaicplot(counts, xlab="Branch", ylab="payment", main= "Branch vs payment",
           col ="purple")
```



There was no particular payment type being favored in particular branch.

```
#Mosaic plot of customer.type vs payment
counts <- table(df$Customer.type, df$Payment)
#Create a mosaic plot
mosaicplot(counts, xlab="customer.type", ylab="payment", main= "customer.type vs payment",
           col ="brown")
```



The members customer type preferred the credit card payment The normal customers preferred the Wallet payment method

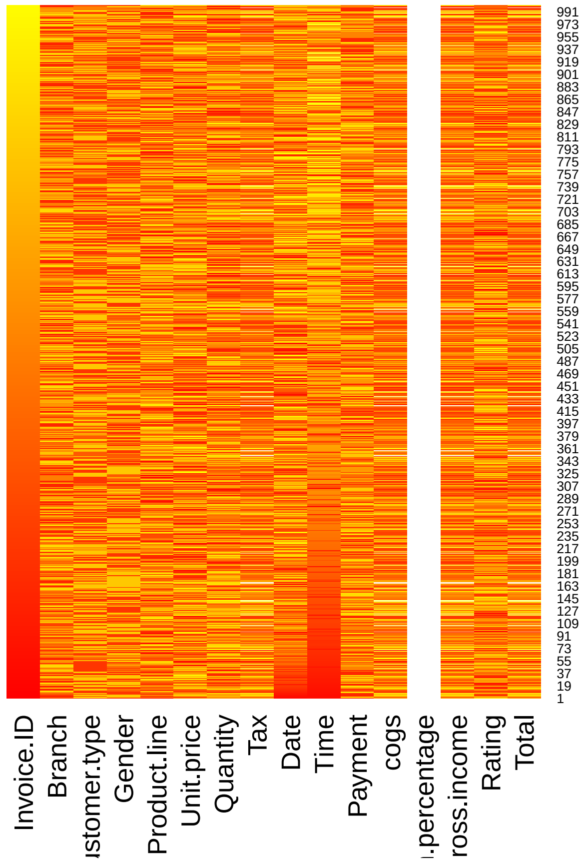
c) Multivariate analysis.

```
#Label encoding
#Load the library
library(superml)
```

```
## Loading required package: R6
```

```
label <- LabelEncoder$new()
#Label encode string columns to numerical
df$Branch <- label$fit_transform(df$Branch)
df$Gender <- label$fit_transform(df$Gender)
df$Customer.type <- label$fit_transform(df$Customer.type)
df$Product.line <- label$fit_transform(df$Product.line)
df$Date <- label$fit_transform(df$Date)
df$Time <- label$fit_transform(df$Time)
df$Payment <- label$fit_transform(df$Payment)
df$Invoice.ID <- label$fit_transform(df$Invoice.ID)
```

```r
#Convert the df to matrix
df_matrix <- as.matrix(df)
#Plot the heatmap of df_matrix
df_heatmap <- heatmap(df_matrix, Rowv=NA, Colv=NA, col = heat.colors(256), scale="column", margins=c(5,
```



6. Implement the solution

a). Dimensional reduction(t-SNE)

```r
#Curate the data for Analysis
#Load the Rtsne
library(Rtsne)
Customer.type <- df$Customer.type
df$Customer.type <- as.factor(df$Customer.type)
```

```r
#Colors for plotting
colors = rainbow(length(unique(df$Customer.type)))
names(colors) = unique(df$Customer.type)
```

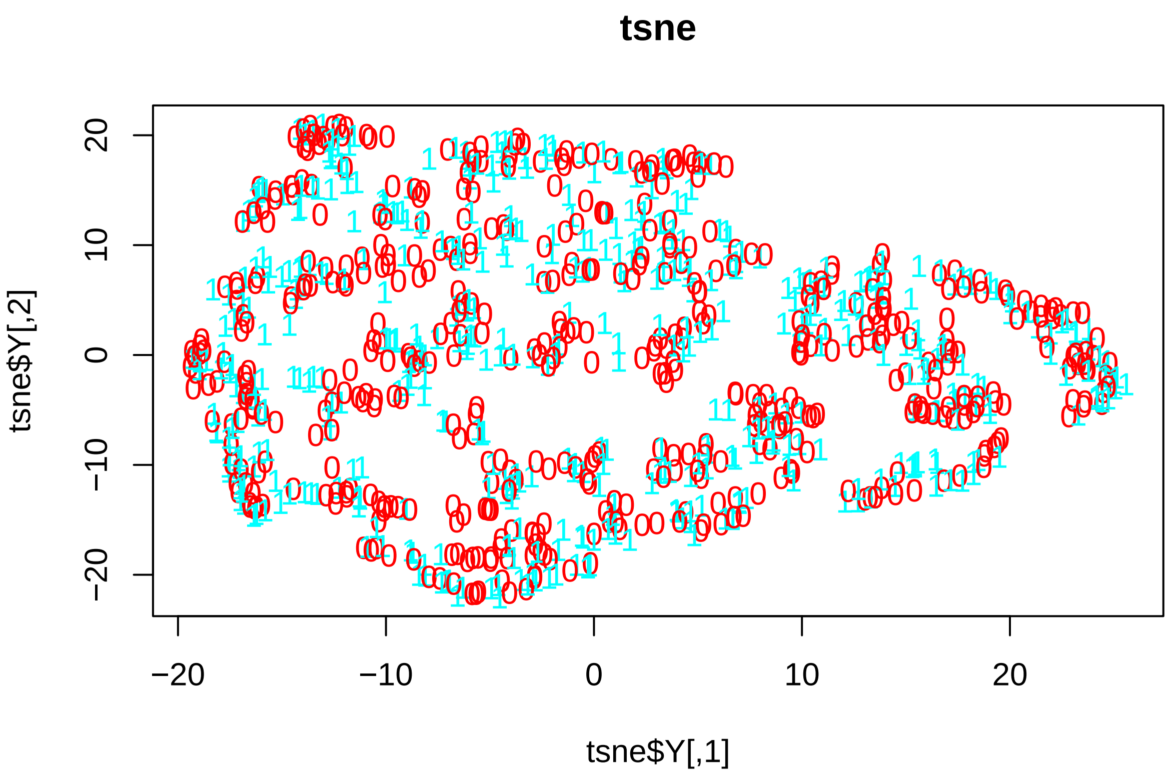
```r
#Executing the algorithm on curated dataset
tsne <- Rtsne(df[,-3], dims = 2, perplexity=30, verbose=TRUE, max_iter = 500)
```

```
## Performing PCA
## Read the 1000 x 15 data matrix successfully!
## OpenMP is working. 1 threads.
## Using no_dims = 2, perplexity = 30.000000, and theta = 0.500000
## Computing input similarities...
## Building tree...
## Done in 0.20 seconds (sparsity = 0.107394)!
## Learning embedding...
## Iteration 50: error is 66.408551 (50 iterations in 0.16 seconds)
## Iteration 100: error is 60.976055 (50 iterations in 0.11 seconds)
## Iteration 150: error is 60.959430 (50 iterations in 0.11 seconds)
## Iteration 200: error is 60.959704 (50 iterations in 0.10 seconds)
## Iteration 250: error is 60.959149 (50 iterations in 0.10 seconds)
## Iteration 300: error is 0.863882 (50 iterations in 0.11 seconds)
## Iteration 350: error is 0.734369 (50 iterations in 0.14 seconds)
## Iteration 400: error is 0.706861 (50 iterations in 0.12 seconds)
## Iteration 450: error is 0.691556 (50 iterations in 0.11 seconds)
## Iteration 500: error is 0.683921 (50 iterations in 0.11 seconds)
## Fitting performed in 1.17 seconds.
```

```r
# Getting the duration of execution
#
exeTimeTsne <- system.time(Rtsne(df[,-3], dims = 2, perplexity=30, verbose=TRUE, max_iter = 500))
```

```
## Performing PCA
## Read the 1000 x 15 data matrix successfully!
## OpenMP is working. 1 threads.
## Using no_dims = 2, perplexity = 30.000000, and theta = 0.500000
## Computing input similarities...
## Building tree...
## Done in 0.17 seconds (sparsity = 0.107394)!
## Learning embedding...
## Iteration 50: error is 69.487328 (50 iterations in 0.14 seconds)
## Iteration 100: error is 61.019299 (50 iterations in 0.11 seconds)
## Iteration 150: error is 60.990902 (50 iterations in 0.10 seconds)
## Iteration 200: error is 60.988891 (50 iterations in 0.11 seconds)
## Iteration 250: error is 60.986756 (50 iterations in 0.10 seconds)
## Iteration 300: error is 0.865779 (50 iterations in 0.11 seconds)
## Iteration 350: error is 0.738298 (50 iterations in 0.11 seconds)
## Iteration 400: error is 0.704246 (50 iterations in 0.11 seconds)
## Iteration 450: error is 0.691191 (50 iterations in 0.11 seconds)
## Iteration 500: error is 0.687236 (50 iterations in 0.11 seconds)
## Fitting performed in 1.12 seconds.
```

```r
#Plot the graph
plot(tsne$Y, t='n', main="tsne")
text(tsne$Y, labels=df$Customer.type, col=colors[df$Customer.type])
```



The t-SNE reduction model has classify the type of customers of carrefour form the high dimensional dataset into a low dimensional data. The customers have been as the member which is zero(red colors) and member which is "1"(light blue color).

b). Feature selections(Filter method)

```
#Load the two libraries
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(caret)
```

```
## Loading required package: ggplot2
```

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

```
#Calculate the correllationmatrix
correlationMatrix <- cor(df[, -3])
```

```
## Warning in cor(df[, -3]): the standard deviation is zero
```

```r
#Omit te n/a and Na in the function
#correlationMatrix <- na.omit(correlationMatrix)
```

```r
#Find attributes that are highly correllated
highlyCorrelated <- findCorrelation(correlationMatrix, cutoff = 0.75)

#features of highly correlated matrix
highlyCorrelated
```

```
## [1] 11 15  7
```

```r
names(df[,highlyCorrelated])
```

```
## [1] "Payment"  "Rating"   "Quantity"
```

```r
#Remove the redudant features
df_2 <- df[-highlyCorrelated]
#Preview df2
head(df_2, 3)
```

```
##   Invoice.ID Branch Customer.type Gender Product.line Unit.price     Tax Date
## 1          0      0             0      0            0      74.69 26.1415    0
## 2          1      1             1      0            1      15.28  3.8200    1
## 3          2      0             1      1            2      46.33 16.2155    2
##   Time   cogs gross.margin.percentage gross.income    Total
## 1    0 522.83                4.761905      26.1415 548.9715
## 2    1  76.40                4.761905       3.8200  80.2200
## 3    2 324.31                4.761905      16.2155 340.5255
```

Through the filter method we have successful filter out the the highly correlated features in the data sets. the features filtered are rating, payment, quantity, gross margin percentage.

6. Conclusions

1. Majority of the customers purchased item of quantity less than 2.
2. majority of the tax on products are less than 10
3. All the branches have equal number of customer types.
4. Most of the customers prefer cash mode of payment.
5. The price of the commodities increase with increase in tax.
6. The amount of tax increases with the increase in the quantity purchased.
7. The distribution of customers based on the customer type is heterogeneous.

7. Recomendations.

1. Introduce incentives such as free package bags to customers whom purchased more than ten items at ago.

2. product promotion that target each customers should done using the same channel since the distribution of the customers is heterogeneous.

3. liaise with the government institution such as KRA to reduce the tax on the commodity in order to lower the price of the commodity.

4. The same method pf product promotion should be carried out in all the branches since they have the same type of customers type.

8. Challenge the solution.

The model does not work if the dims parameter is hyper tuned. These makes it difficult to gauge the performance of the model at different dims parameter other than the stated one.

9. Follow up question.

a). Do we have the right data?

```
Yes, the data was appropriate
```

b). Do we need another data?

No, the data was appropriate.

c).Do we have the right question?

Yes, the question is clear and straight forward.