CarrefourMarketingAnalysis

Joy Machuka

9/10/2021

Problem Statement

Carrefour needs to increase their sales. They therefore get a data analyst to help them come up with marketing strategies to help them achieve this.

Defining the Question

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).

Metrics of Success

Our analysis will be considered successful when we are able to perform an efficient dimensionality reduction and feature selection and come up with good marketing strategies thereafter.

Context

Carrefour is one of the supermarkets in Kenya. They are ranked amongst the top shopping centres in Kenya hence they are always seeking to offer good customer service and in turn employ effective marketing strategies to increase their sales.

Experimental Design

Load the Data Checking the Data Data cleaning Exploratory Data Analysis Implement the Solution Challenge the Solution

Loading the data

```
carrefour <- read.csv("http://bit.ly/CarreFourDataset")
head(carrefour)
```

```
##    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
```

```
class(carrefour)
```

```
## [1] "data.frame"
```

```
dim(carrefour)
```

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

Our dataframe has 1000 rows and 16 columns.

```
names(carrefour)
```

```
##  [1] "Invoice.ID"              "Branch"
##  [3] "Customer.type"           "Gender"
##  [5] "Product.line"            "Unit.price"
##  [7] "Quantity"                "Tax"
##  [9] "Date"                    "Time"
## [11] "Payment"                 "cogs"
## [13] "gross.margin.percentage" "gross.income"
## [15] "Rating"                  "Total"
```

```
str(carrefour)
```

```
## '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 ...
```

We have 8 numerical columns and 8 char columns. We will convert the char categorical columns to factors for analysis.

Data Cleaning

```
#checking for nulls
anyNA(carrefour)
```

```
## [1] FALSE
```

We have no missing values.

```
#checking for duplicates
duplicated_rows <- carrefour[duplicated(carrefour),]
duplicated_rows
```

```
##  [1] Invoice.ID              Branch                  Customer.type
##  [4] Gender                  Product.line            Unit.price
##  [7] Quantity                Tax                     Date
## [10] Time                    Payment                 cogs
## [13] gross.margin.percentage gross.income            Rating
## [16] Total
## <0 rows> (or 0-length row.names)
```

We have no duplicate values

```
#change column names to lower case
lower <- function(x) {
    colnames(x) <- tolower(colnames(x))
    x
}
carrefour <- lower(carrefour)
names(carrefour)
```

```
##  [1] "invoice.id"             "branch"
##  [3] "customer.type"          "gender"
##  [5] "product.line"           "unit.price"
```

```
##  [7] "quantity"                "tax"
##  [9] "date"                    "time"
## [11] "payment"                 "cogs"
## [13] "gross.margin.percentage" "gross.income"
## [15] "rating"                  "total"
```

```
#drop the ID column
carrefour1 <- carrefour[,-1]
names(carrefour1)
```

```
##  [1] "branch"                  "customer.type"
##  [3] "gender"                  "product.line"
##  [5] "unit.price"              "quantity"
##  [7] "tax"                     "date"
##  [9] "time"                    "payment"
## [11] "cogs"                    "gross.margin.percentage"
## [13] "gross.income"            "rating"
## [15] "total"
```

```
dim(carrefour1)
```

```
## [1] 1000   15
```

New dataframe has now 15 columns 1000 rows

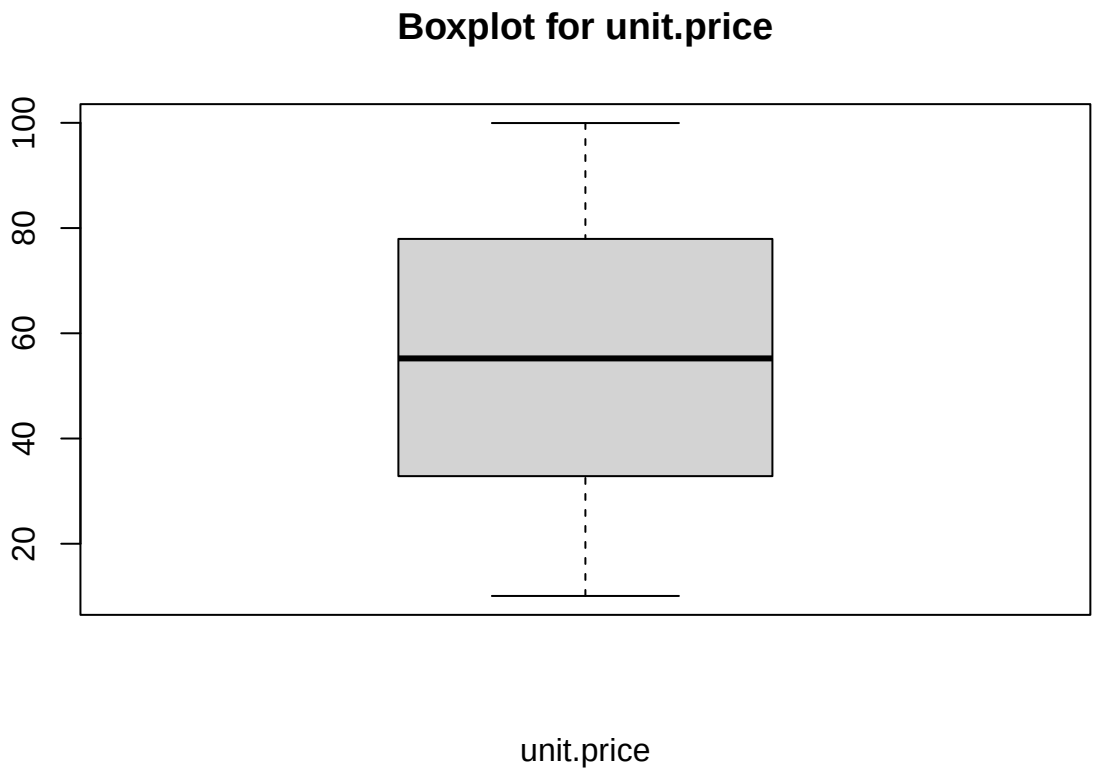
```
str(carrefour1)
```

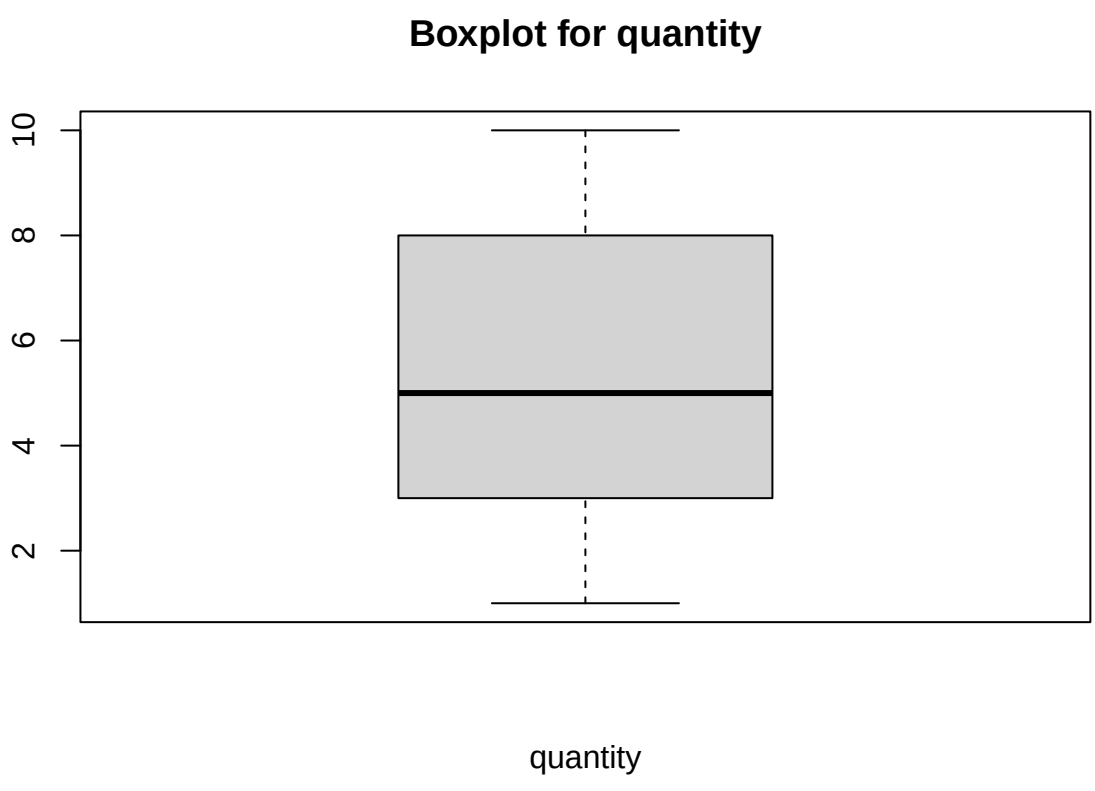
```
## 'data.frame':    1000 obs. of  15 variables:
##  $ branch                 : chr  "A" "C" "A" "A" ...
##  $ customer.type          : chr  "Member" "Normal" "Normal" "Member" ...
##  $ gender                 : chr  "Female" "Female" "Male" "Male" ...
##  $ product.line           : chr  "Health and 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 ...
```

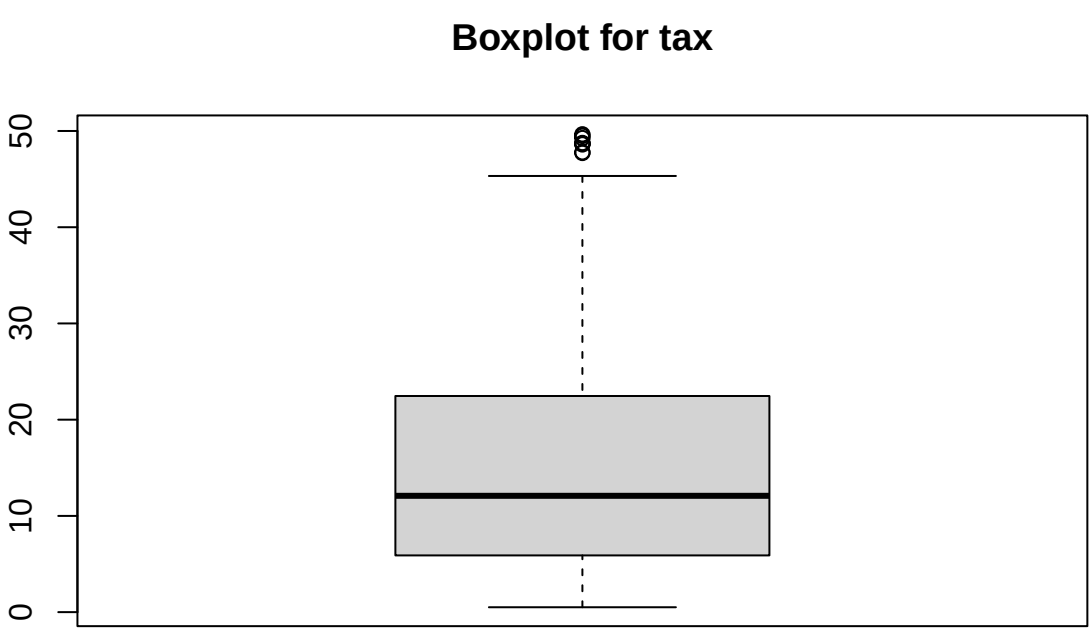
```
num_cols <- carrefour1[, c(5,6,7,11,13,14,15)]
names(num_cols)
```

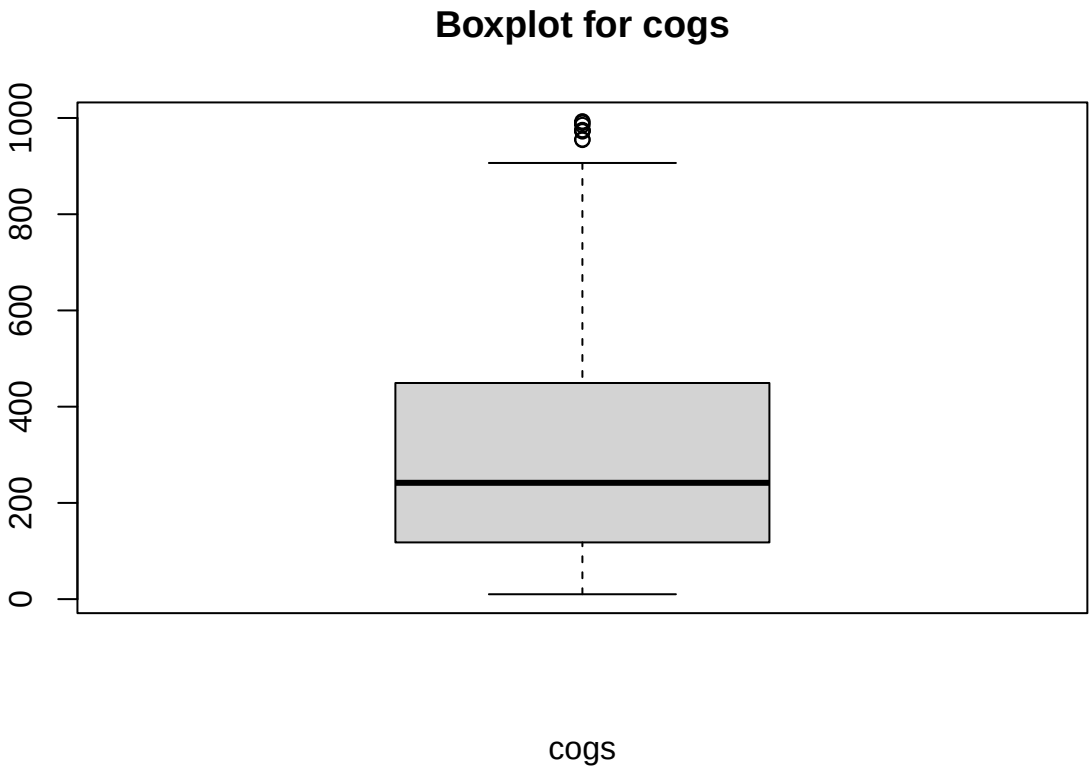
```
## [1] "unit.price"   "quantity"     "tax"          "cogs"         "gross.income"
## [6] "rating"       "total"
```

```
outliers = function(x){
  for(i in colnames(x)){
    boxplot(carrefour1[[i]], xlab=i, main=paste0("Boxplot for ",i))
  }
}
outliers(num_cols)
```

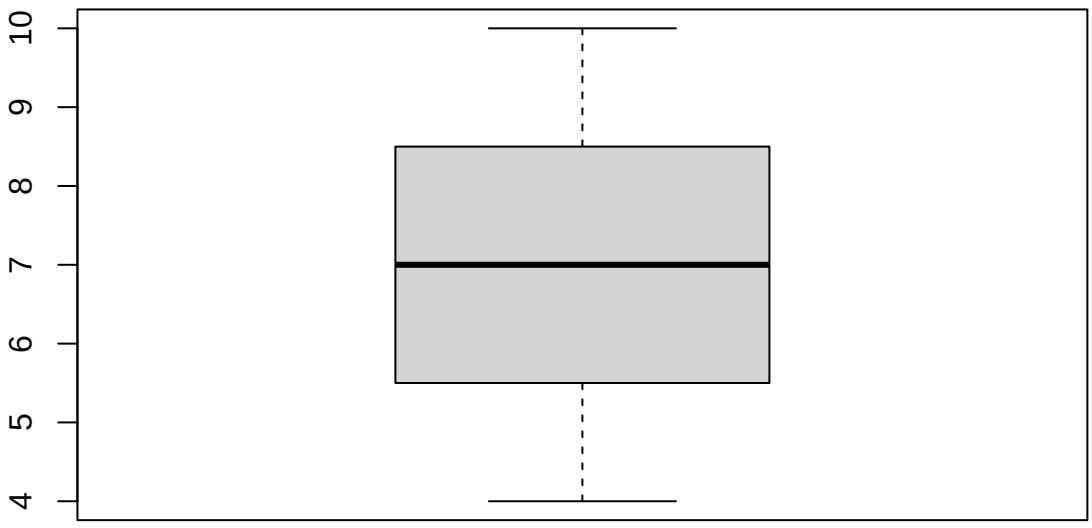


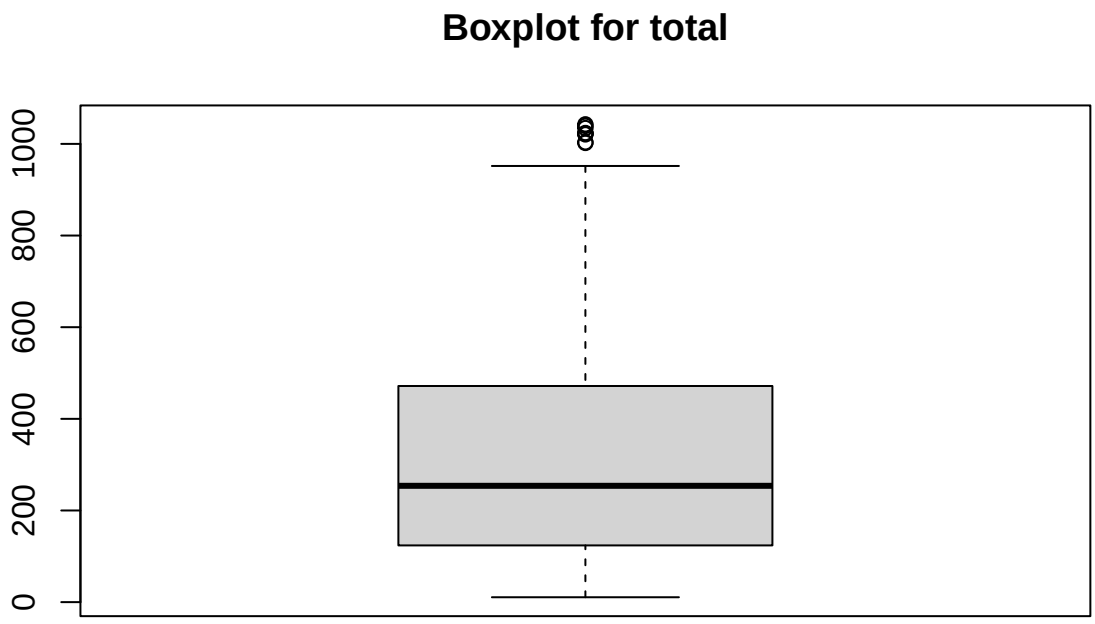

Boxplot for quantity
10
8
6
4
2
quantity


Boxplot for tax
0
10
20
30
40
50
tax


Boxplot for cogs
0
200
400
600
800
1000
cogs

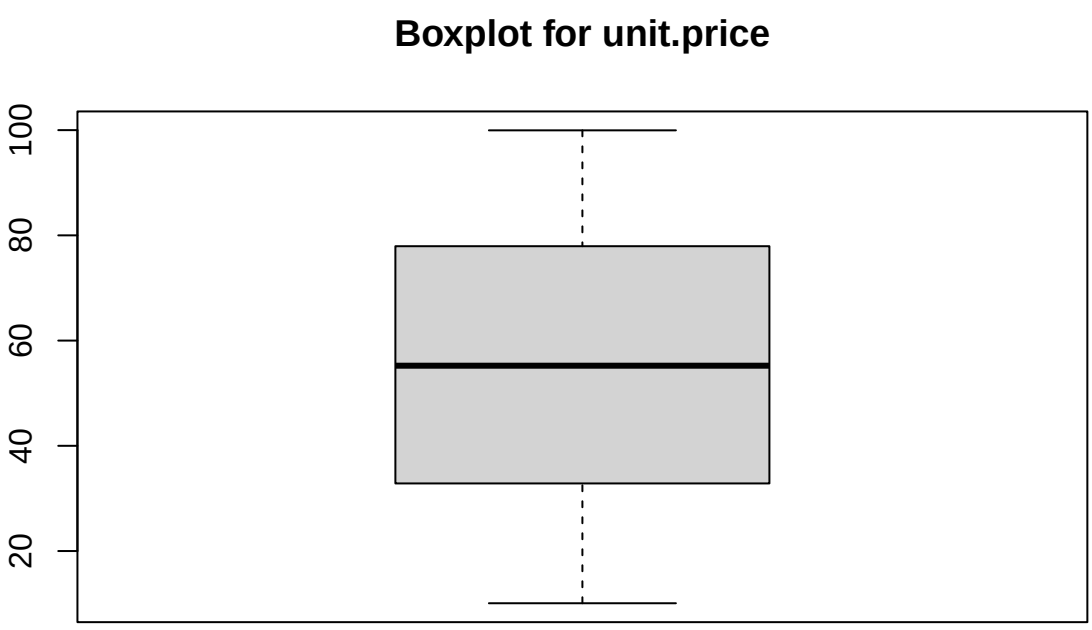

Boxplot for gross.income
0
10
20
30
40
50
gross.income

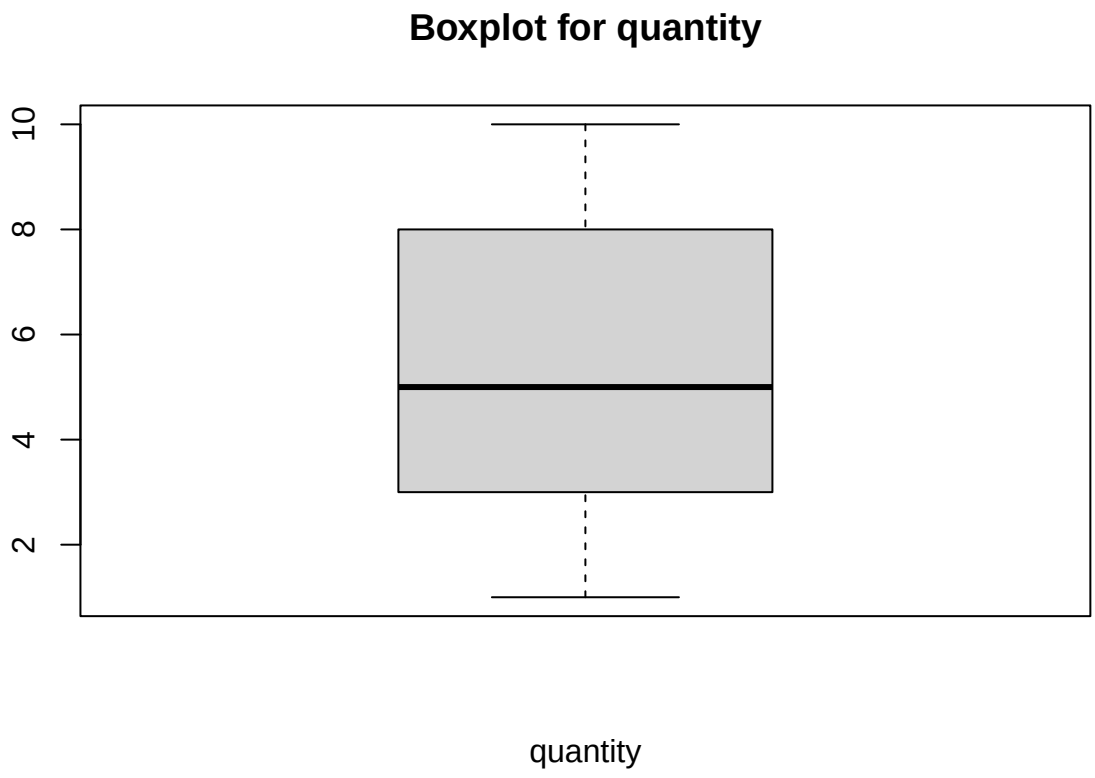

Boxplot for rating
10
9
8
7
6
5
4
rating

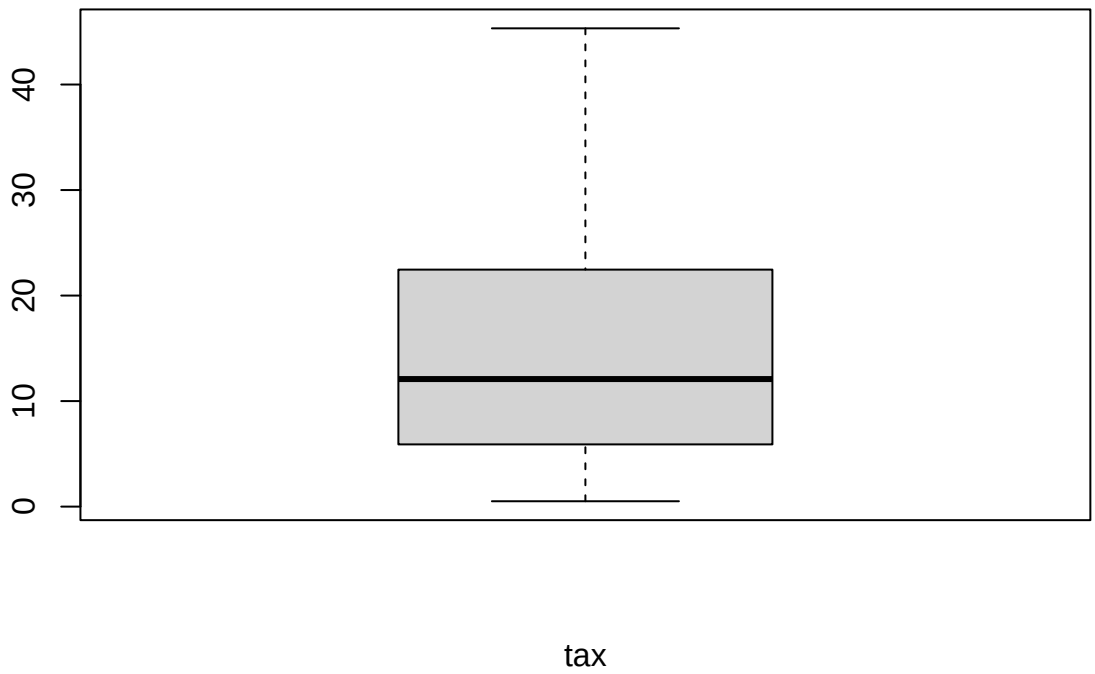


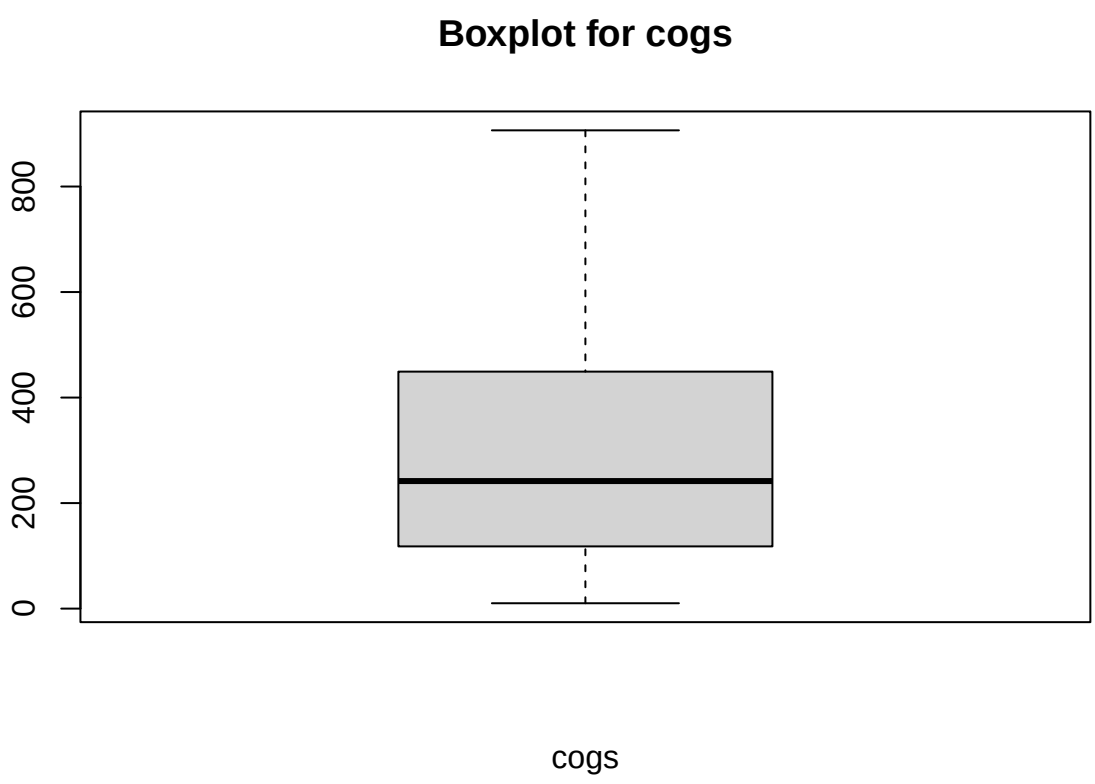

```
#replacing outliers with 5th and 95th percentile
outreplace <- function(x){
   qnt <- quantile(x, probs=c(.25, .75), na.rm = T)
   caps <- quantile(x, probs=c(.05, .95), na.rm = T)
   H <- 1.5 * IQR(x, na.rm = T)
   x[x < (qnt[1] - H)] <- caps[1]
   x[x > (qnt[2] + H)] <- caps[2]
   return(x)
}
carrefour1$tax <- outreplace(carrefour1$tax)
carrefour1$cogs <-outreplace(carrefour1$cogs)
carrefour1$gross.income <- outreplace(carrefour1$gross.income)
carrefour1$total <- outreplace(carrefour1$total )
```

```
outliers = function(x){
  for(i in colnames(x)){
    boxplot(carrefour1[[i]], xlab=i, main=paste0("Boxplot for ",i))
  }
}
outliers(num_cols)
```

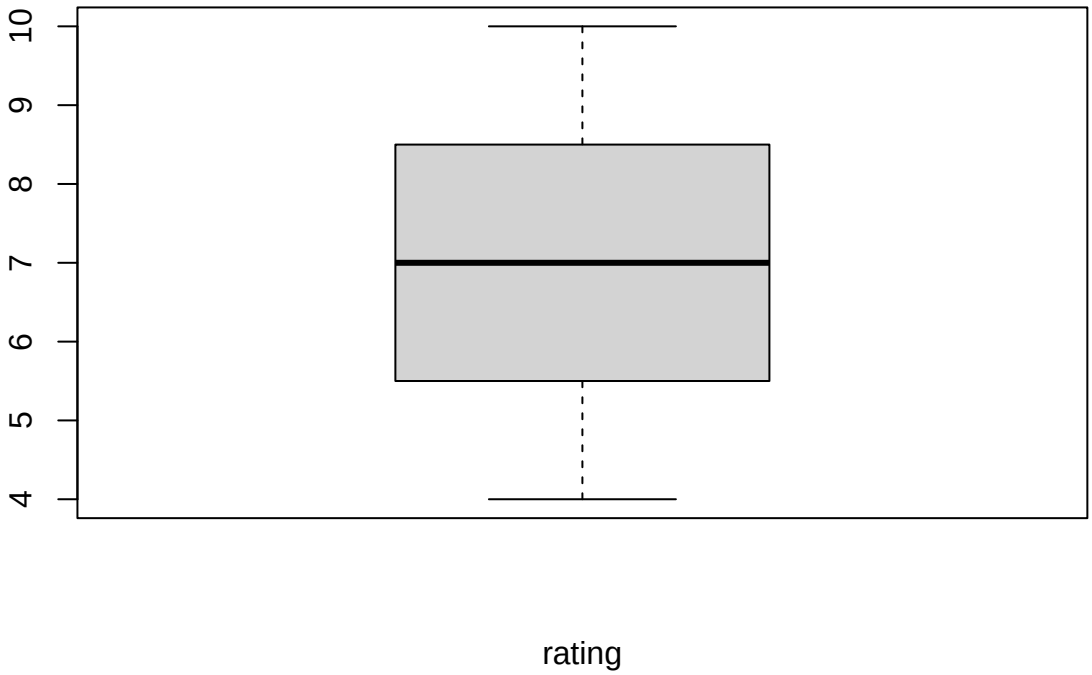

Boxplot for unit.price
100
80
60
40
20
unit.price

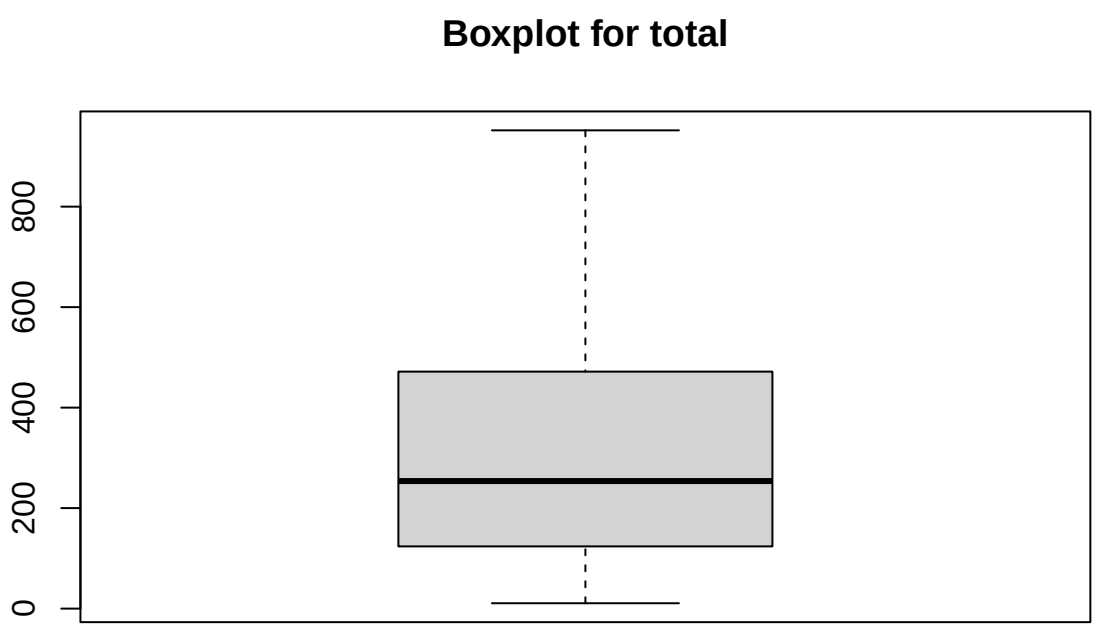

Boxplot for quantity
10
8
6
4
2
quantity


Boxplot for tax
0
10
20
30
40
tax


Boxplot for cogs
0
200
400
600
800
cogs


Boxplot for gross.income
0
10
20
30
40
gross.income


Boxplot for rating
10
9
8
7
6
5
4
rating



We have removed all outliers from the numerical columns

#Exploratory Data Analysis ##Univariate Analysis

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
carrefour1 <- mutate_at(carrefour1, vars(branch, customer.type,gender,product.line,payment), as.factor)
str(carrefour1)
```

```
## 'data.frame':    1000 obs. of  15 variables:
##  $ 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                   : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ time                   : chr  "13:08" "10:29" "13:23" "20:33" ...
##  $ 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 ...
```

```
#Summary statistics
summary(carrefour1)
```

```
##  branch  customer.type    gender                      product.line
##  A:340   Member:501    Female:501   Electronic accessories:170
##  B:332   Normal:499    Male  :499   Fashion accessories   :178
##  C:328                              Food and beverages    :174
##                                     Health and beauty     :152
##                                     Home and lifestyle    :160
##                                     Sports and travel     :166
##    unit.price       quantity          tax              date
##  Min.   :10.08   Min.   : 1.00   Min.   : 0.5085   Length:1000
##  1st Qu.:32.88   1st Qu.: 3.00   1st Qu.: 5.9249   Class :character
##  Median :55.23   Median : 5.00   Median :12.0880   Mode  :character
##  Mean   :55.67   Mean   : 5.51   Mean   :15.2932
##  3rd Qu.:77.94   3rd Qu.: 8.00   3rd Qu.:22.4453
##  Max.   :99.96   Max.   :10.00   Max.   :45.3250
##      time                 payment         cogs        gross.margin.percentage
##  Length:1000        Cash       :344   Min.   : 10.17   Min.   :4.762
##  Class :character   Credit card:311   1st Qu.:118.50   1st Qu.:4.762
##  Mode  :character   Ewallet    :345   Median :241.76   Median :4.762
##                                       Mean   :305.86   Mean   :4.762
##                                       3rd Qu.:448.90   3rd Qu.:4.762
##                                       Max.   :906.50   Max.   :4.762
##   gross.income         rating          total
##  Min.   : 0.5085   Min.   : 4.000   Min.   : 10.68
##  1st Qu.: 5.9249   1st Qu.: 5.500   1st Qu.:124.42
##  Median :12.0880   Median : 7.000   Median :253.85
##  Mean   :15.2932   Mean   : 6.973   Mean   :321.16
##  3rd Qu.:22.4453   3rd Qu.: 8.500   3rd Qu.:471.35
##  Max.   :45.3250   Max.   :10.000   Max.   :951.83
```

branch A has 340, branch 332 and branch C has 328 transactions. Females are 501 and 499 males 344 cash payment 311 credit cat payments and 345 using Ewallet There were 170 electronic accessories, 178 fashion accessories,174 food and beverages,152 health and beauty, 160 home and lifestyle and 166 sports and travel.

```
#mode function
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
```

```
#apply it on the duration columns
getmode(carrefour1$quantity)
```

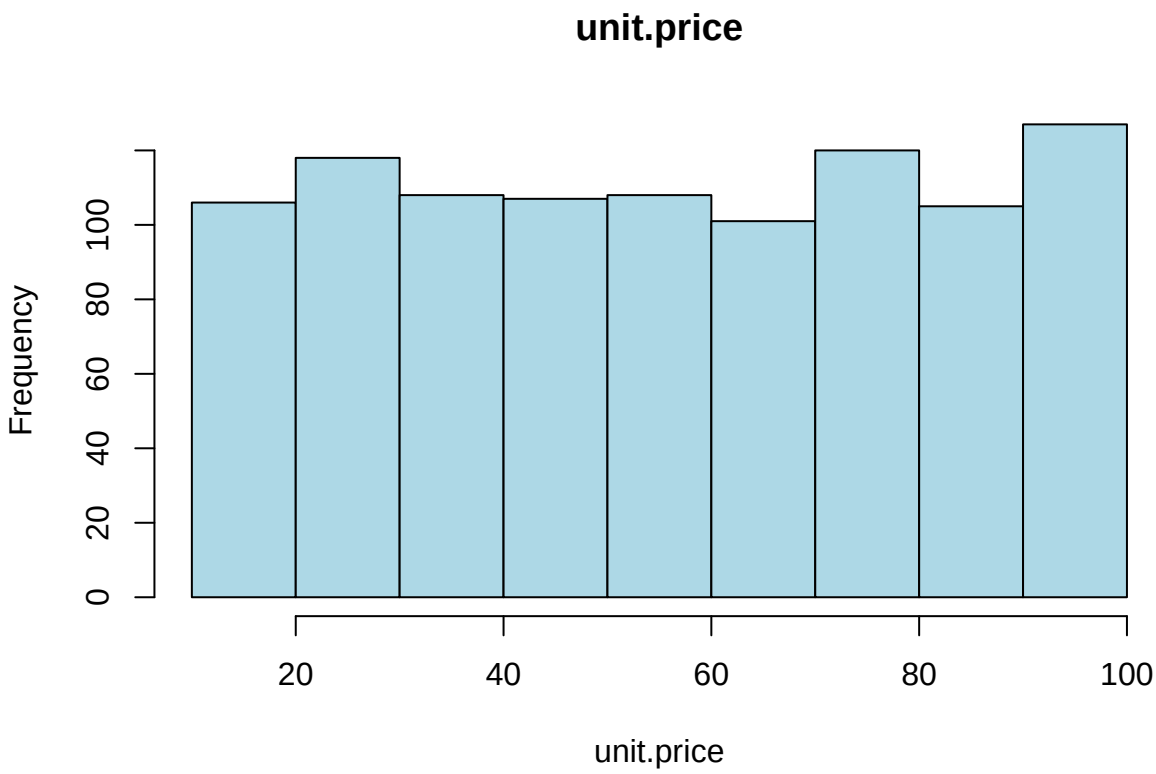
```
## [1] 10
```

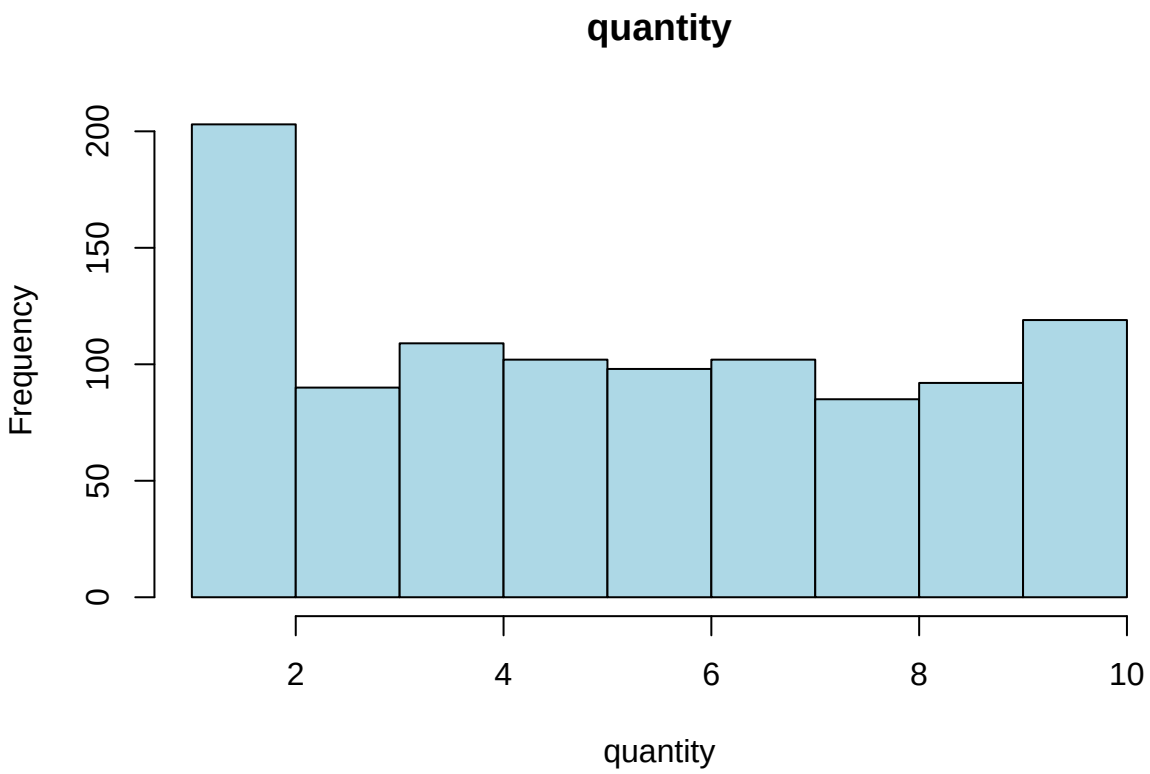
Most number of purchased items per invoice were 10

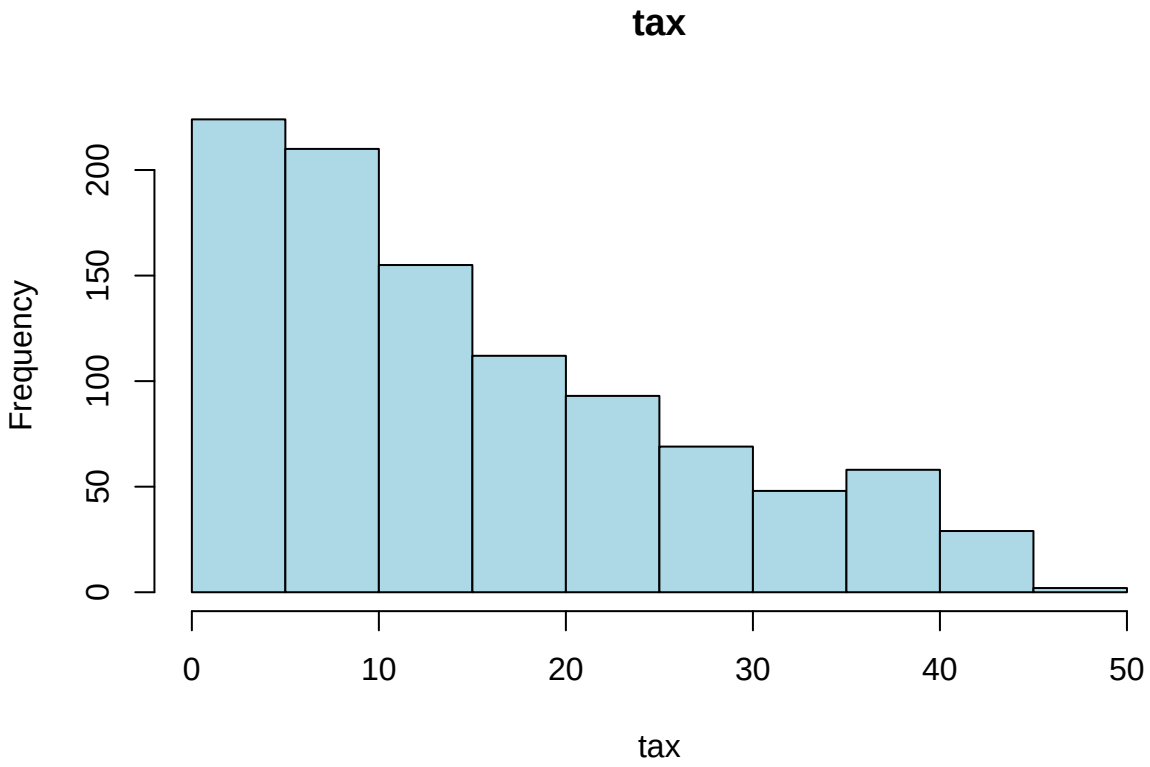
```
#descriptive statistics of the dataframe
psych::describe(carrefour1)
```

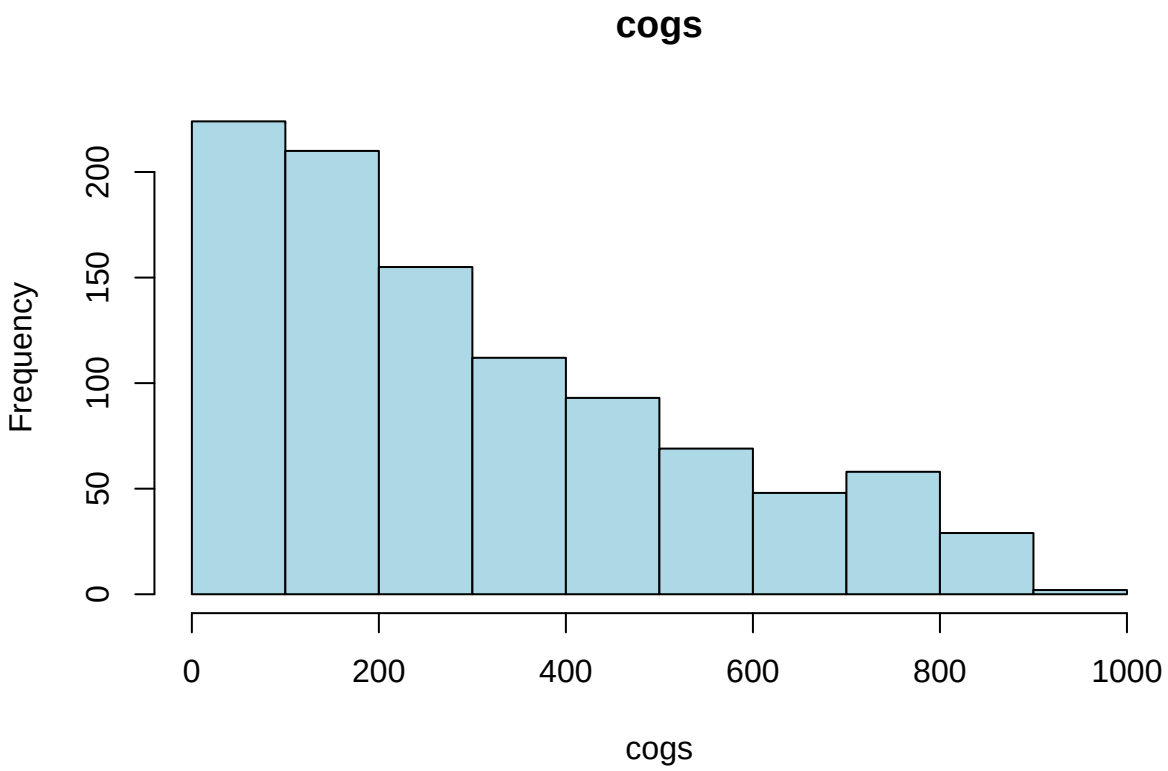
```
##                          vars    n   mean     sd median trimmed    mad   min
## branch*                     1 1000   1.99   0.82   2.00    1.99   1.48  1.00
## customer.type*              2 1000   1.50   0.50   1.00    1.50   0.00  1.00
## gender*                     3 1000   1.50   0.50   1.00    1.50   0.00  1.00
## product.line*               4 1000   3.45   1.72   3.00    3.44   1.48  1.00
## unit.price                  5 1000  55.67  26.49  55.23   55.62  33.37 10.08
## quantity                    6 1000   5.51   2.92   5.00    5.51   2.97  1.00
## tax                         7 1000  15.29  11.50  12.09   14.00  11.13  0.51
## date*                       8 1000  45.58  25.89  47.00   45.63  34.10  1.00
## time*                       9 1000 252.18 147.07 249.00  252.49 190.51  1.00
## payment*                   10 1000   2.00   0.83   2.00    2.00   1.48  1.00
## cogs                       11 1000 305.86 229.92 241.76  279.91 222.65 10.17
## gross.margin.percentage    12 1000   4.76   0.00   4.76    4.76   0.00  4.76
## gross.income               13 1000  15.29  11.50  12.09   14.00  11.13  0.51
## rating                     14 1000   6.97   1.72   7.00    6.97   2.22  4.00
## total                      15 1000 321.16 241.42 253.85  293.91 233.78 10.68
##                             max  range  skew kurtosis   se
## branch*                    3.00   2.00  0.02    -1.51 0.03
## customer.type*             2.00   1.00  0.00    -2.00 0.02
## gender*                    2.00   1.00  0.00    -2.00 0.02
## product.line*              6.00   5.00  0.06    -1.28 0.05
## unit.price                99.96  89.88  0.01    -1.22 0.84
## quantity                  10.00   9.00  0.01    -1.22 0.09
## tax                       45.33  44.82  0.82    -0.32 0.36
## date*                     89.00  88.00 -0.03    -1.23 0.82
## time*                    506.00 505.00  0.00    -1.25 4.65
## payment*                   3.00   2.00  0.00    -1.55 0.03
## cogs                     906.50 896.33  0.82    -0.32 7.27
## gross.margin.percentage    4.76   0.00   NaN      NaN 0.00
## gross.income              45.33  44.82  0.82    -0.32 0.36
## rating                    10.00   6.00  0.01    -1.16 0.05
## total                    951.82 941.15  0.82    -0.32 7.63
```

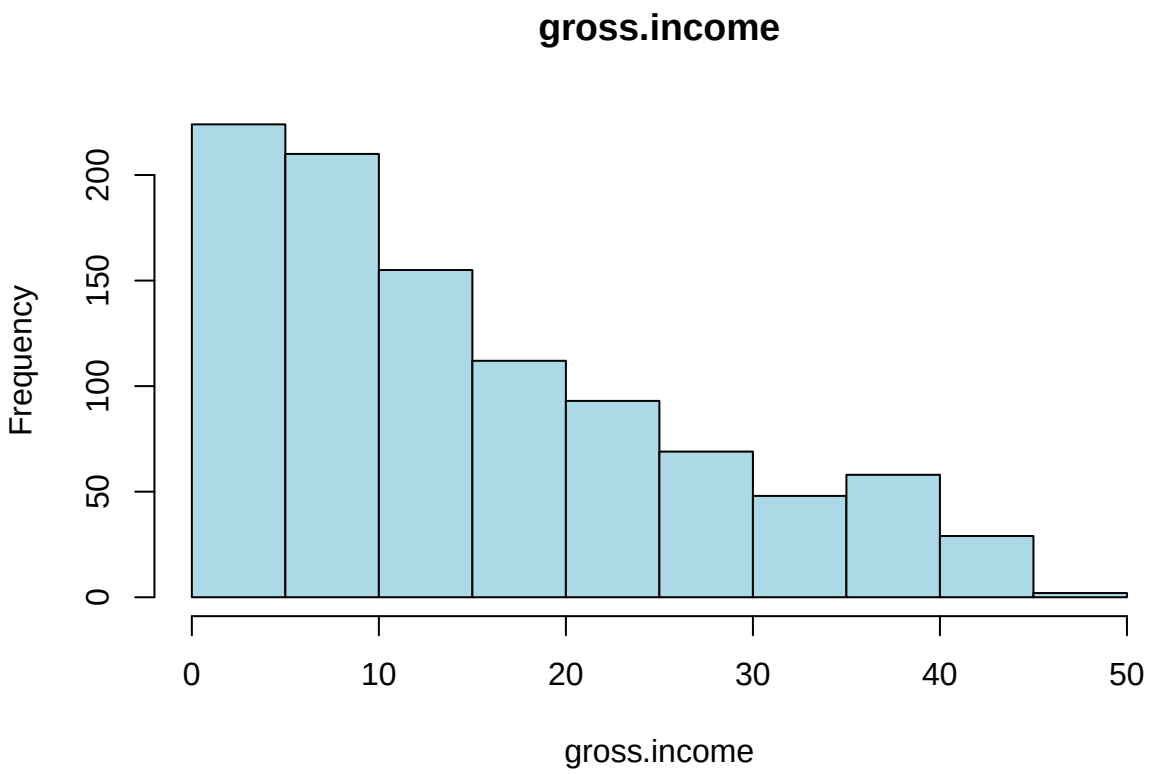
```
#plotting histograms of the numerical columns
histogram = function(x){
  for(i in colnames(x)){
    hist(carrefour1[[i]], breaks = 10,main =i,xlab = i,col = "lightblue")
  }
}
histogram(num_cols)
```

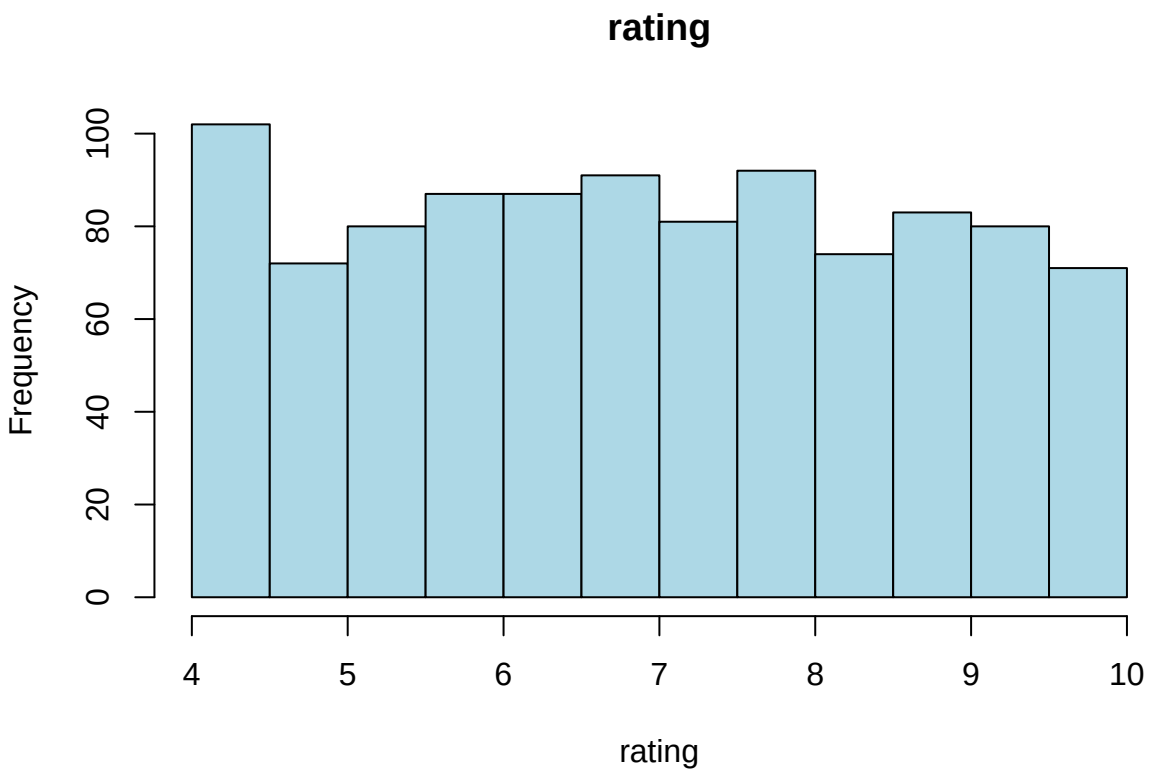

unit.price
Frequency
0
20
40
60
80
100
20
40
60
80
100
unit.price

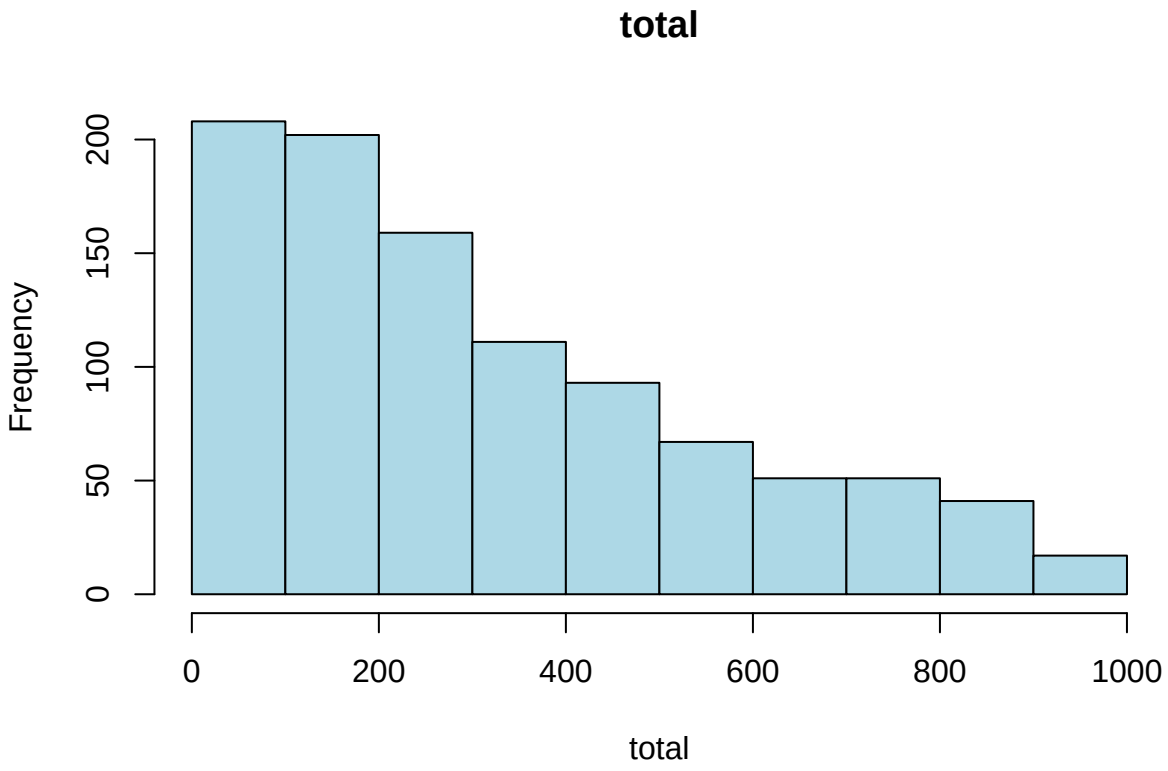

quantity
Frequency
0
50
100
150
200
2
4
6
8
10
quantity


tax
Frequency
0
50
100
150
200
0
10
20
30
40
50
tax

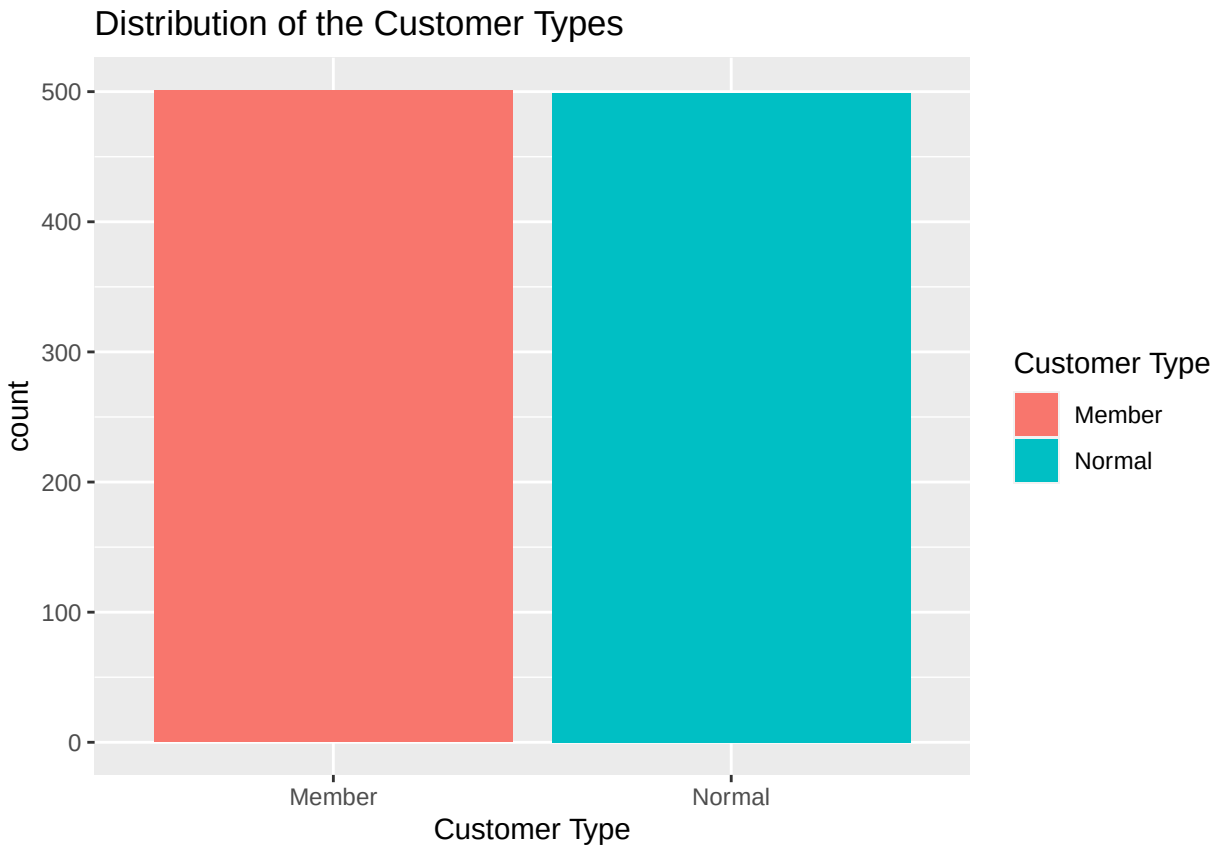

cogs
Frequency
0
50
100
150
200
0
200
400
600
800
1000
cogs


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

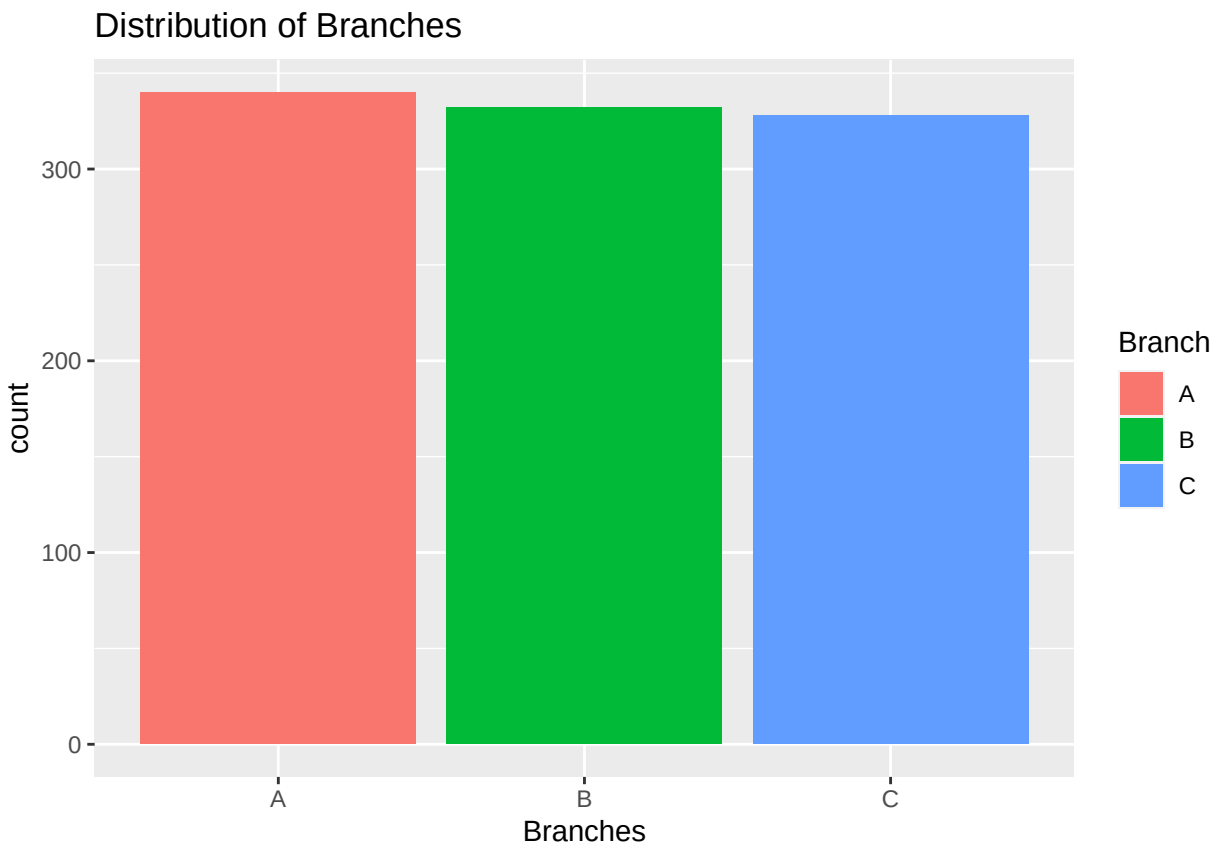

rating
Frequency
0
20
40
60
80
100
4
5
6
7
8
9
10
rating



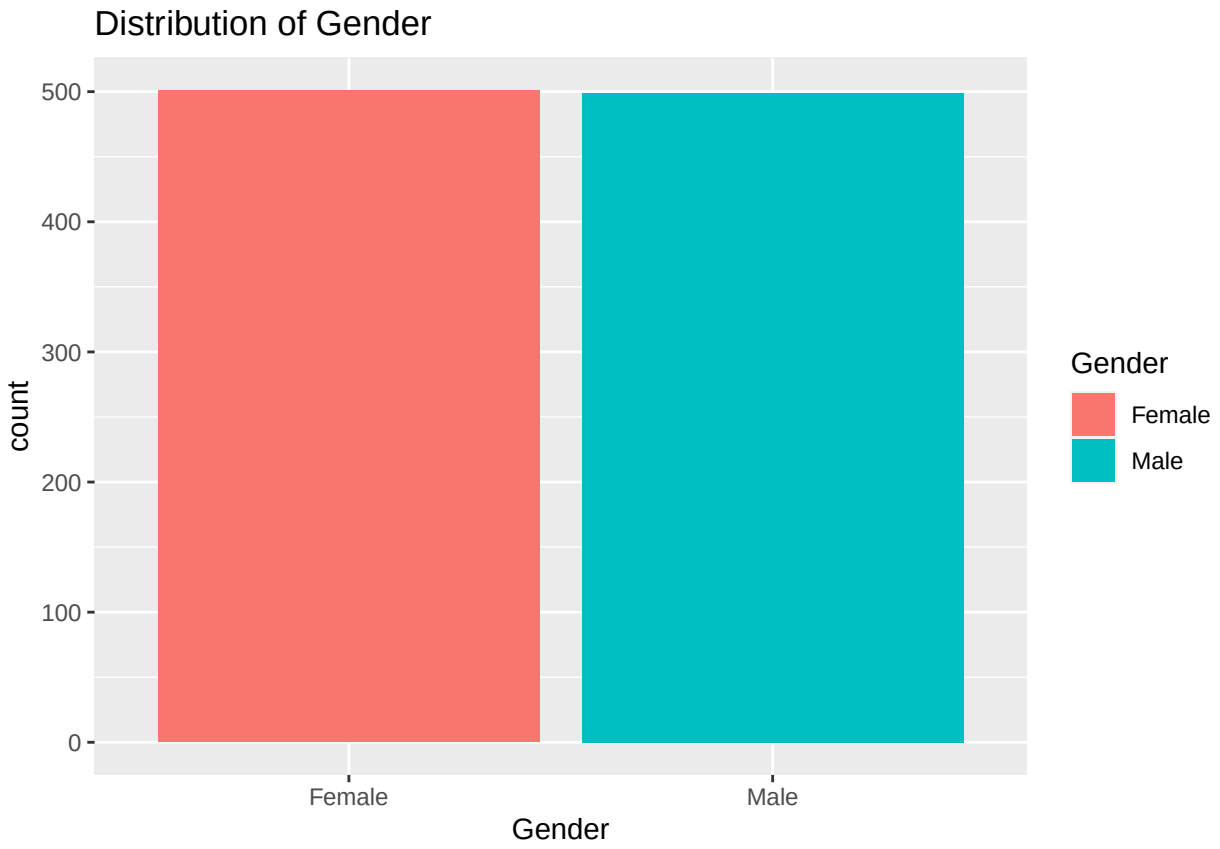

```
library(ggplot2)
# Customer Type
customerdist <- ggplot(carrefour1 ,aes(x=customer.type , fill=customer.type)) + geom_bar() + labs(title
customerdist +scale_fill_discrete(name = "Customer Type")
```



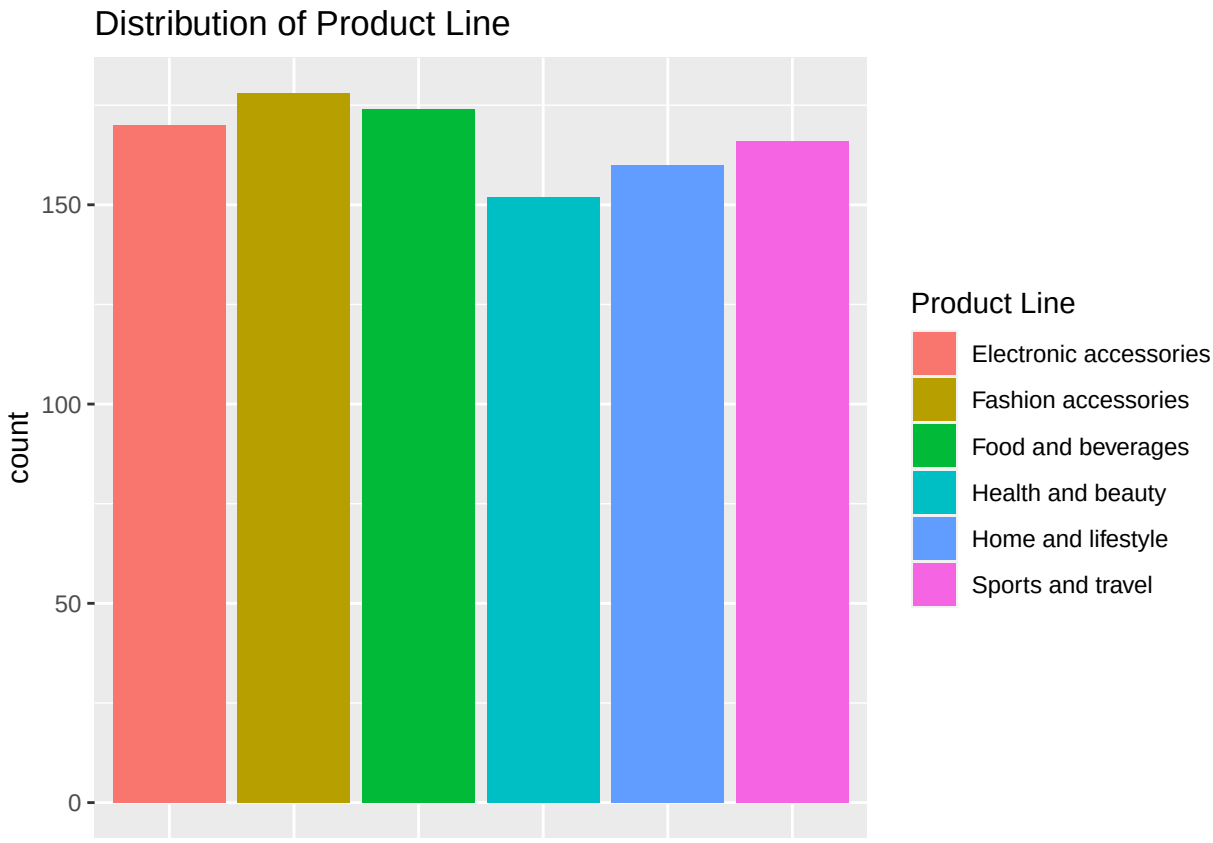

```
# How many Branches we have
branchdist <- ggplot(carrefour1, aes(x=branch, fill=branch)) + geom_bar()+labs(title = "Distribution of
branchdist
```



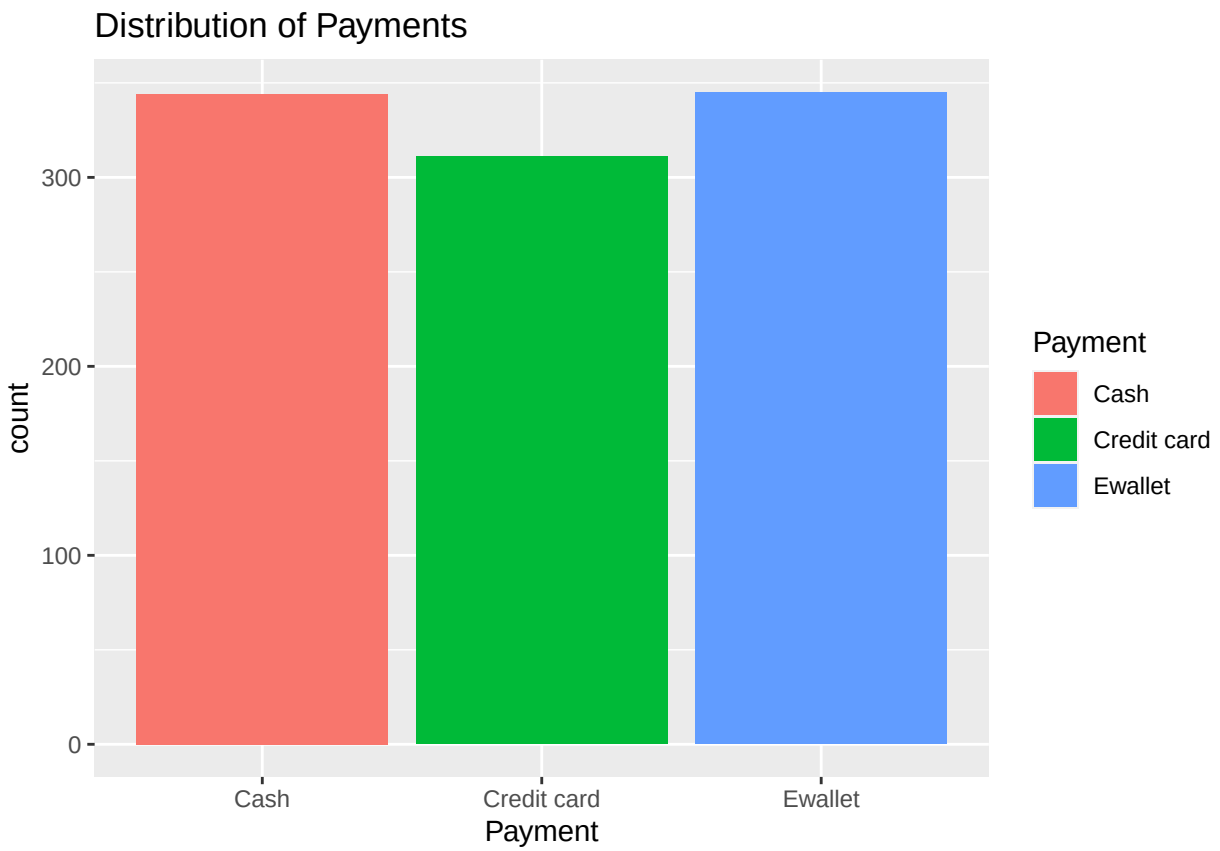

```
# Gender Distribution
genderdist <- ggplot(carrefour1, aes(x=gender, fill=gender)) + geom_bar()+labs(title = "Distribution of
genderdist
```




```
# Types Product Line
productlinedist <- ggplot(carrefour1,aes(x=product.line, fill=product.line))+ geom_bar()+ labs(title = 
productlinedist + theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

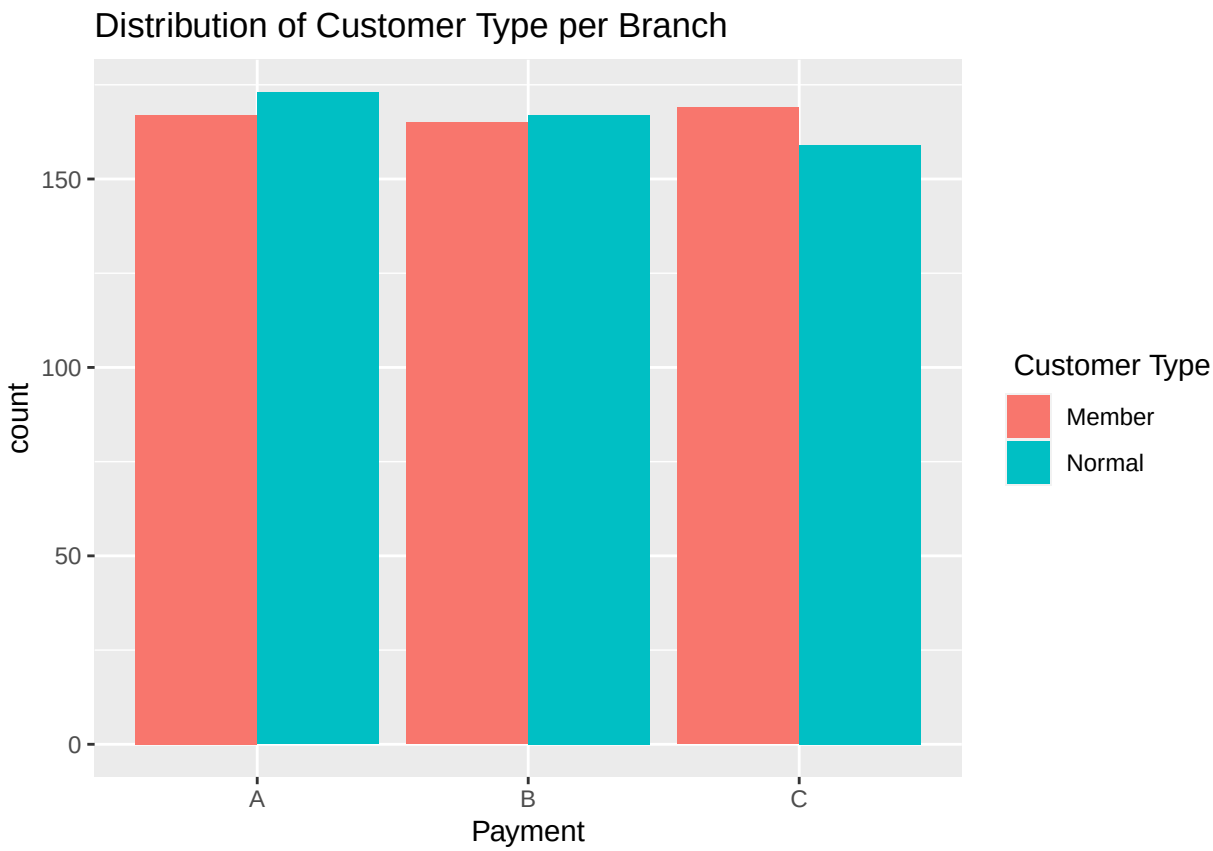



```
# Types Payments
paymentdist <- ggplot(carrefour1, aes(x=payment, fill=payment)) + geom_bar()+labs(title = "Distribution
paymentdist
```

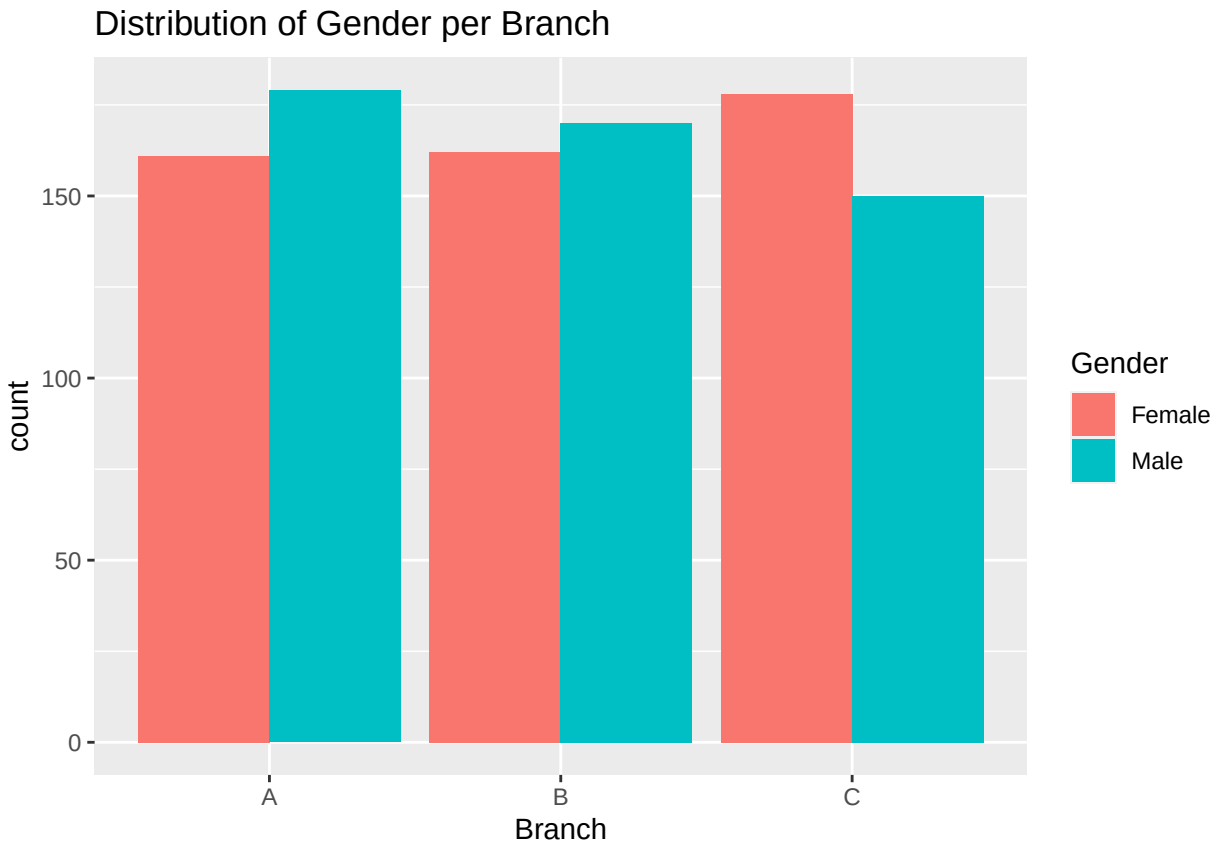


Bivariate Analysis

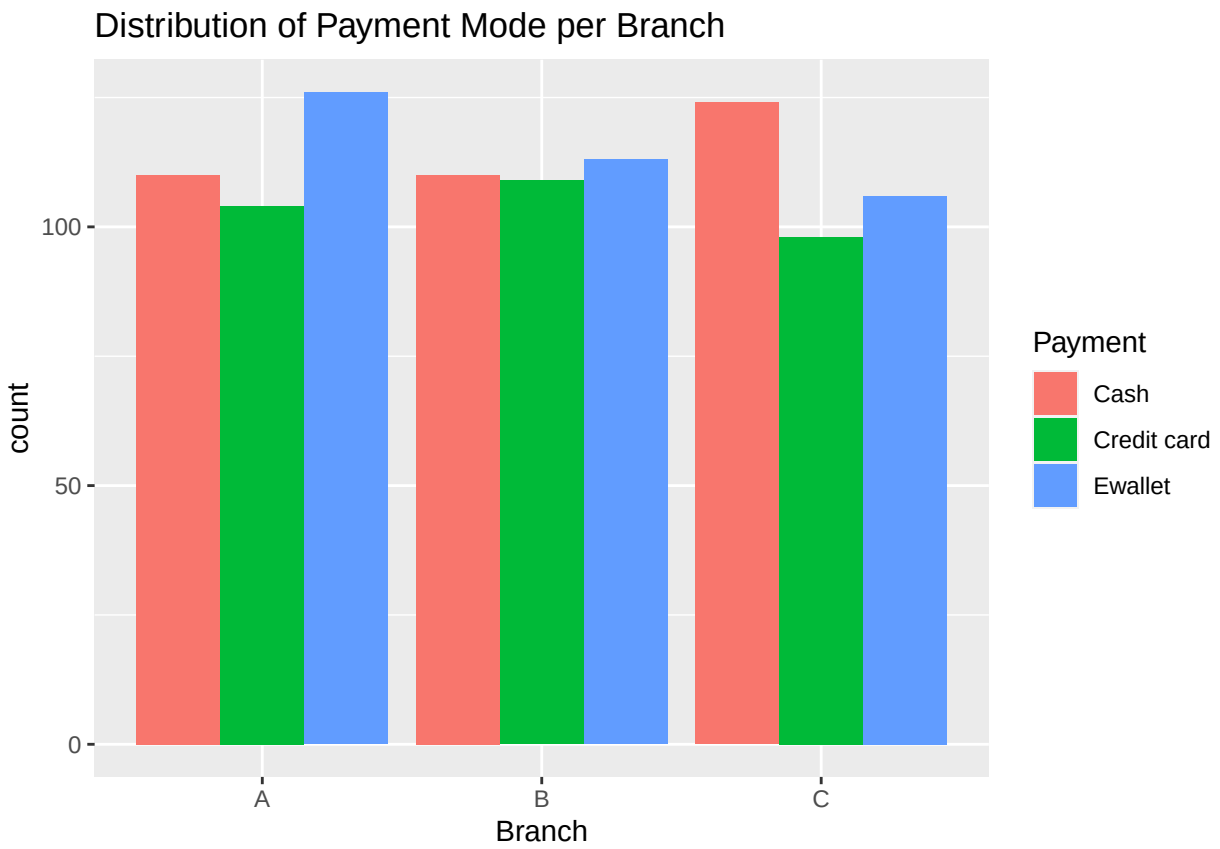
```
#Customer Type groupedby Branch
c_typecomparison <- ggplot(carrefour1, aes(x=branch, fill=customer.type)) + geom_bar(position = "dodge"
c_typecomparison
```



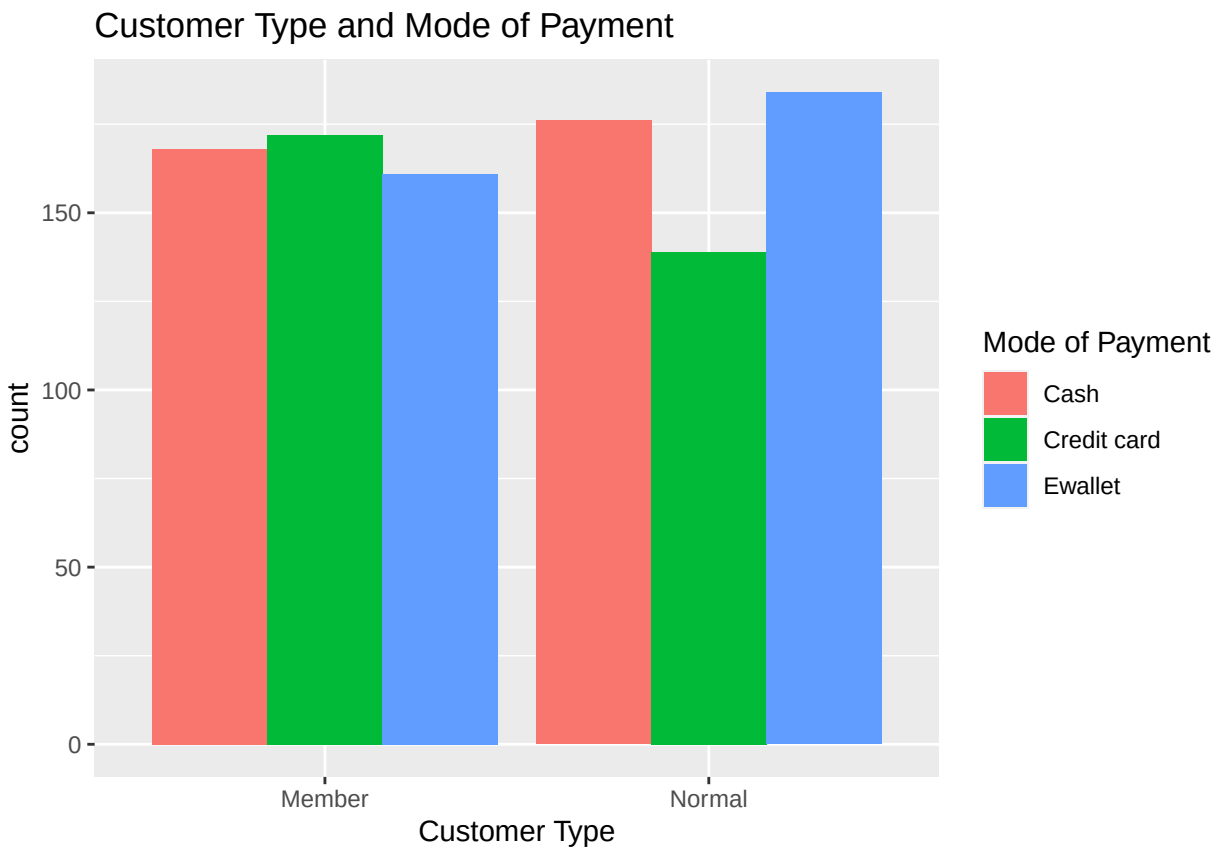

```
#Gender per Branch
gendercomp <- ggplot(carrefour1, aes(x=branch, fill=gender)) + geom_bar(position = "dodge")+labs(title =
gendercomp
```



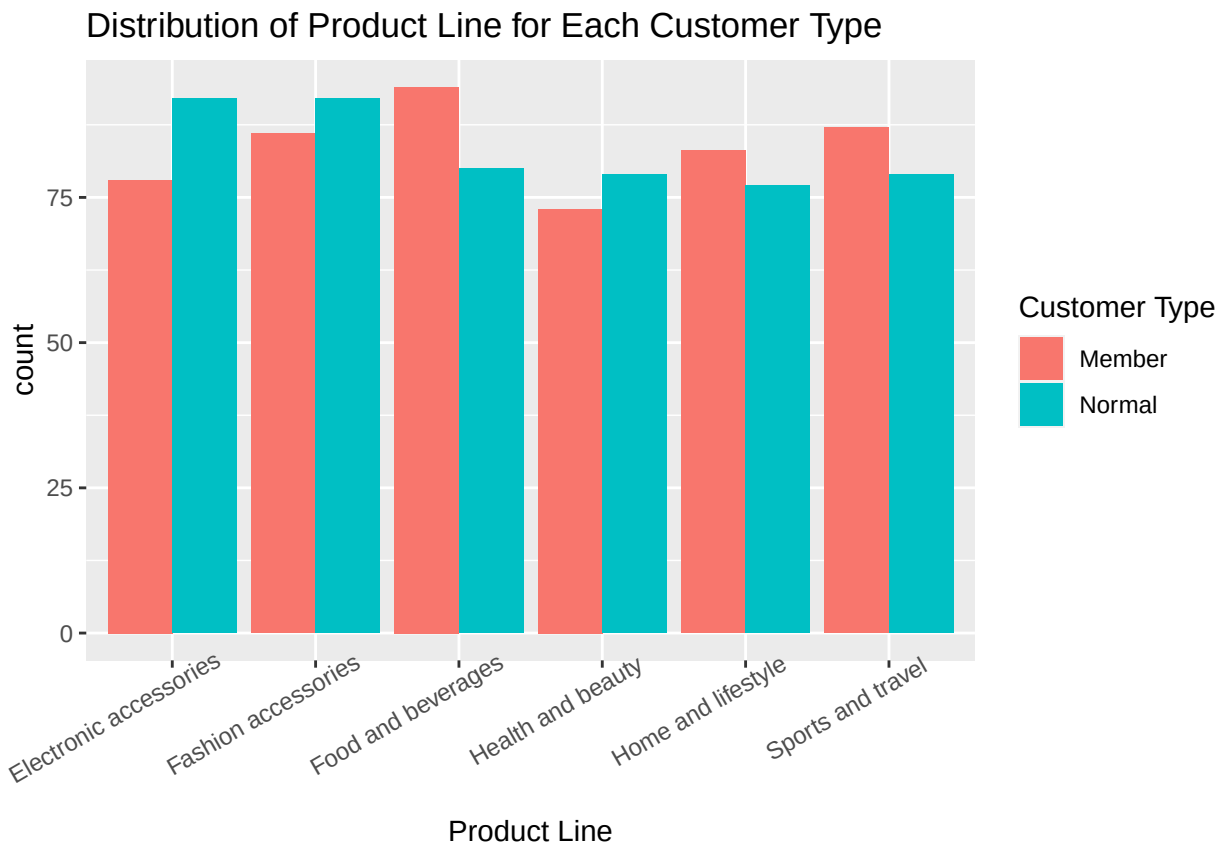

```
#Payment mode per Branch
paymentcomp <- ggplot(carrefour1, aes(x=branch, fill=payment)) + geom_bar(position = "dodge")+labs(titl
paymentcomp
```



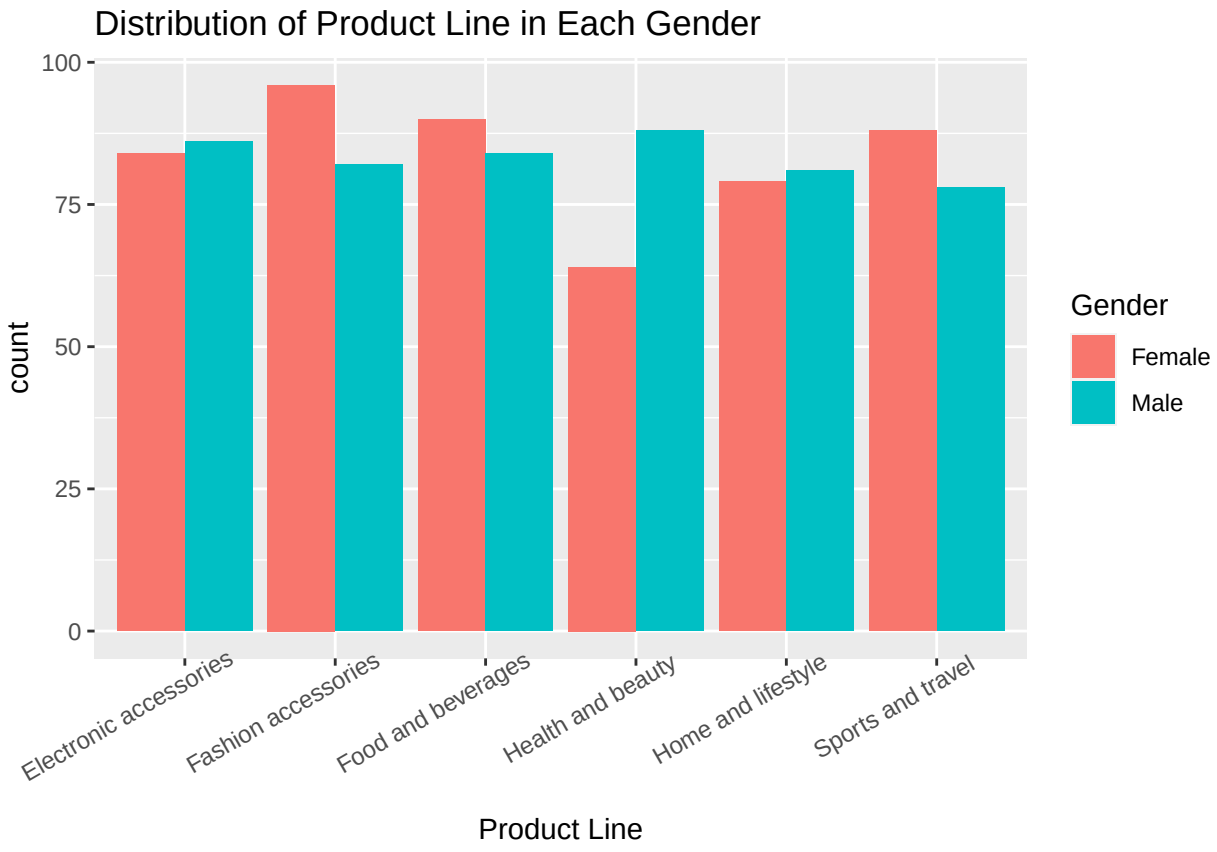

```
# Customer Type vs Mode of Payment
c_typecomp2 <- ggplot(carrefour1, aes(x=customer.type, fill=payment)) + geom_bar(position = "dodge")+lab
c_typecomp2
```



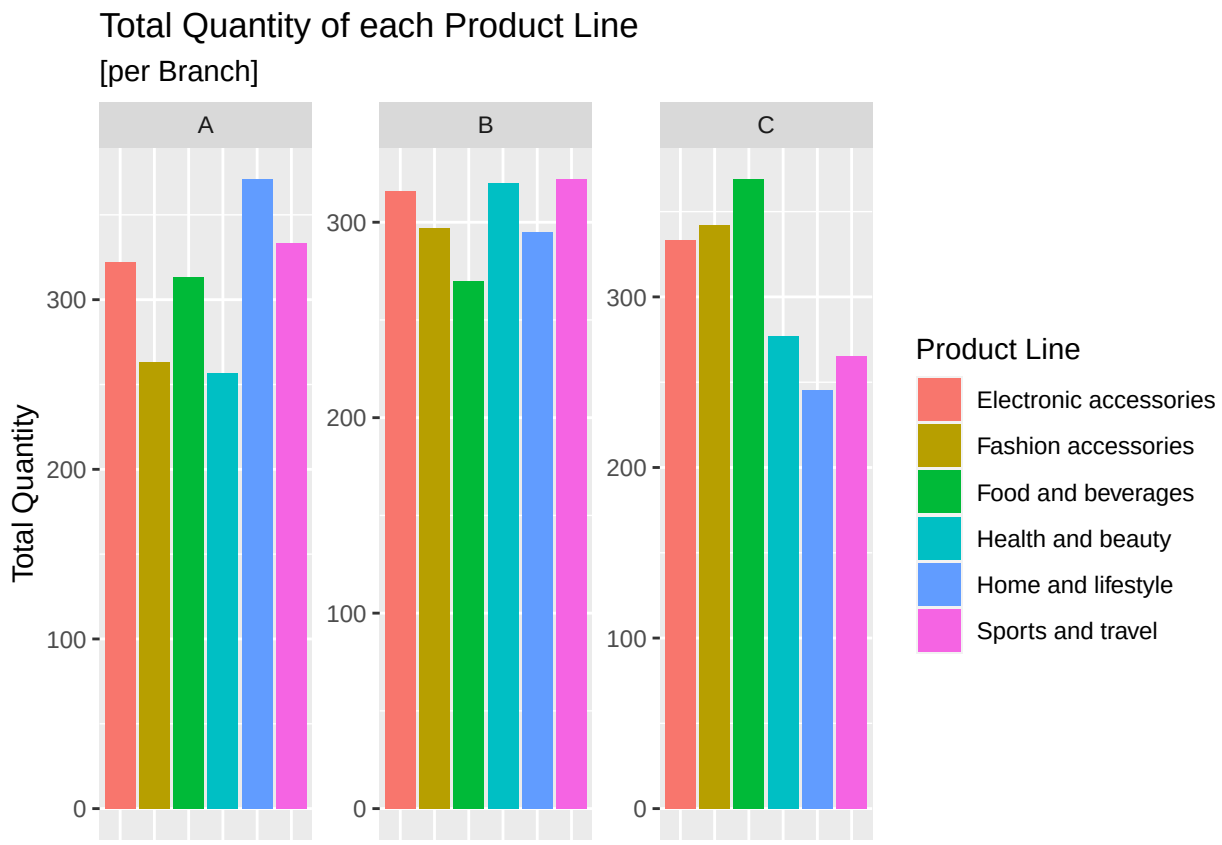

```
# Customer Type and Product Line
prodlinecomp2 <- ggplot(carrefour1, aes(x=product.line, fill=customer.type)) + geom_bar(position = "dod
prodlinecomp2 +theme(axis.text.x = element_text(angle = 30, hjust=0.8))
```



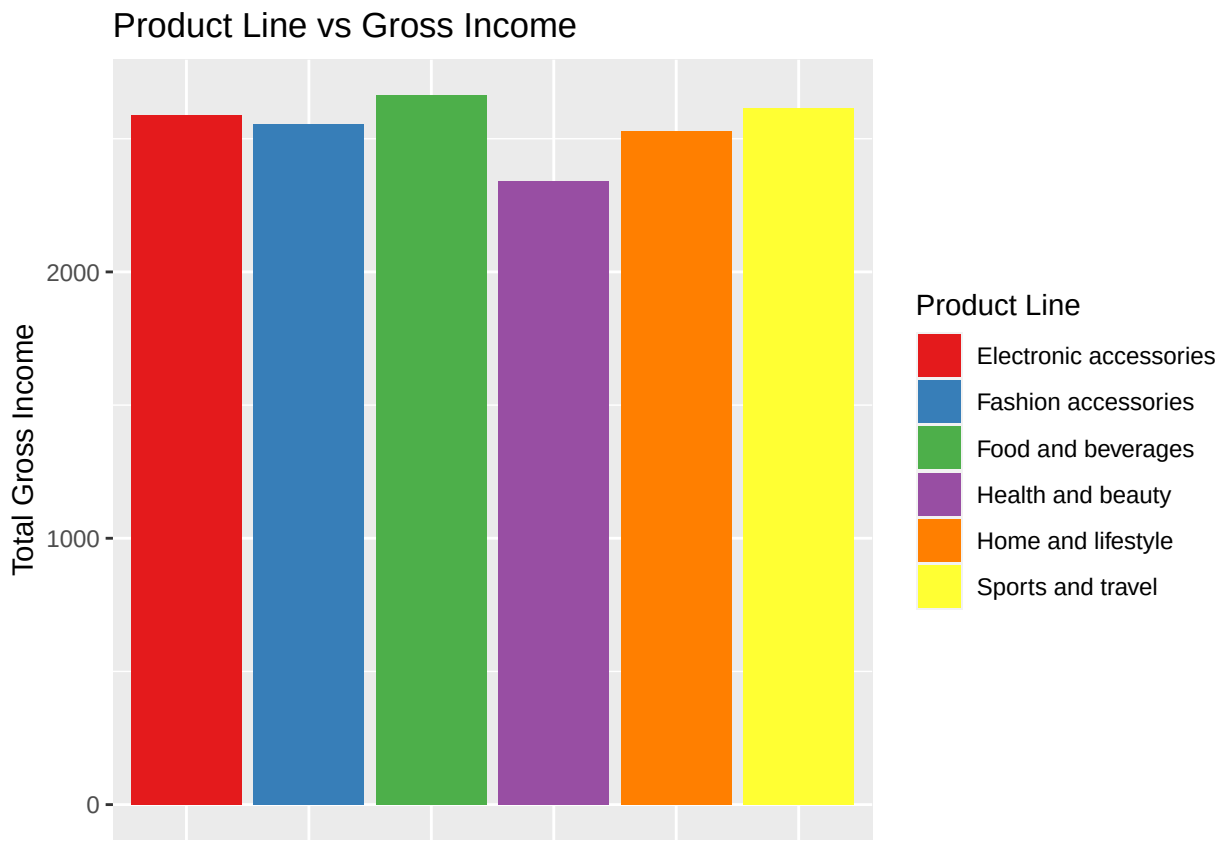

```
# Product Line vs Gender
prodlinecomp3 <- ggplot(carrefour1, aes(x=product.line, fill=gender)) + geom_bar(position = "dodge")+lab
prodlinecomp3 +theme(axis.text.x = element_text(angle = 30, hjust=0.8))
```



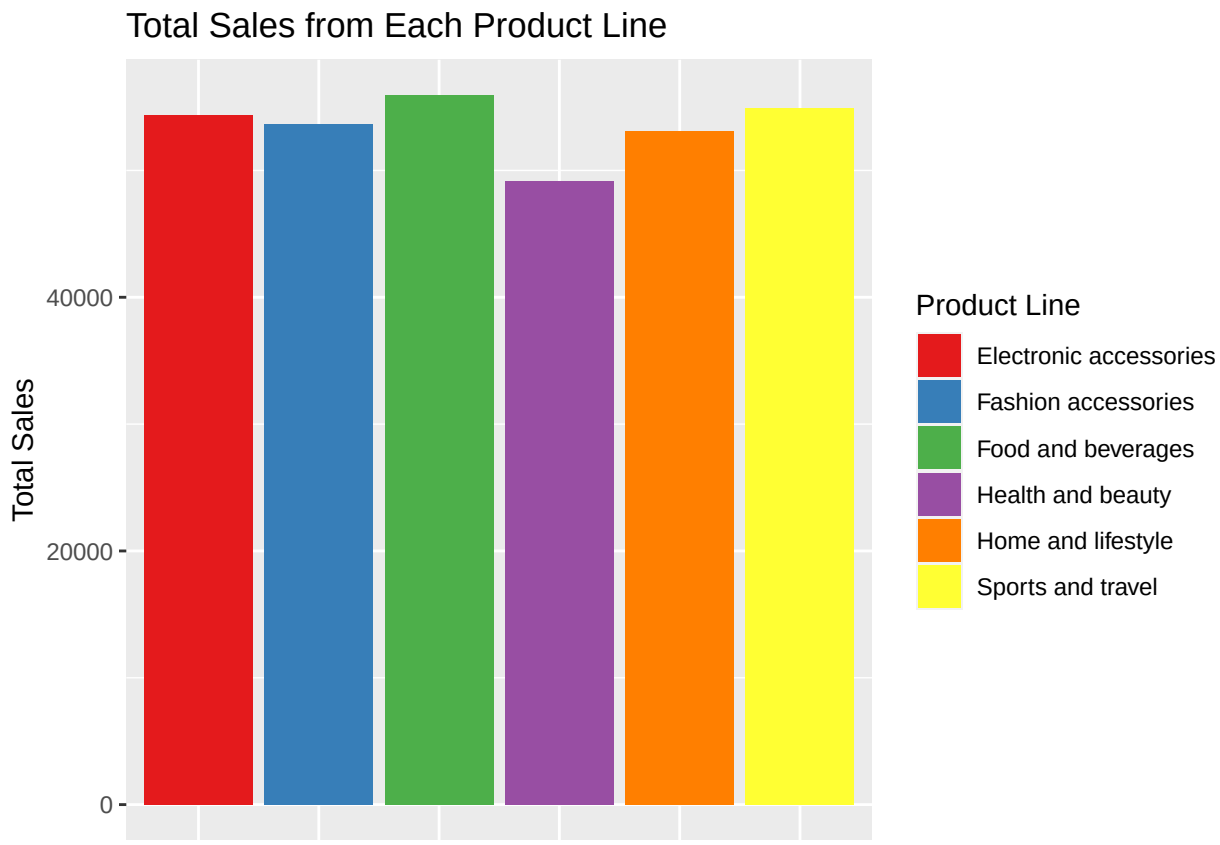

```
#Product line vs quantity per branch
prodlinecomparison <- ggplot(carrefour1,aes(x=product.line,y=quantity,fill=product.line))+geom_bar(stat
            axis.text.x=element_blank(),
            axis.ticks.x=element_blank())
prodlinecomparison
```



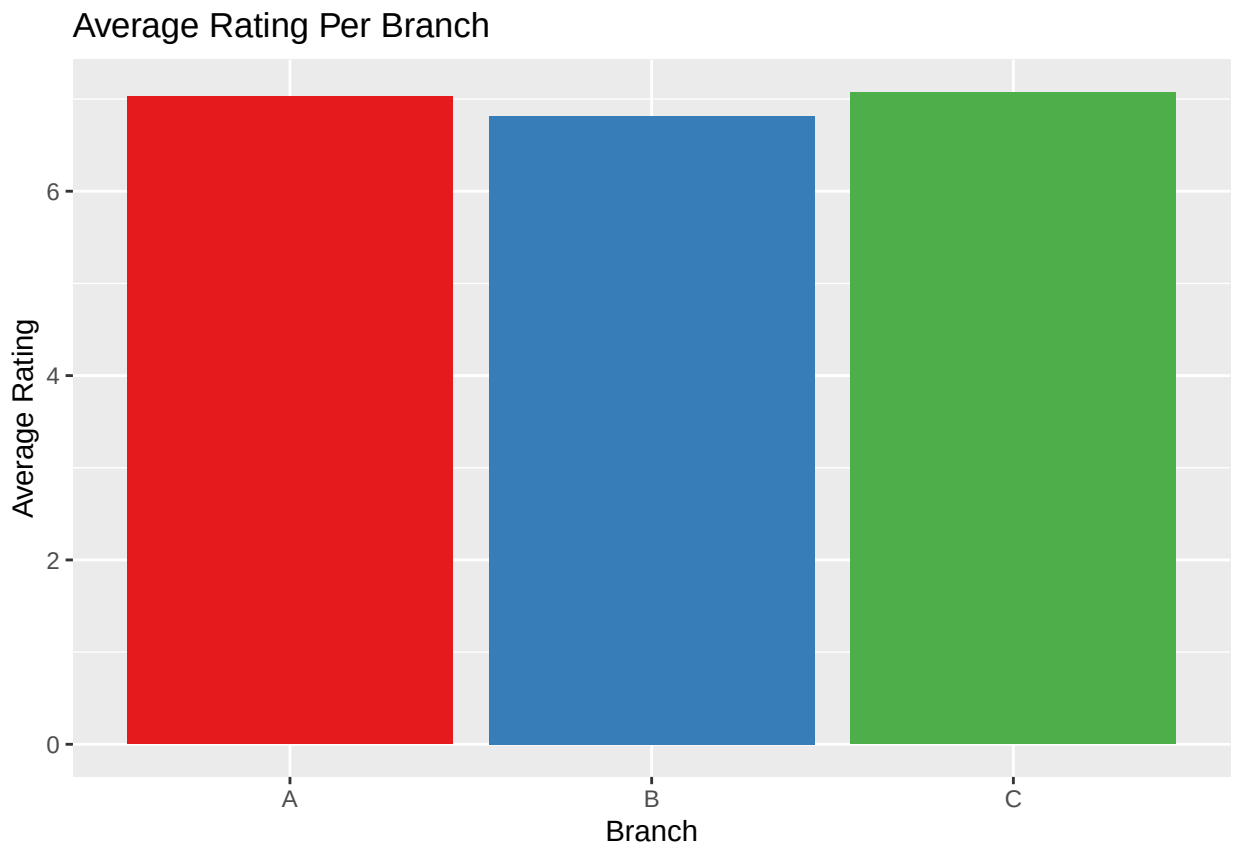

```
# Product Line with Highest Gross Income
plineby_gross <- carrefour1 %>%
  group_by(product.line) %>%
  summarise(gross = sum(gross.income))
gross_inc <- ggplot(plineby_gross, aes(x=product.line, y= gross, fill=product.line))+geom_bar(stat = "i
gross_inc + scale_fill_brewer(name="Product Line",palette="Set1")+theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```




```
# Product Line total sales
plineby_sale <- carrefour1 %>%
  group_by(product.line) %>%
  summarise(total = sum(total))
total_sale <- ggplot(plineby_sale, aes(x=product.line, y= total, fill=product.line))+geom_bar(stat = "i
total_sale + scale_fill_brewer(name="Product Line",palette="Set1")+theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

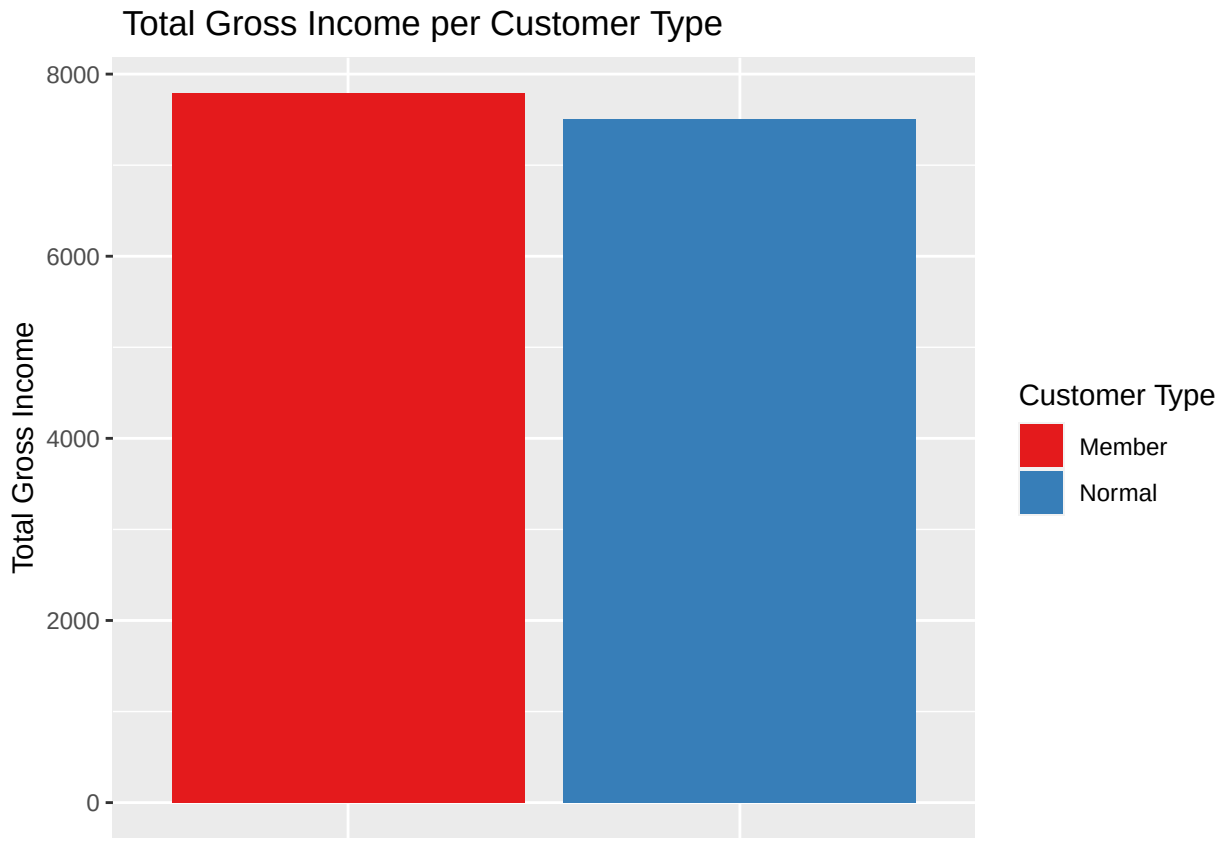



```
#Banch ratings
branchby_rate <- carrefour1 %>%
  group_by(branch) %>%
  summarise(rate = mean(rating))
rate <- ggplot(branchby_rate, aes(x=branch, y= rate, fill=branch))+geom_bar(stat = "identity")+ labs(ti
rate + scale_fill_brewer(palette="Set1")+theme(legend.position = "none")
```




```
#Branch totalsales
branchby_sale <- carrefour1 %>%
  group_by(branch) %>%
  summarise(total = sum(total))
total_sale <- ggplot(branchby_sale, aes(x=branch, y= total, fill=branch))+geom_bar(stat = "identity")+
total_sale + scale_fill_brewer(name="Branch",palette="Set1")+theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```


```
#customer type  gross income comparison
custtypeby_gross <- carrefour1 %>%
  group_by(customer.type) %>%
  summarise(gross = sum(gross.income))
cust_type <- ggplot(custtypeby_gross, aes(x=customer.type, y= gross, fill=customer.type))+geom_bar(stat
cust_type + scale_fill_brewer( name= "Customer Type",palette="Set1")+theme(axis.title.x=element_blank()
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```




```
# Covariance
covariance = cov(num_cols)
View(round(covariance,2))
```

```
# Correlation matrix
corr_matrix = cor(num_cols)
corr <- as.data.frame(round(corr_matrix,2))
corr
```

```
##              unit.price quantity   tax  cogs gross.income rating total
## unit.price         1.00     0.01  0.63  0.63         0.63  -0.01  0.63
## quantity           0.01     1.00  0.71  0.71         0.71  -0.02  0.71
## tax                0.63     0.71  1.00  1.00         1.00  -0.04  1.00
## cogs               0.63     0.71  1.00  1.00         1.00  -0.04  1.00
## gross.income       0.63     0.71  1.00  1.00         1.00  -0.04  1.00
## rating            -0.01    -0.02 -0.04 -0.04        -0.04   1.00 -0.04
## total              0.63     0.71  1.00  1.00         1.00  -0.04  1.00
```

PCA

```
str(carrefour1)
```

```
## 'data.frame':    1000 obs. of  15 variables:
```

```
##  $ 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                   : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ time                   : chr  "13:08" "10:29" "13:23" "20:33" ...
##  $ 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 ...
```

```
num_features <- carrefour1[, c(5,6,7, 11,13:15)]
names(num_features)
```

```
## [1] "unit.price"   "quantity"     "tax"          "cogs"         "gross.income"
## [6] "rating"       "total"
```

```
carrefour.pca <- prcomp(num_features, center = TRUE, scale. = TRUE)
carrefour.pca
```

```
## Standard deviations (1, .., p=7):
## [1] 2.218853e+00 1.000234e+00 9.938935e-01 2.973248e-01 4.404879e-16
## [6] 2.485173e-16 1.053133e-16
##
## Rotation (n x k) = (7 x 7):
##                       PC1          PC2          PC3         PC4           PC5
## unit.price   -0.29155950  0.266868775 -0.695957402 -0.59951431 -9.027383e-16
## quantity     -0.32506750 -0.211969611  0.632411590 -0.67041449 -1.131948e-15
## tax          -0.44972695  0.004204886  0.001831449  0.21845969 -8.340802e-01
## cogs         -0.44972695  0.004204886  0.001831449  0.21845969  6.584550e-02
## gross.income -0.44972695  0.004204886  0.001831449  0.21845969  3.347094e-01
## rating        0.01751726  0.940095323  0.340125651  0.01511523 -7.525791e-18
## total        -0.44972695  0.004204886  0.001831449  0.21845969  4.335253e-01
##                       PC6           PC7
## unit.price   -7.344044e-17 -6.845086e-17
## quantity      8.202822e-17 -1.099768e-16
## tax          -2.210692e-01 -7.374712e-02
## cogs          8.611599e-01  6.378135e-02
## gross.income -3.683882e-01  7.087029e-01
## rating        9.791839e-17 -9.167026e-19
## total        -2.717025e-01 -6.987371e-01
```

```
summary(carrefour.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4       PC5       PC6
## Standard deviation     2.2189 1.0002 0.9939 0.29732 4.405e-16 2.485e-16
```

```
## Proportion of Variance 0.7033 0.1429 0.1411 0.01263 0.000e+00 0.000e+00
## Cumulative Proportion  0.7033 0.8462 0.9874 1.00000 1.000e+00 1.000e+00
##                            PC7
## Standard deviation     1.053e-16
## Proportion of Variance 0.000e+00
## Cumulative Proportion  1.000e+00
```

```
str(carrefour.pca)
```

```
## List of 5
##  $ sdev    : num [1:7] 2.22 1.00 9.94e-01 2.97e-01 4.40e-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 15.29 305.86 15.29 ...
##   ..- attr(*, "names")= chr [1:7] "unit.price" "quantity" "tax" "cogs" ...
##  $ scale   : Named num [1:7] 26.49 2.92 11.5 229.92 11.5 ...
##   ..- attr(*, "names")= chr [1:7] "unit.price" "quantity" "tax" "cogs" ...
##  $ x       : num [1:1000, 1:7] -2.051 2.323 -0.203 -1.541 -2.854 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:7] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"
```

Feature Selection

```
library(caret)
```

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

```
##
## Attaching package: 'lattice'
```

```
## The following object is masked _by_ '.GlobalEnv':
##
##     histogram
```

```
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
highlyCorrelated <- findCorrelation(corr_matrix, cutoff=0.75)
```

```
highlyCorrelated
```

```
## [1] 4 7 3
```

```
names(carrefour1[,highlyCorrelated])
```

```
## [1] "product.line" "tax"          "gender"
```

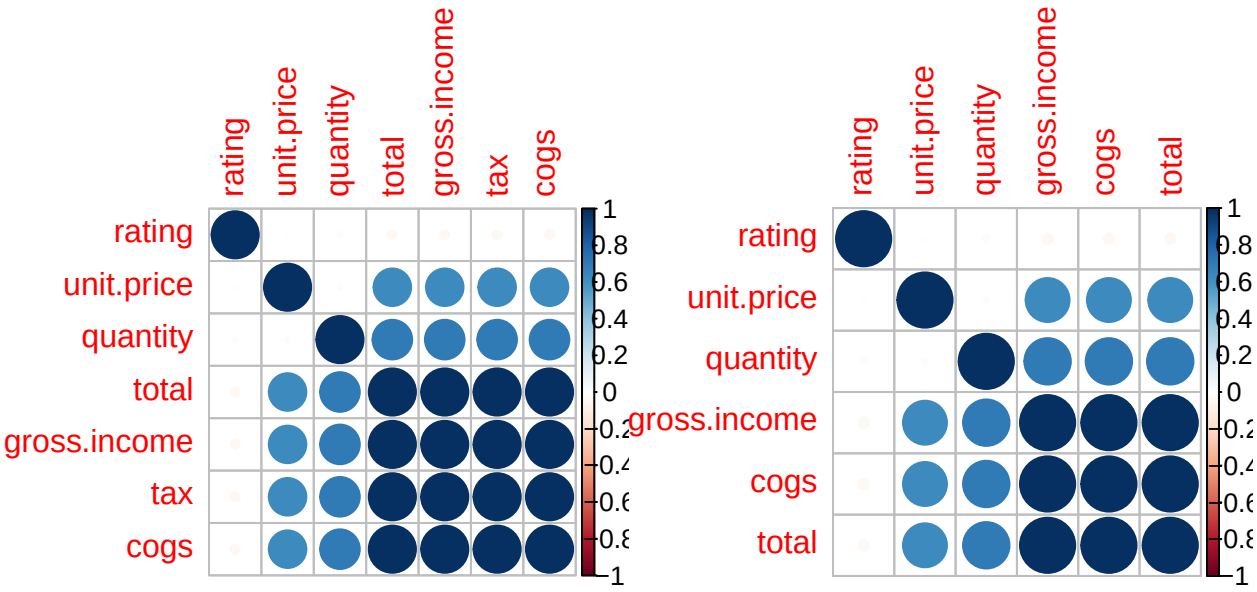
```
carrefour2<-carrefour1[-highlyCorrelated]
str(carrefour2)
```

```
## 'data.frame':    1000 obs. of  12 variables:
##  $ 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 ...
##  $ 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 ...
##  $ 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                : 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 ...
```

```
num_features2 <- carrefour2[, c(3,4,8, 10:12)]
names(num_features2)
```

```
## [1] "unit.price"   "quantity"     "cogs"         "gross.income" "rating"
## [6] "total"
```

```
# Performing our graphical comparison
# ---
#
par(mfrow = c(1, 2))
corrplot(corr_matrix, order = "hclust")
corrplot(cor(num_features2), order = "hclust")
```



Variables acceptable were: unit.price, quantity, cogs, gross.income, rating and total"

Wrapper Method

```
library(clustvarsel)
```

```
## Loading required package: mclust
```

```
## Package 'mclust' version 5.4.7
## Type 'citation("mclust")' for citing this R package in publications.
```

```
## Package 'clustvarsel' version 2.3.4
```

```
## Type 'citation("clustvarsel")' for citing this R package in publications.
```

```
library(mclust)
```

```
head(num_features2)
```

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

```
# out = clustvarsel(num_features2, G=1:6)
# out
# # Features selected
# data <- num_features2[,out$subset]
# head(data)
```

Our wrapper method accepted gross income but was taking long to complete the search hence terminated it.

Feature ranking

```
library(FSelector)

feat <- num_features2
Scores <- linear.correlation(total~., feat)
Scores
```

```
##              attr_importance
## unit.price        0.63384019
## quantity          0.70705129
## cogs              1.00000000
## gross.income      1.00000000
## rating            0.03392365
```

```
# Selected Features
Subset <- cutoff.k(Scores, 5)
as.data.frame(Subset)
```

```
##         Subset
## 1         cogs
## 2 gross.income
## 3     quantity
## 4   unit.price
## 5       rating
```

With feature ranking we had cogs, gross.income, quantity, unit.price and rating