feature selection

kelvin njunge

9/9/2021

PROBLEM DEFINITION

a) Specifying the Question

reducing your dataset to a low dimensional dataset using the PCA

b) Defining the metrics for success

This section of the project entails reducing your dataset to a low dimensional dataset using the PCA. You will be required to perform your analysis and provide insights gained from your analysis.

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

1. Define the question, the metric for success, the context, experimental design taken.
2. Read and explore the given dataset.
3. reducing your dataset to a low dimensional dataset using the PCA

e) Relevance of the data

The data used for this project will inform the marketing department on the most relevant marketing strategies that will result in the highest no. of sales (total price including tax)

Loading data

```
getwd()
```

```
## [1] "C:/Users/Ricky/Documents"
```

```r
sales <- read.csv("C:\\Users\\Ricky\\Documents\\Supermarket_Dataset_1 - Sales Data.csv")

head(sales)
```

```
##    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
# checkiing for size of the dataset
dim(sales)
```

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

```r
# Summary
summary(sales)
```

```
##   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
```

Tidying the data

```
# Checking for unique values are in variable
rapply(sales,function(x)length(unique(x)))
```

```
##              Invoice.ID                  Branch           Customer.type
##                    1000                       3                       2
##                  Gender           Product.line              Unit.price
##                       2                       6                     943
##                Quantity                     Tax                    Date
##                      10                     990                      89
##                    Time                 Payment                    cogs
##                     506                       3                     990
## gross.margin.percentage            gross.income                  Rating
##                       1                     990                      61
##                   Total
##                     990
```

```
# checking for duplicates
#df[duplicated(df), ]
```

```
# checking for missing values
#colSums(is.na(df))
```

```
#Dropping columns
sales <- subset(sales, select = -c(Invoice.ID,gross.margin.percentage))
```

```
head(sales)
```

```
##   Branch Customer.type Gender           Product.line Unit.price Quantity
## 1      A        Member Female      Health and beauty      74.69        7
## 2      C        Normal Female Electronic accessories      15.28        5
## 3      A        Normal   Male     Home and lifestyle      46.33        7
## 4      A        Member   Male      Health and beauty      58.22        8
## 5      A        Normal   Male      Sports and travel      86.31        7
## 6      C        Normal   Male Electronic accessories      85.39        7
##       Tax      Date  Time     Payment   cogs gross.income Rating    Total
## 1 26.1415  1/5/2019 13:08     Ewallet 522.83      26.1415    9.1 548.9715
## 2  3.8200  3/8/2019 10:29        Cash  76.40       3.8200    9.6  80.2200
```

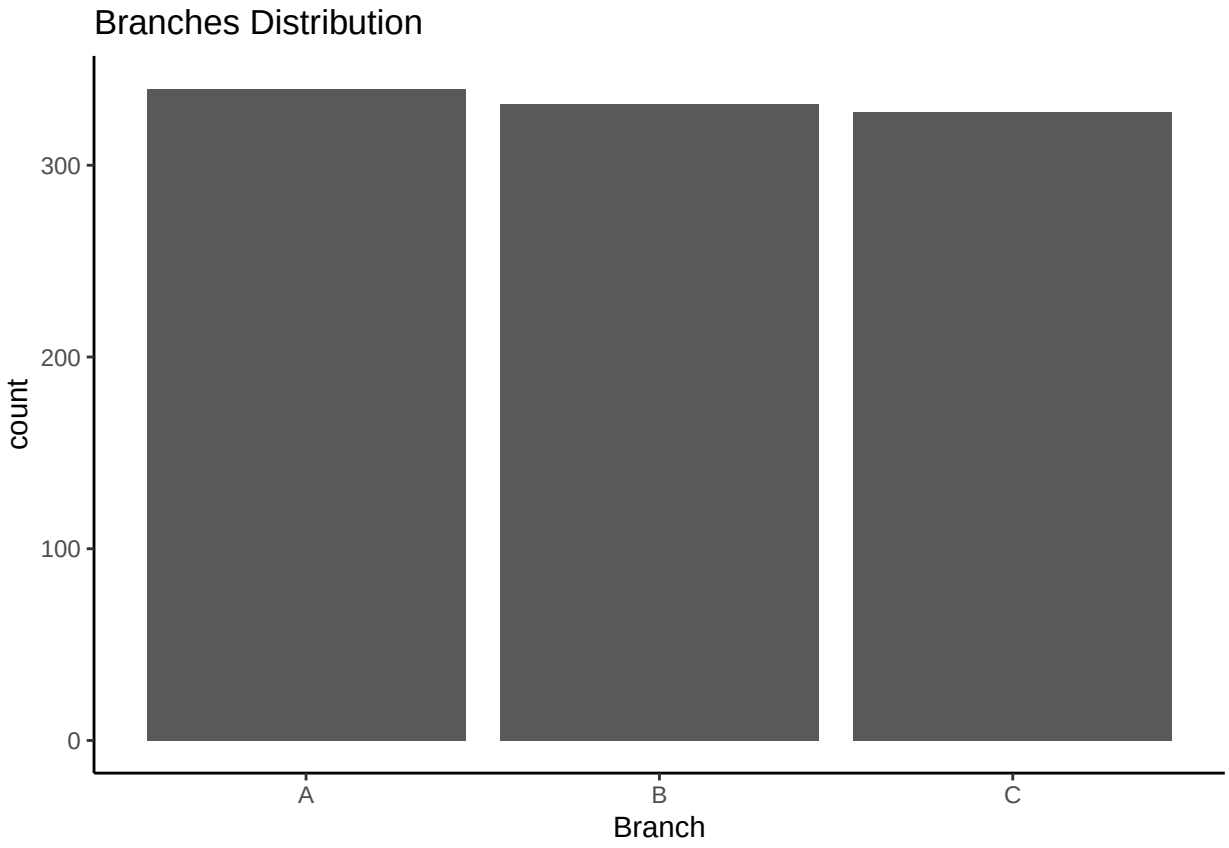
```
## 3 16.2155  3/3/2019 13:23 Credit card 324.31      16.2155    7.4 340.5255
## 4 23.2880 1/27/2019 20:33     Ewallet 465.76      23.2880    8.4 489.0480
## 5 30.2085  2/8/2019 10:37     Ewallet 604.17      30.2085    5.3 634.3785
## 6 29.8865 3/25/2019 18:30     Ewallet 597.73      29.8865    4.1 627.6165
```

Exploratory Data analysis

Univariate Analysis

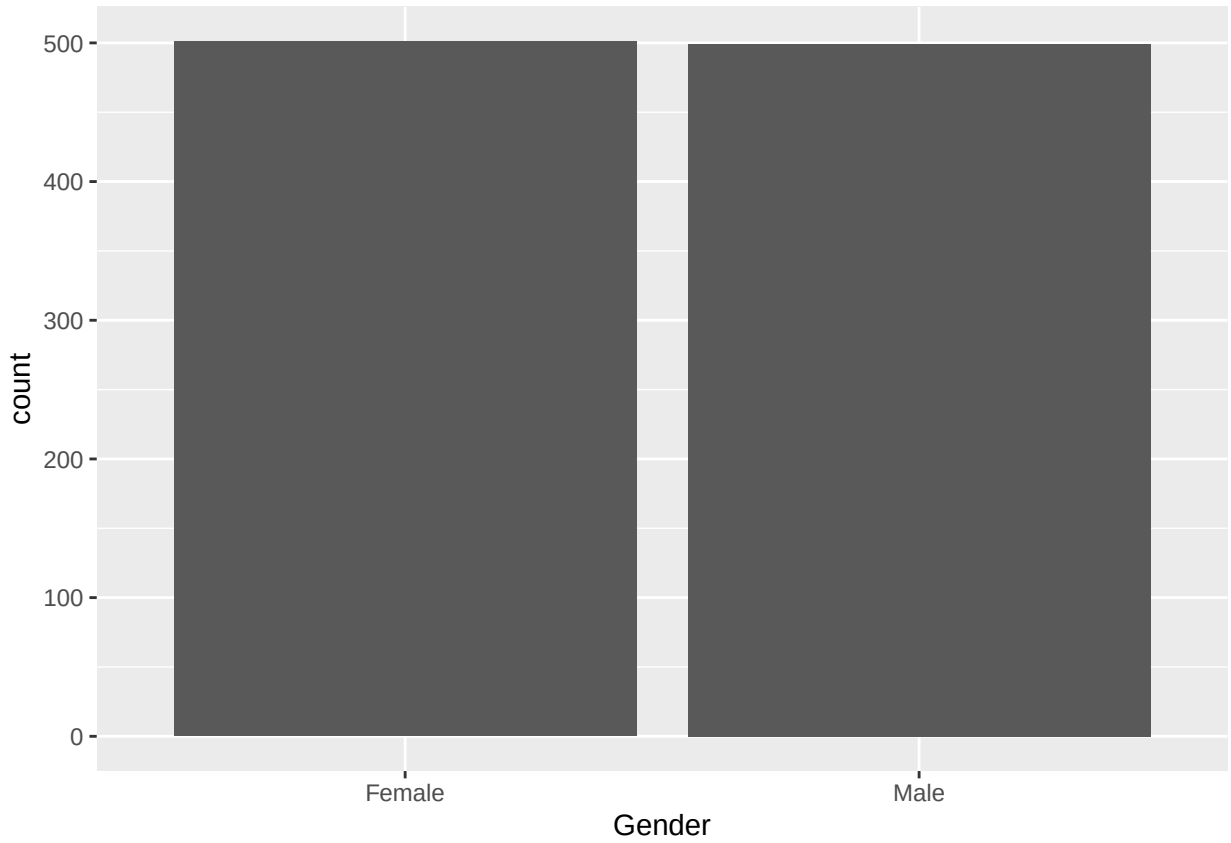
```
# creating a mode function
mode <- function(x){
  uniqx <- unique(x)
  uniqx[which.max(tabulate(match(x, uniqx)))]}
```

```
library(ggplot2)
ggplot(sales,aes(Branch)) + geom_bar(stat='count') + labs(title='Branches Distribution') + theme_classi
```



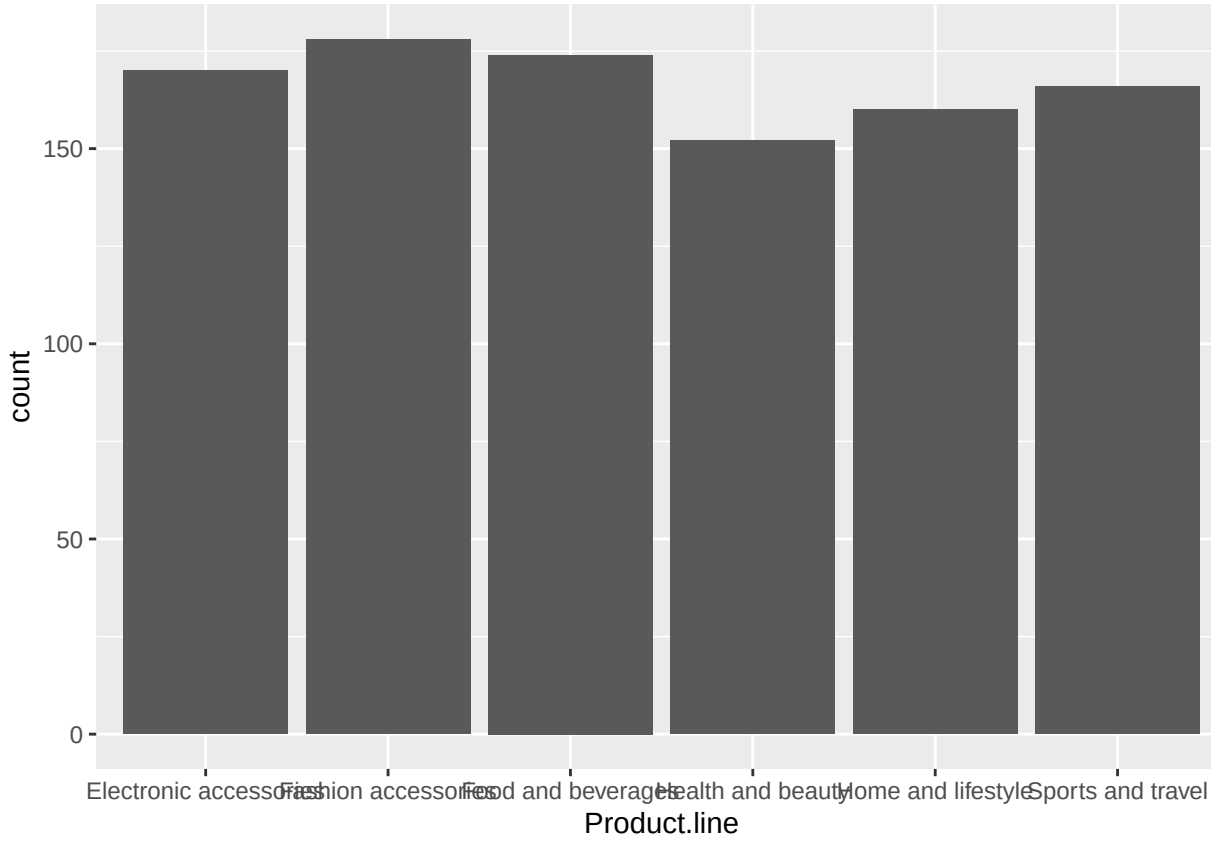
Branch
Branch distribution is roughly equal

```r
# Visualization
ggplot(sales, aes(Gender)) +
  geom_bar(stat="count")
```



Gender
The gender distribution in the dataset is balanced.

```r
# visualization
ggplot(sales, aes(Product.line)) +
  geom_bar()
```



Customer type

Fashion Accessories and, Food and Beverage tie for the most bought categories but the distribution does not suggest an imbalance in general.

```
uprice.mean <- mean(sales$Unit.price)
uprice.mean
```

Unit Price

```
## [1] 55.67213
```

```
# Mode
uprice.mode <- mode(sales$Unit.price)
uprice.mode
```

```
## [1] 83.77
```

```
# Median
uprice.median <- median(sales$Unit.price)
uprice.median
```

```
## [1] 55.23
```

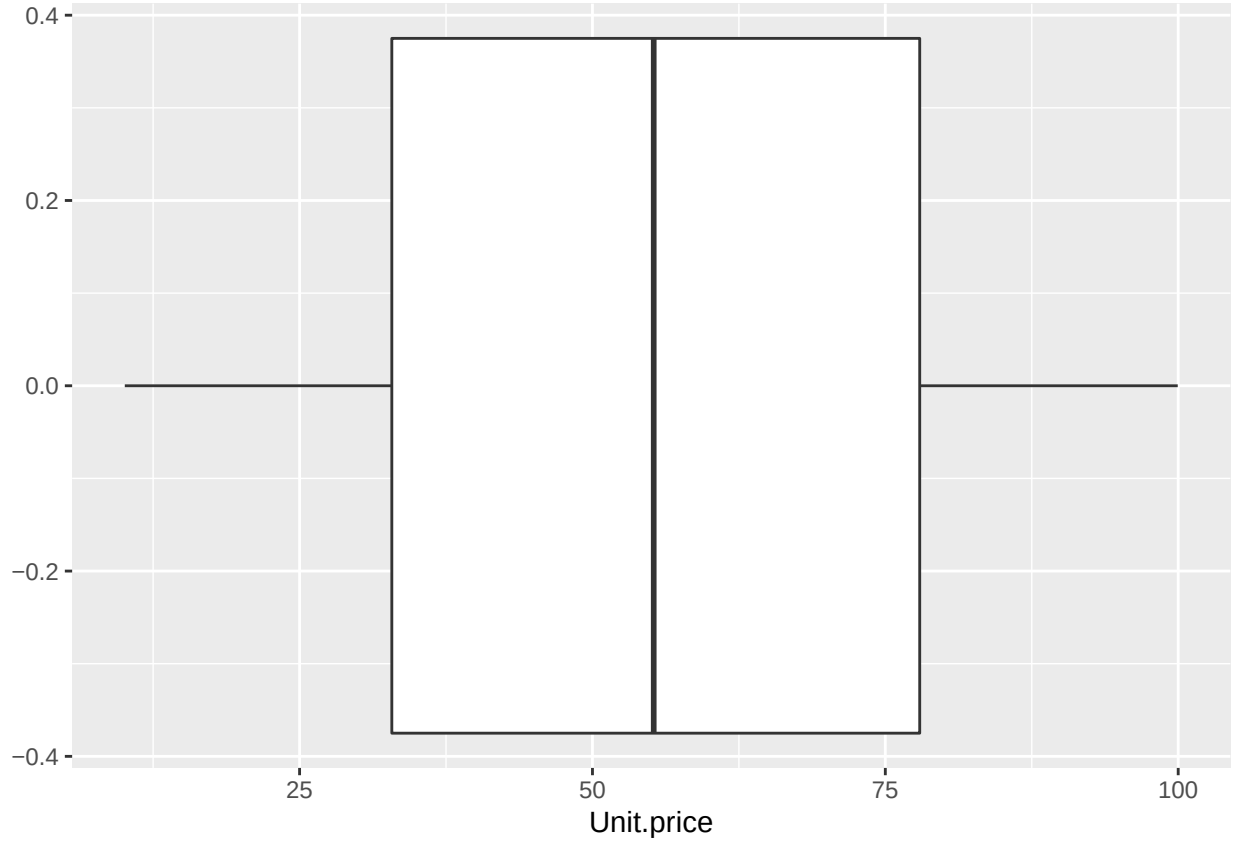
```
# Standard Deviation
uprice.sd <- sd(sales$Unit.price)
uprice.sd
```

```
## [1] 26.49463
```

```
# Range
uprice.range <- range(sales$Unit.price)
uprice.range
```

```
## [1] 10.08 99.96
```

```
# Visualization
ggplot(sales, aes(Unit.price)) +
  geom_boxplot(outlier.colour = "red")
```




```
# mean
quantity.mean <- mean(sales$Quantity)
quantity.mean
```

Quantity

```
## [1] 5.51
```

```
# Mode
quantity.mode <- mode(sales$Quantity)
quantity.mode
```

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

```
# Median
quantity.median <- median(sales$Quantity)
quantity.median
```

```
## [1] 5
```

```
# Standard Deviation
quantity.sd <- sd(sales$Quantity)
quantity.sd
```

```
## [1] 2.923431
```

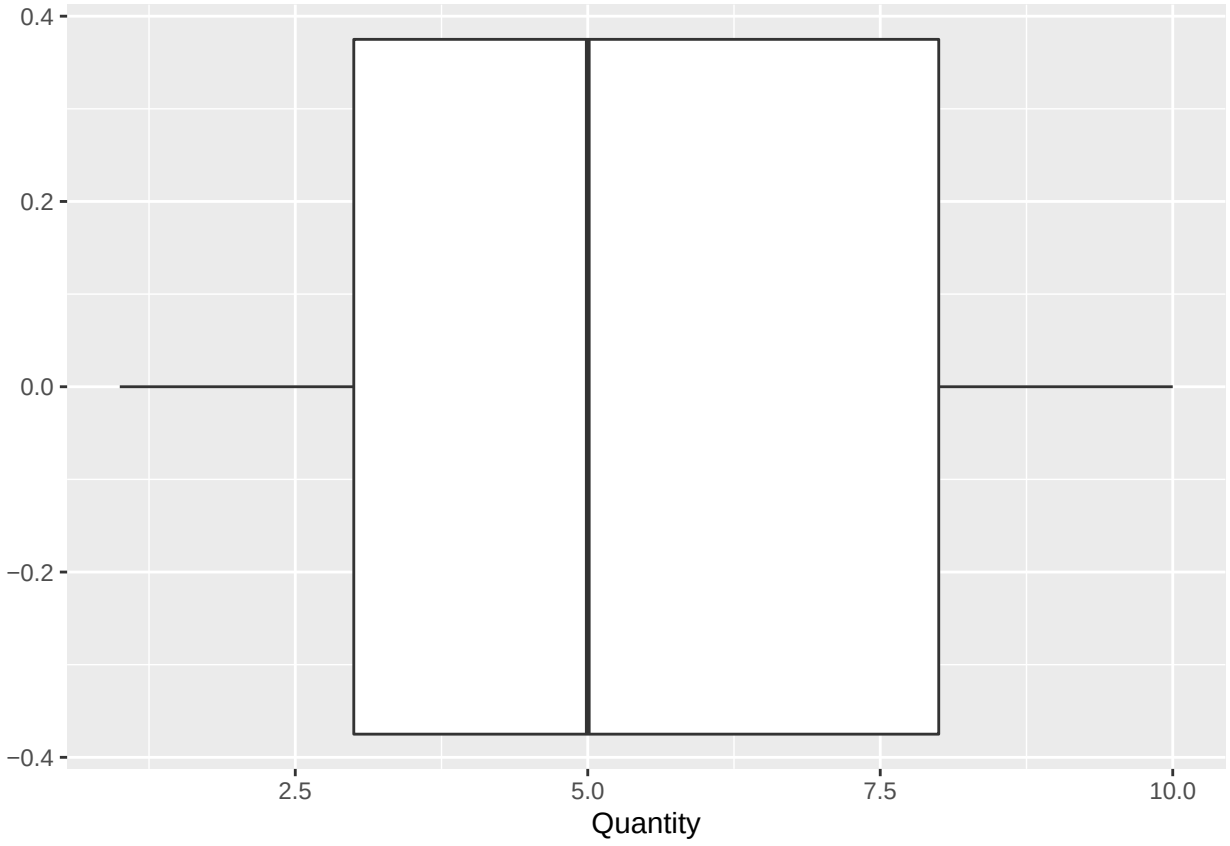
```
# Range
quantity.range <- range(sales$Quantity)
quantity.range
```

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

```
# Quantiles
quantity.quants <- quantile(sales$Quantity)
quantity.quants
```

```
##   0%  25%  50%  75% 100%
##    1    3    5    8   10
```

```
# Visualization
ggplot(sales, aes(Quantity)) +
  geom_boxplot(outlier.colour = "red")
```




```
# mean
tax.mean <- mean(sales$Tax)
tax.mean
```

Tax

```
## [1] 15.37937
```

```
# mode
tax.mode <- mode(sales$Tax)
tax.mode
```

```
## [1] 39.48
```

```
# Median
tax.median <- median(sales$Tax)
tax.median
```

```
## [1] 12.088
```

```
# Standard Deviation
tax.sd <- sd(sales$Tax)
tax.sd
```

```
## [1] 11.70883
```

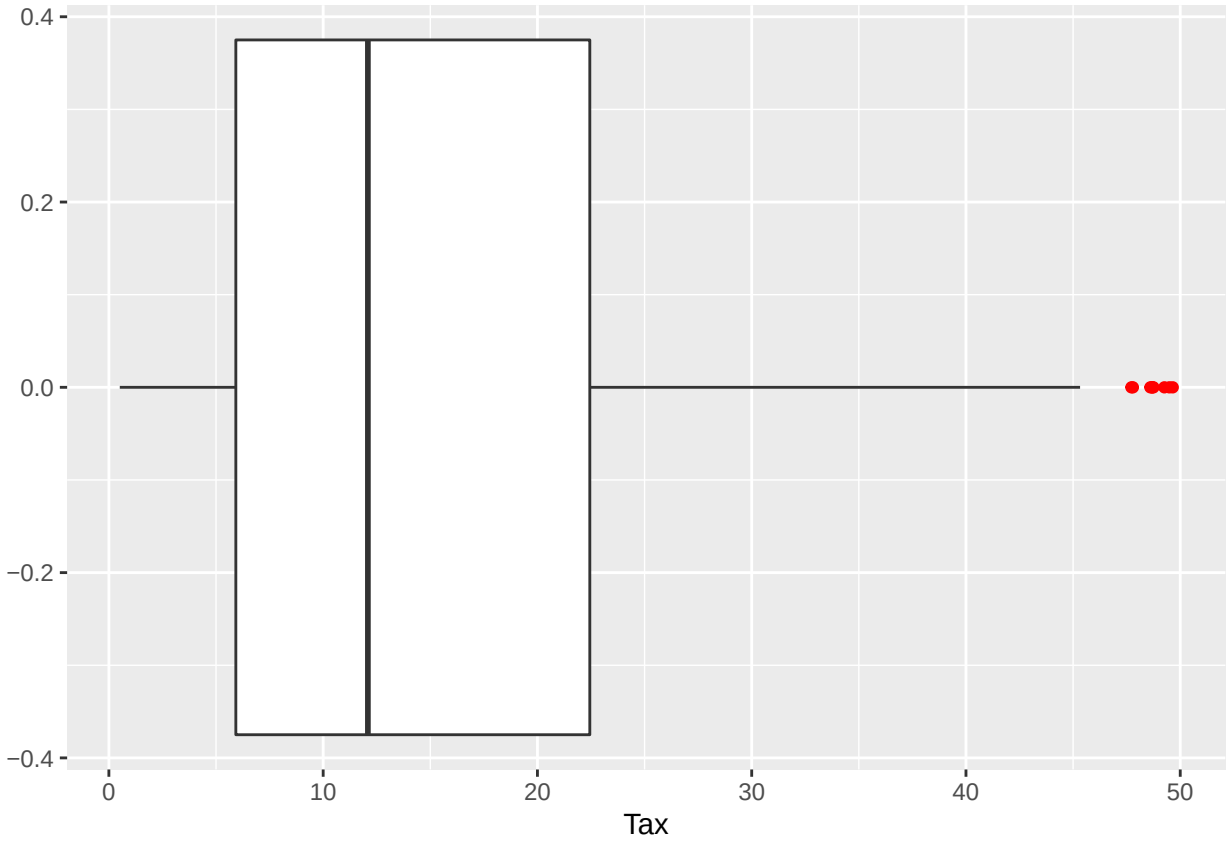
```
# Range
tax.range <- range(sales$Tax)
tax.range
```

```
## [1]  0.5085 49.6500
```

```
# Quantiles
tax.quantiles <- quantile(sales$Tax)
tax.quantiles
```

```
##        0%       25%       50%       75%      100%
##  0.508500  5.924875 12.088000 22.445250 49.650000
```

```
# Visual
ggplot(sales, aes(Tax)) +
  geom_boxplot(outlier.colour = "red")
```

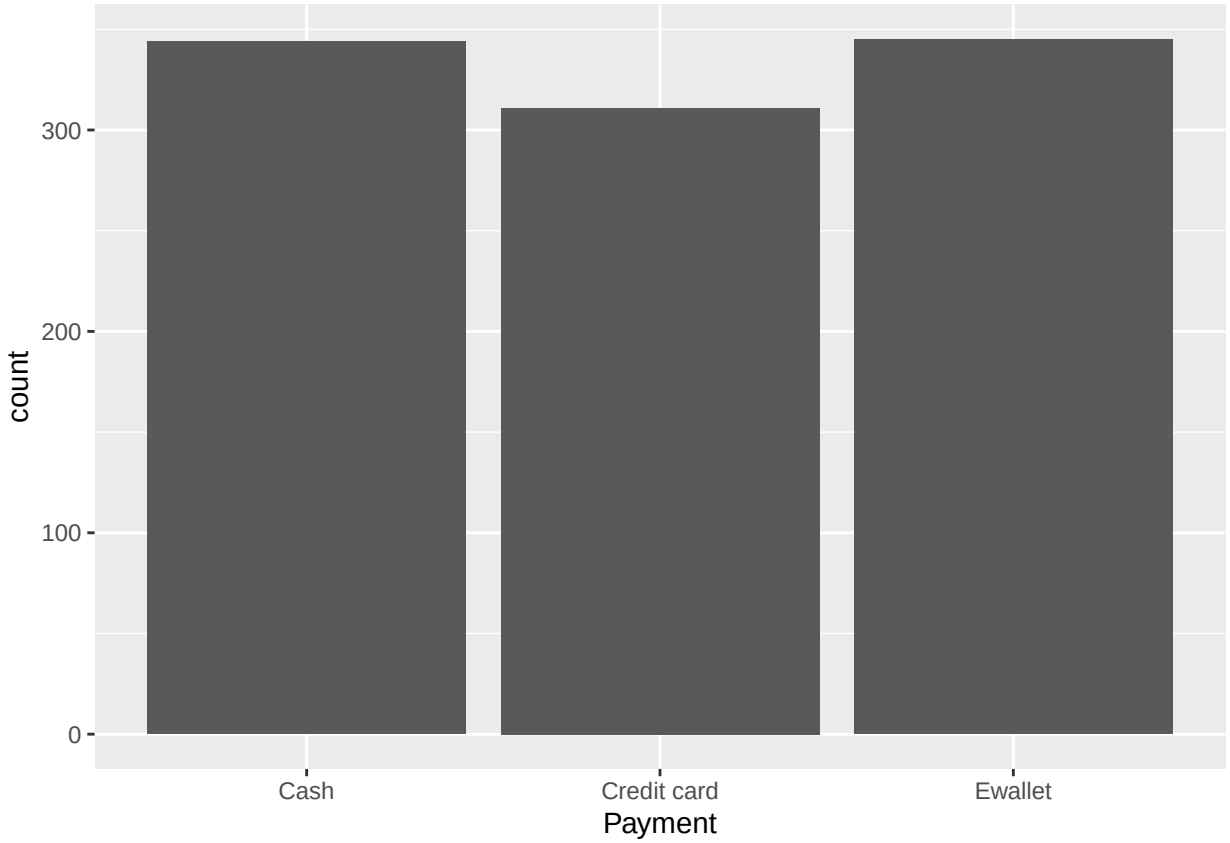


```
# Mode
payment.mode <- mode(sales$Payment)
payment.mode
```

Payment

```
## [1] "Ewallet"
```

```
# visual
ggplot(sales, aes(Payment)) +
  geom_bar(stat="count")
```



There is a fair distribution in the payment variable. However, fewer people tend to pay by Credit Card in these stores

```
# mean
cogs.mean <- mean(sales$cogs)
cogs.mean
```

COGS

```
## [1] 307.5874
```

```
# mode
cogs.mode <- mode(sales$cogs)
cogs.mode
```

```
## [1] 789.6
```

```
# median
cogs.median <- median(sales$cogs)
cogs.median
```

```
## [1] 241.76
```

```
# standard deviation
cogs.sd <- sd(sales$cogs)
cogs.sd
```

```
## [1] 234.1765
```

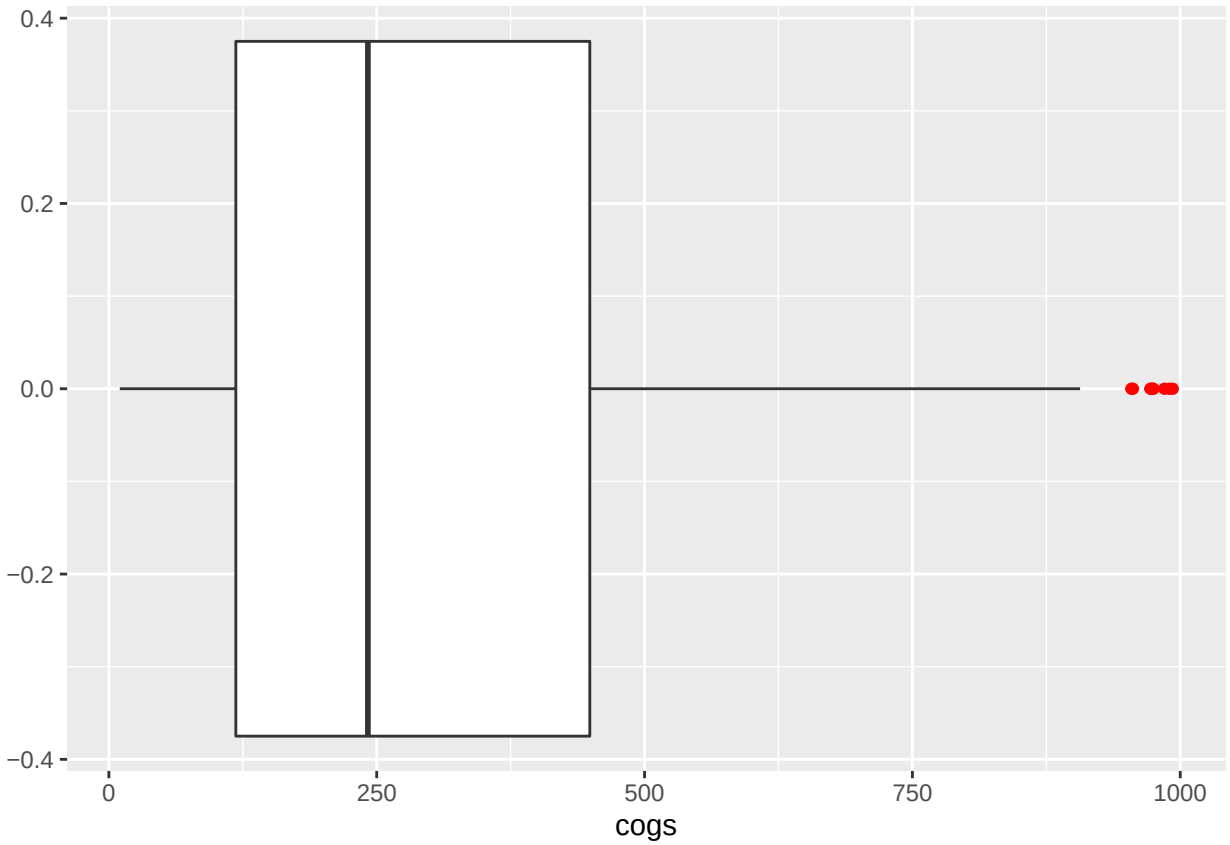
```
# range
cogs.range <- range(sales$cogs)
cogs.range
```

```
## [1]  10.17 993.00
```

```
# quantiles
cogs.quantiles <- quantile(sales$cogs)
cogs.quantiles
```

```
##       0%      25%      50%      75%     100%
##  10.1700 118.4975 241.7600 448.9050 993.0000
```

```
# visual
ggplot(sales, aes(cogs)) +
  geom_boxplot(outlier.colour = "red")
```




```
gross.income <- sales$gross.income
# mean
gross.income.mean <- mean(gross.income)
gross.income.mean
```

Gross Income

```
## [1] 15.37937
```

```
# mode
gross.income.mode <- mode(gross.income)
gross.income.mode
```

```
## [1] 39.48
```

```
# median
gross.income.median <- median(gross.income)
gross.income.median
```

```
## [1] 12.088
```

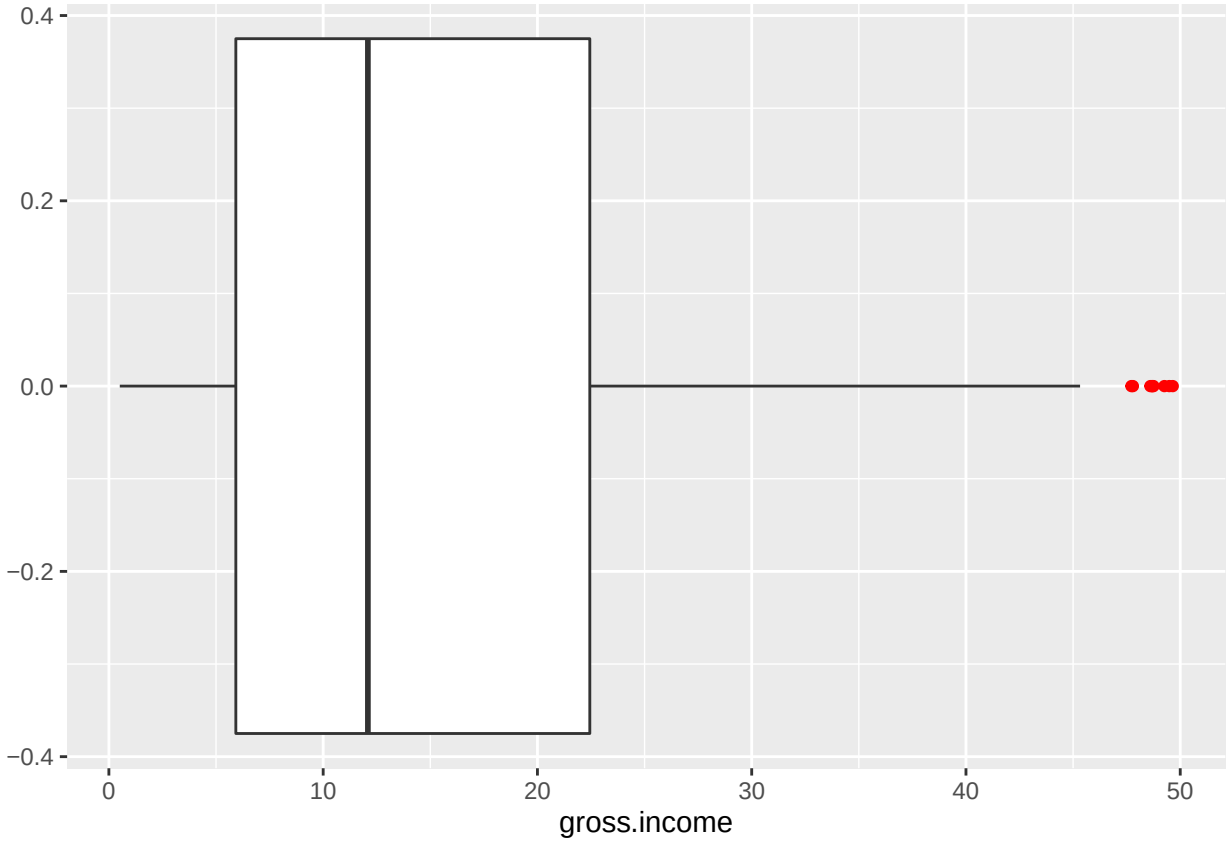
```
# range
gross.income.range <- range(gross.income)
gross.income.range
```

```
## [1]  0.5085 49.6500
```

```
# standard deviation
gross.income.sd <- sd(gross.income)
gross.income.sd
```

```
## [1] 11.70883
```

```
# visual
 ggplot(sales, aes(gross.income)) +
   geom_boxplot(outlier.colour = "red")
```




```
# mean
rate.mean <- mean(sales$Rating)
rate.mean
```

Ratings

```
## [1] 6.9727
```

```
# mode
rate.mode <- mode(sales$Rating)
rate.mode
```

```
## [1] 6
```

```
# median
rate.median <- median(sales$Rating)
rate.median
```

```
## [1] 7
```

```
# standard deviation
rate.sd <- sd(sales$Rating)
rate.sd
```

```
## [1] 1.71858
```

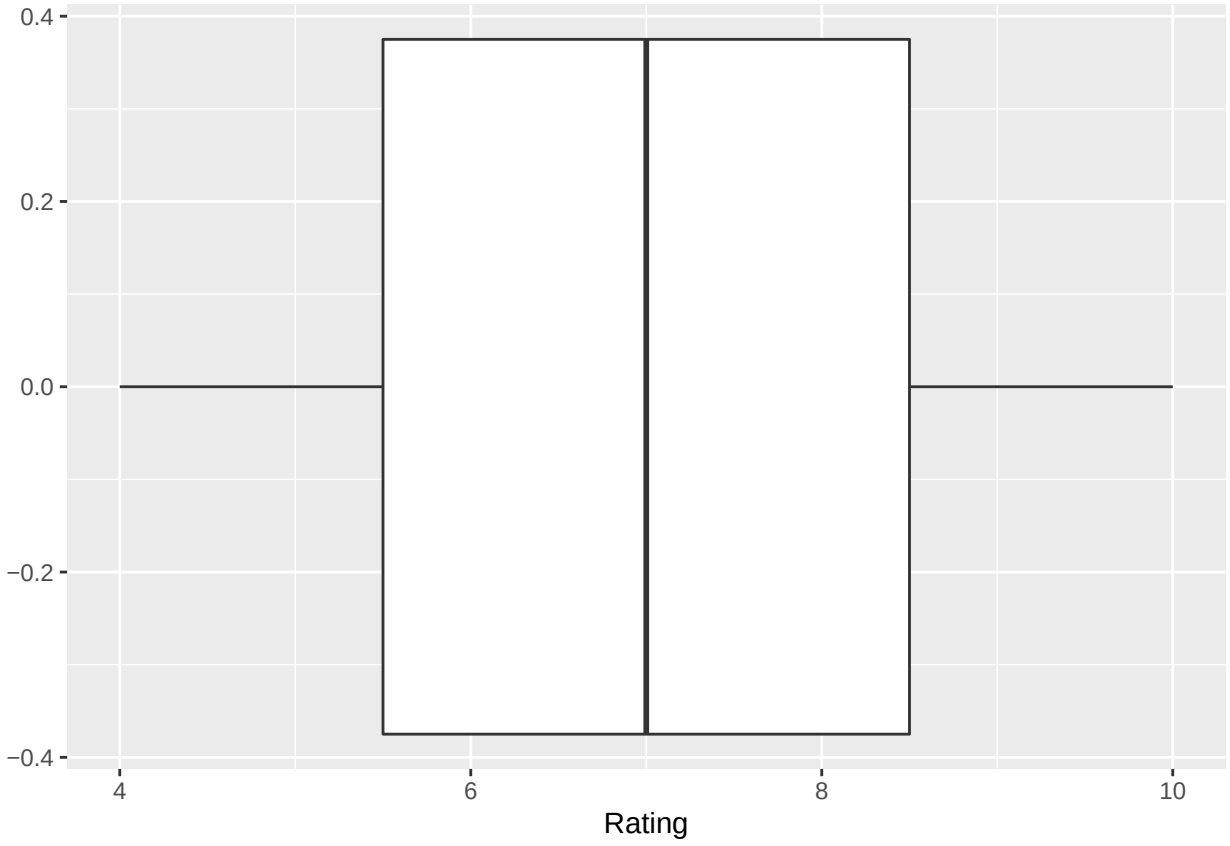
```
# range
rate.range <- range(sales$Rating)
rate.range
```

```
## [1]  4 10
```

```
# quantiles
rate.quantiles <- quantile(sales$Rating)
rate.quantiles
```

```
##   0%  25%  50%  75% 100%
##  4.0  5.5  7.0  8.5 10.0
```

```
# visual
ggplot(sales, aes(Rating)) +
  geom_boxplot(outlier.colour = "red")
```




```
# mean
total.mean <- mean(sales$Total)
total.mean
```

Total

```
## [1] 322.9667
```

```
# median
total.median <- median(sales$Total)
total.median
```

```
## [1] 253.848
```

```
# mode
total.mode <- mode(sales$Total)
total.mode
```

```
## [1] 829.08
```

```r
# standard deviation
total.sd <- sd(sales$Total)
total.sd
```

```
## [1] 245.8853
```

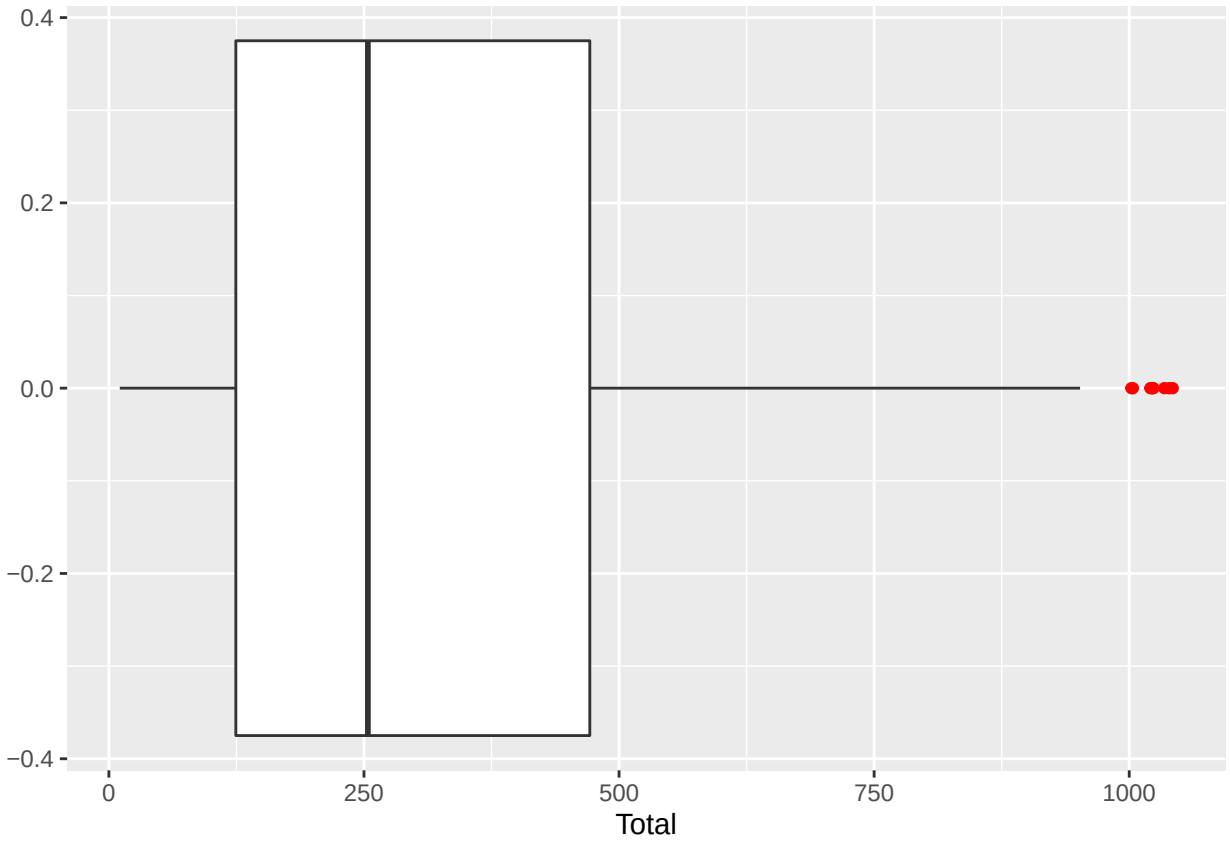
```r
# range
total.range <- range(sales$Total)
total.range
```

```
## [1]   10.6785 1042.6500
```

```r
# quantiles
total.quantiles <- quantile(sales$Total)
total.quantiles
```

```
##        0%       25%       50%       75%      100%
##   10.6785  124.4224  253.8480  471.3502 1042.6500
```

```r
# visual
ggplot(sales, aes(Total)) +
  geom_boxplot(outlier.colour = "red" )
```



```
library(tidyr)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

## The following object is masked from 'package:tidyr':
##
##     extract
```

```
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
```

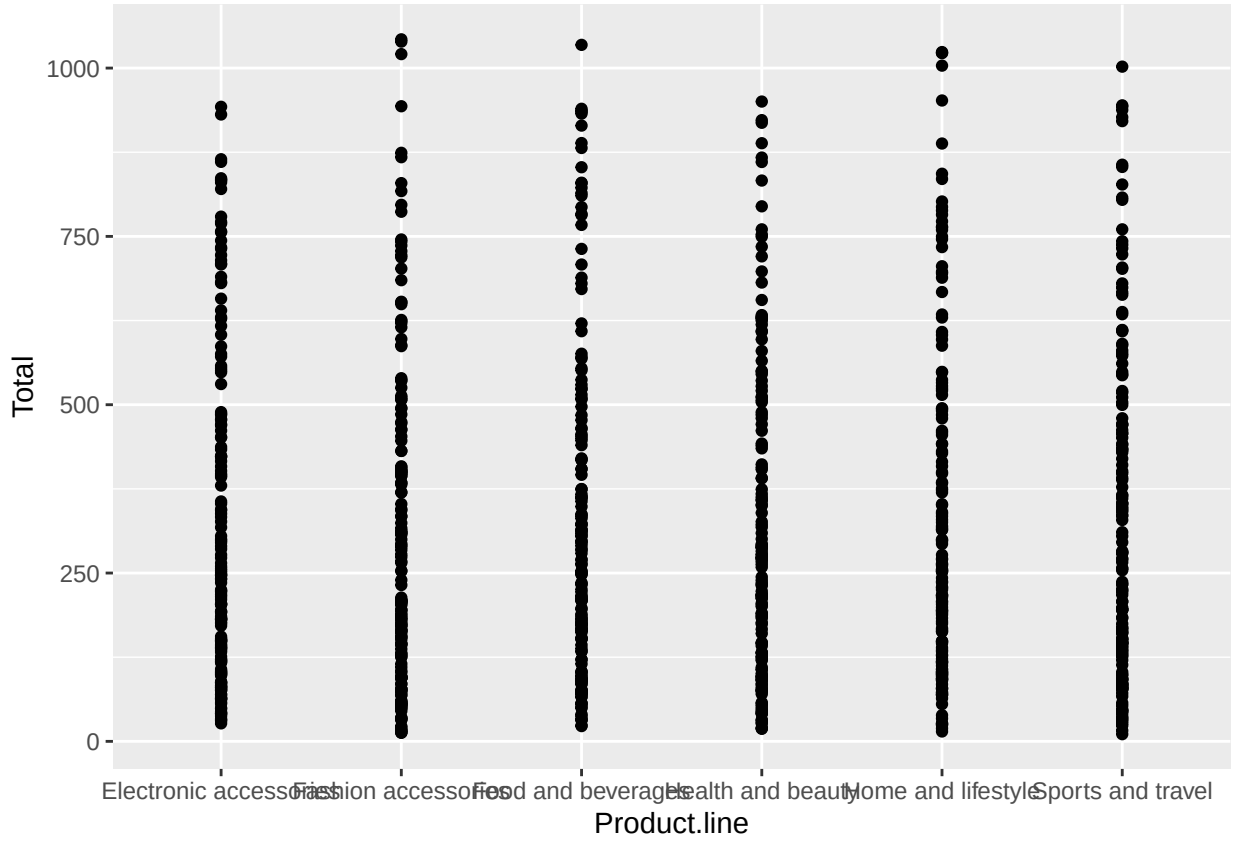
```
library(psych)
```

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

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

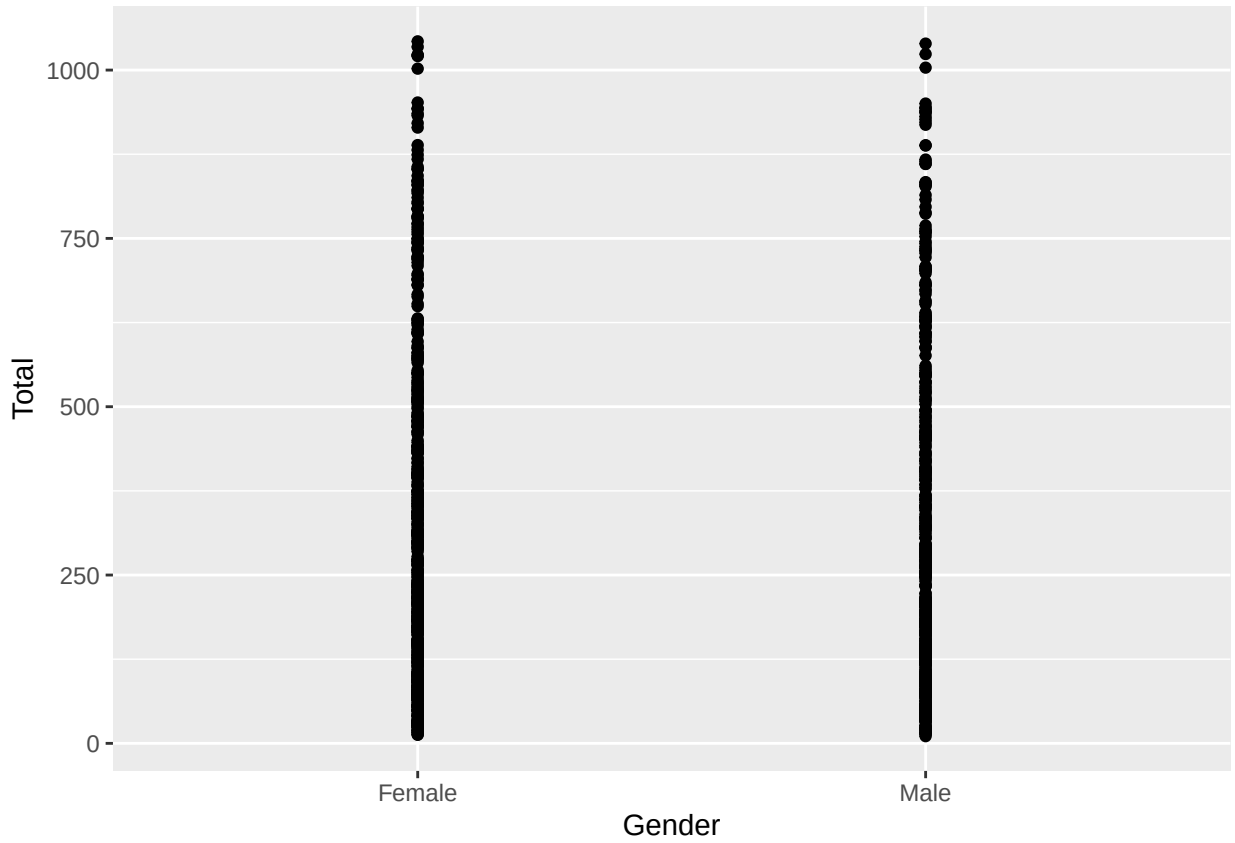
Bivariate Analysis

```
ggplot(sales, aes(x=Product.line, y=Total)) +
  geom_point()
```



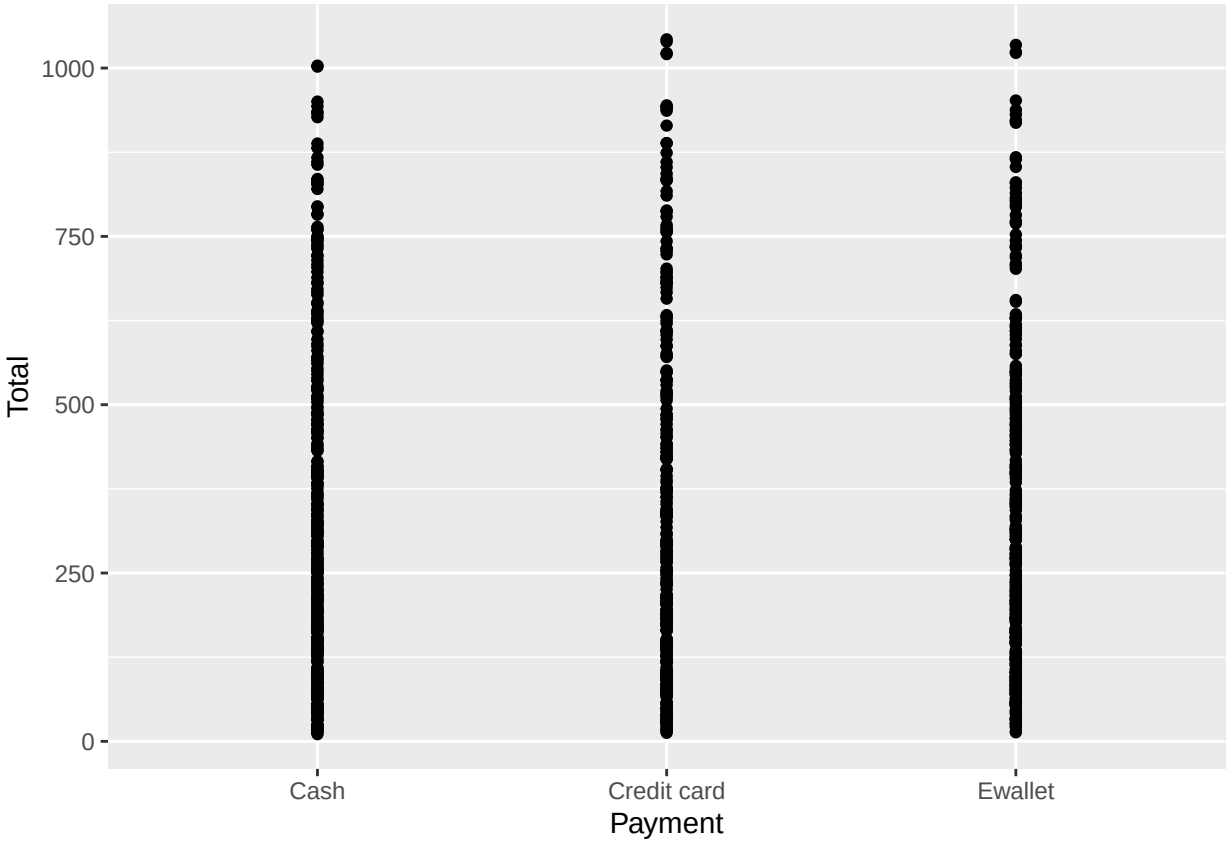
Fashion Accessories have the highest Total prices while health and beauty products have a relatively lower price.

```
ggplot(sales ,aes(Gender, Total)) +
  geom_point()
```



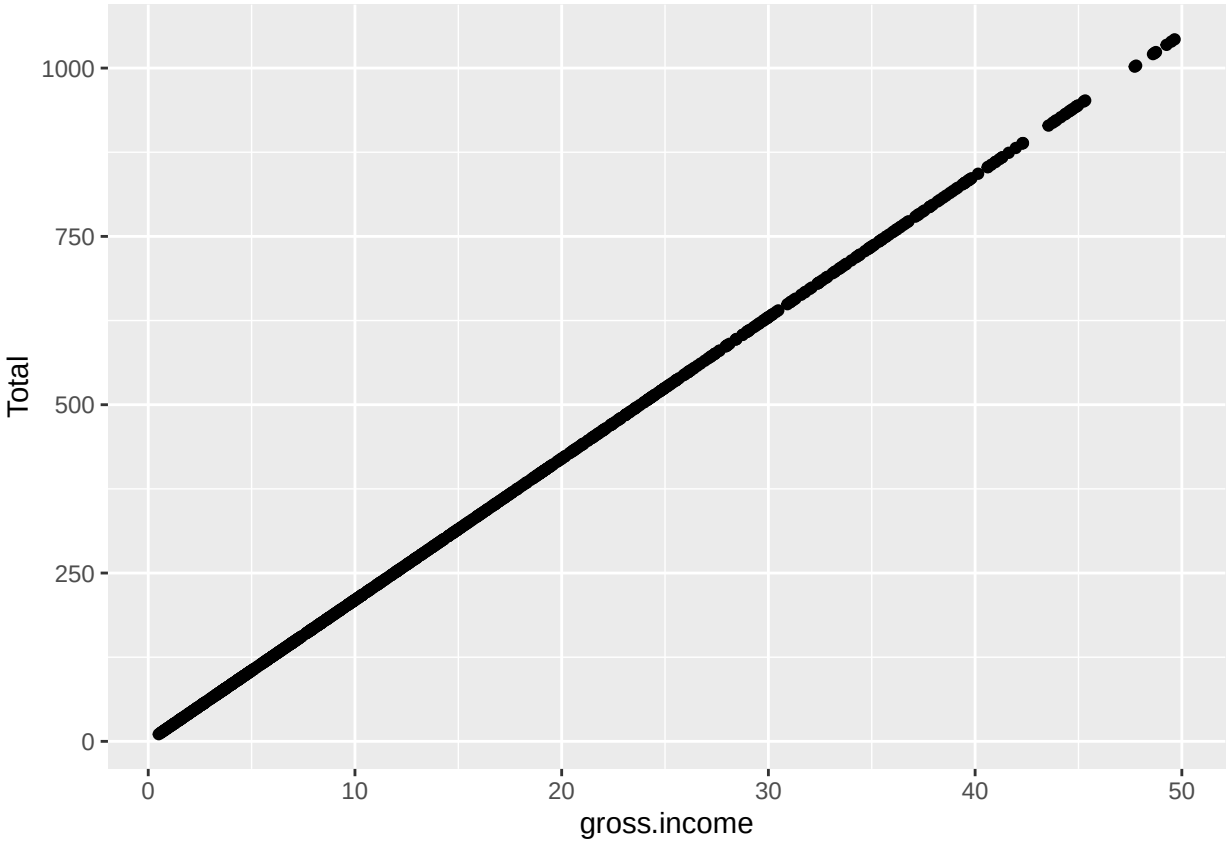
Total Price is equally distributed in terms of gender

```
ggplot(sales, aes(Payment, Total)) +
  geom_point()
```



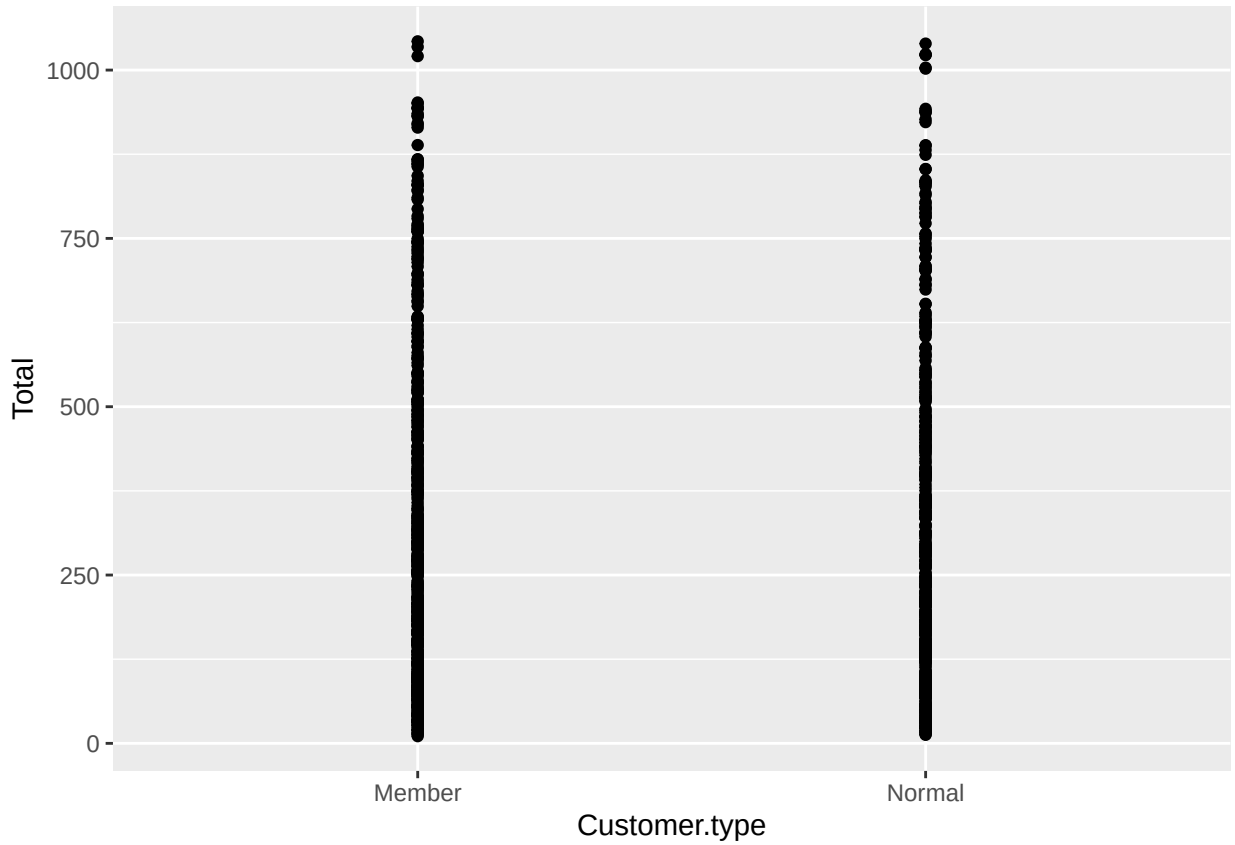
The payment methods are nearly identical for the total prices of items at checkouts with some more expensive ones being attributed with Credit card payments.

```
ggplot(sales, aes(gross.income, Total)) +
  geom_point()
```



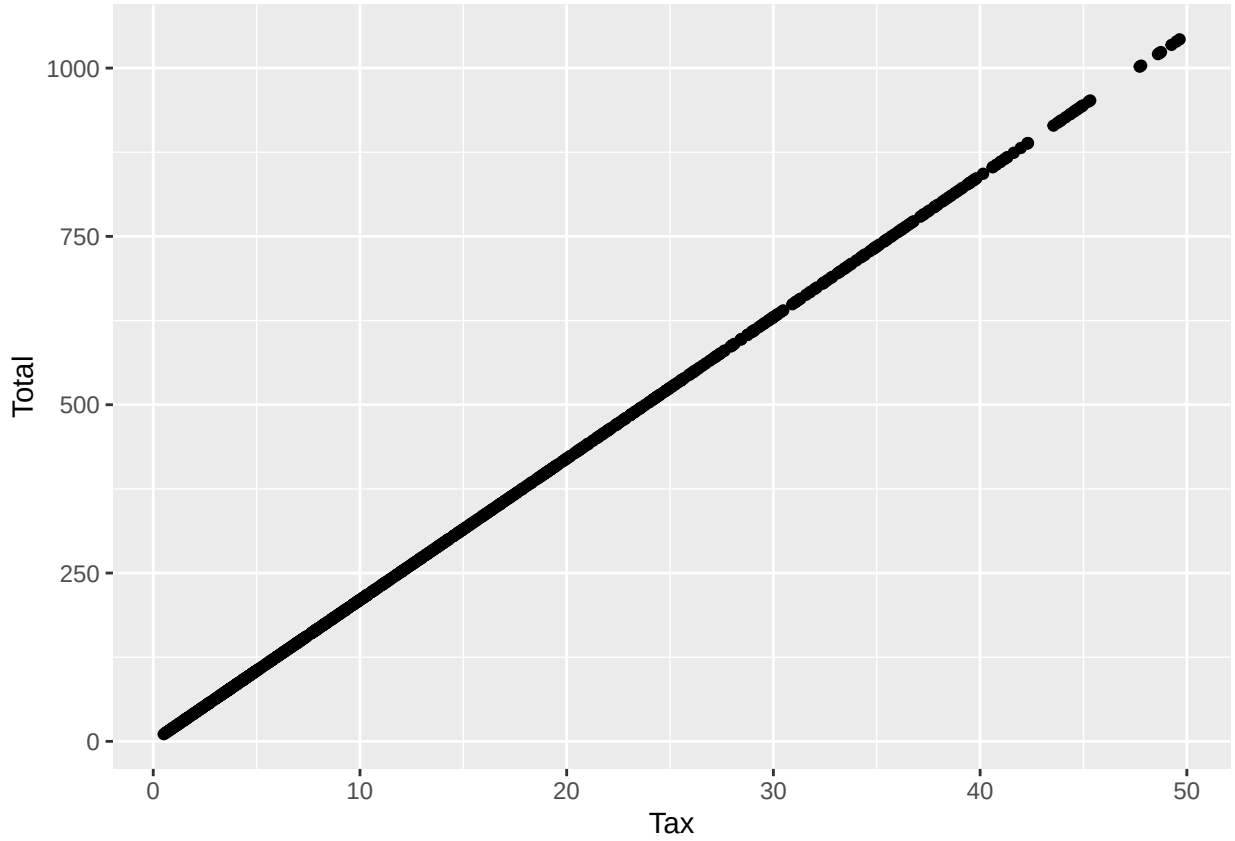
As expected, there is a perfect positive linear relationship with how much the total is at checkout with the consumers gross income.

```
ggplot(sales, aes(Customer.type , Total)) +
  geom_point()
```



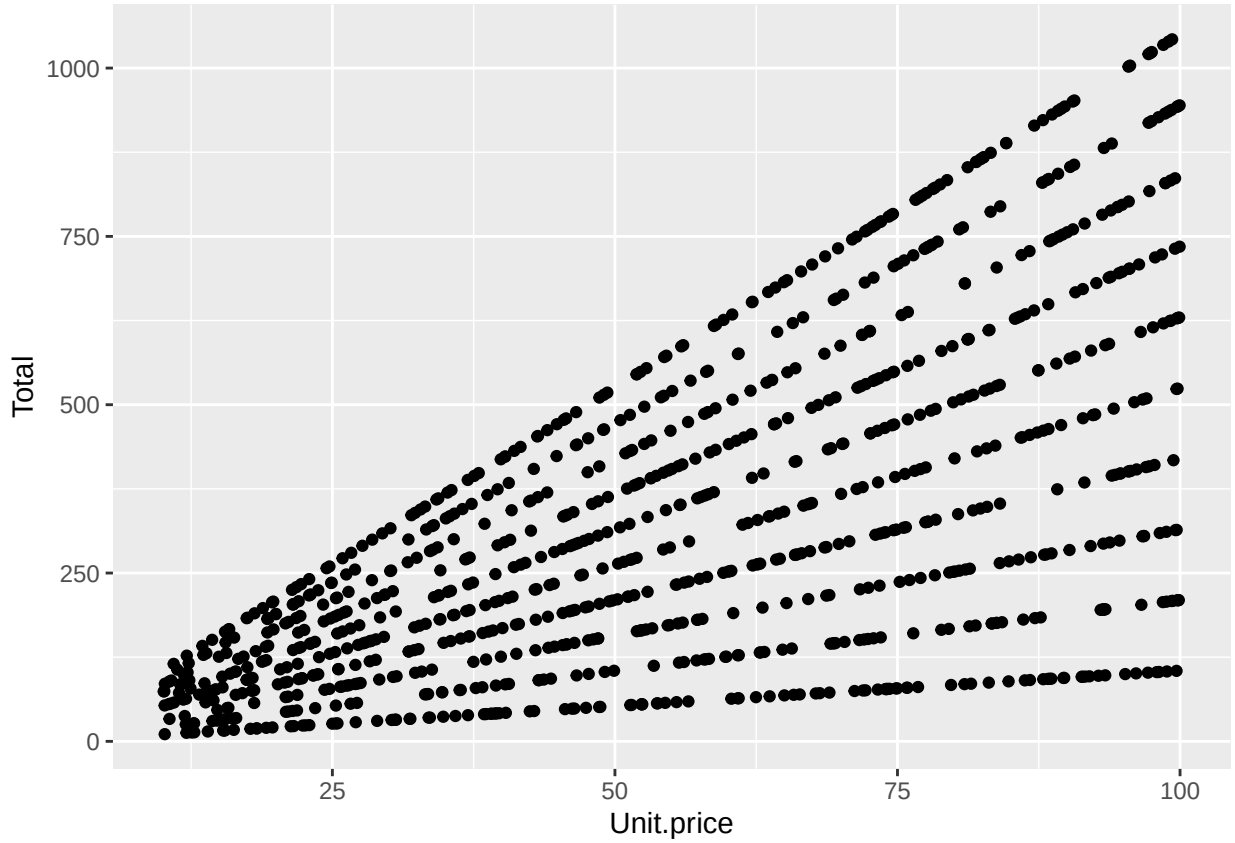
Members and non members have a nearly equal distribution in expenditure with Members having no visible breaks in prices.

```
ggplot(sales, aes(Tax, Total)) +
  geom_point()
```



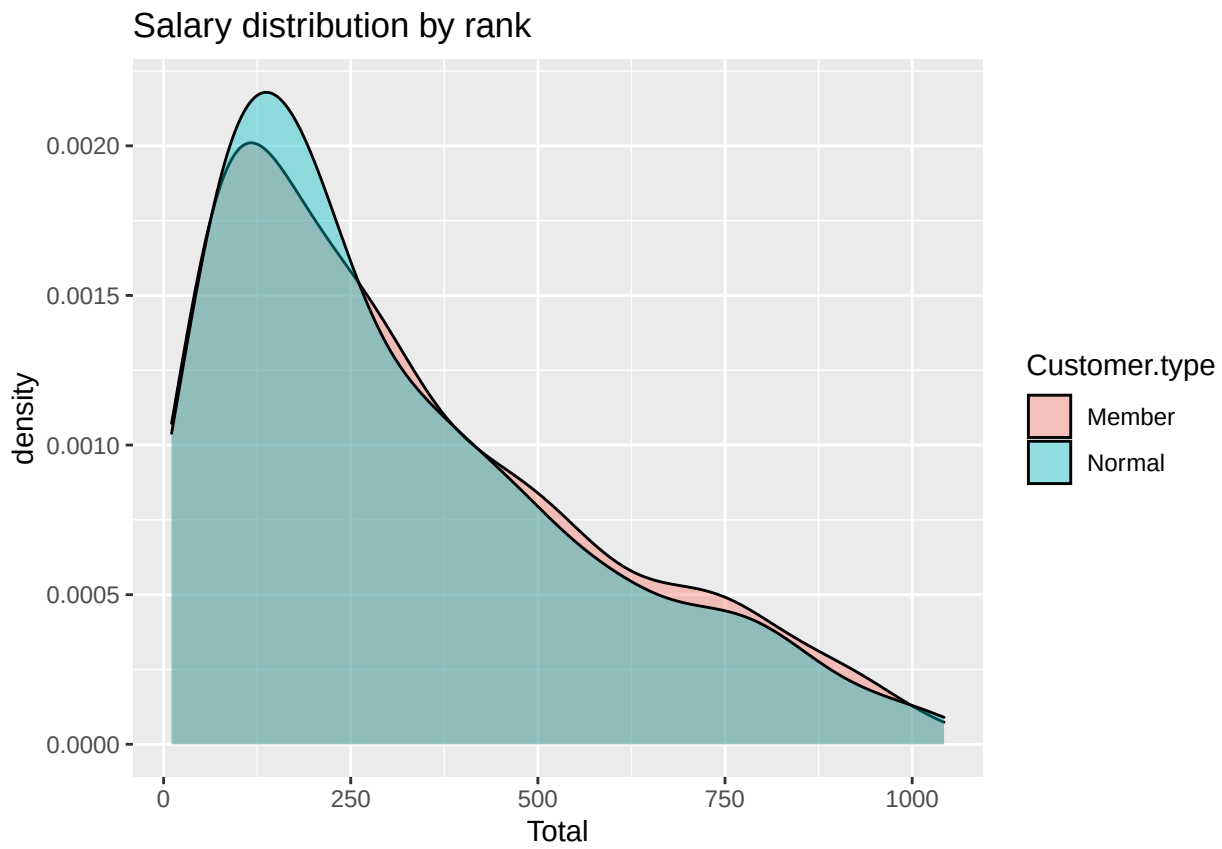
There is a direct linear relationship between tax and total price. As expected, the higher the tax on items, the more they cost.

```
ggplot(sales, aes(Unit.price, Total)) +
  geom_point()
```



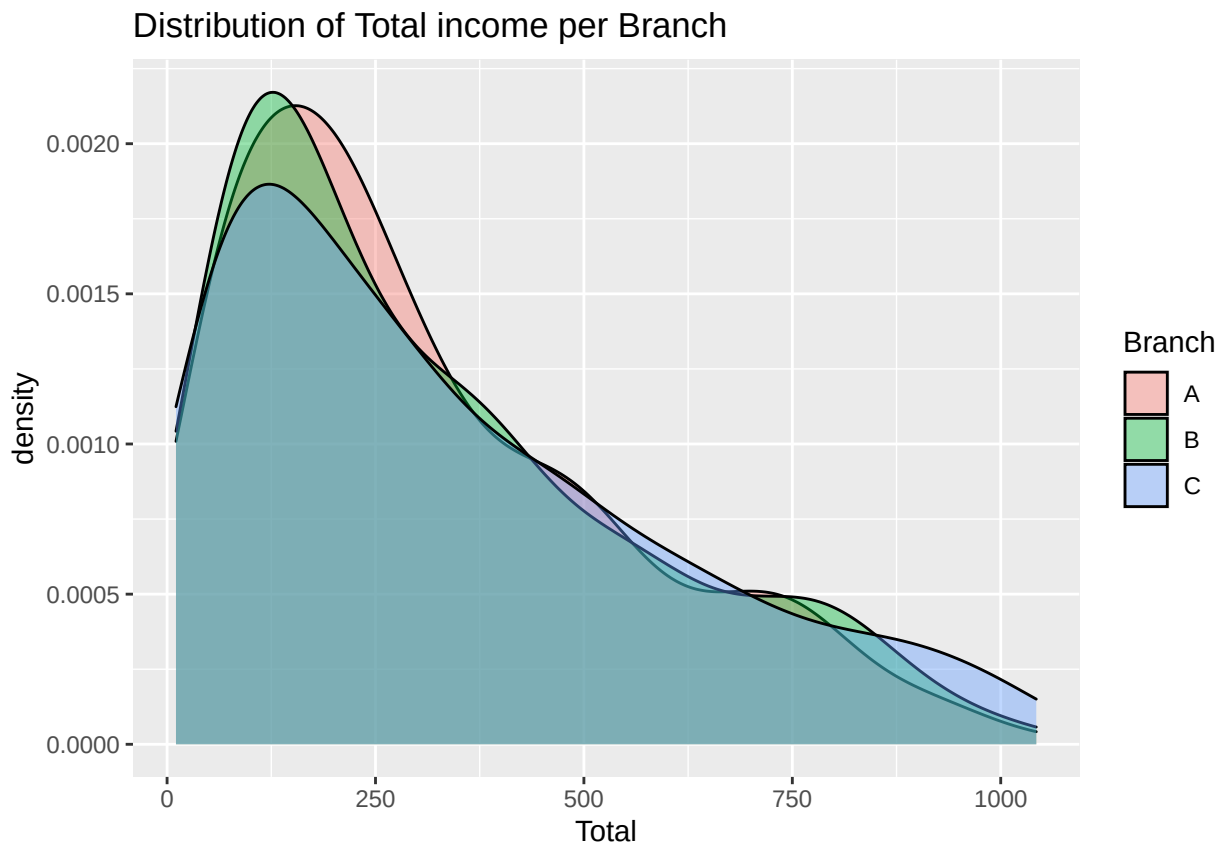
There are several positive linear relationships with the Unit Price variable: the higher it is the higher the total price is. More data would be needed to explain the different lines considering they represent outside factors that influence the relationship. A good example would be the type of products being of different types.

```
#Salary distribution by rank
ggplot(sales,
aes(x = Total,
fill = Customer.type)) +
geom_density(alpha = 0.4) +
labs(title = "Salary distribution by rank")
```



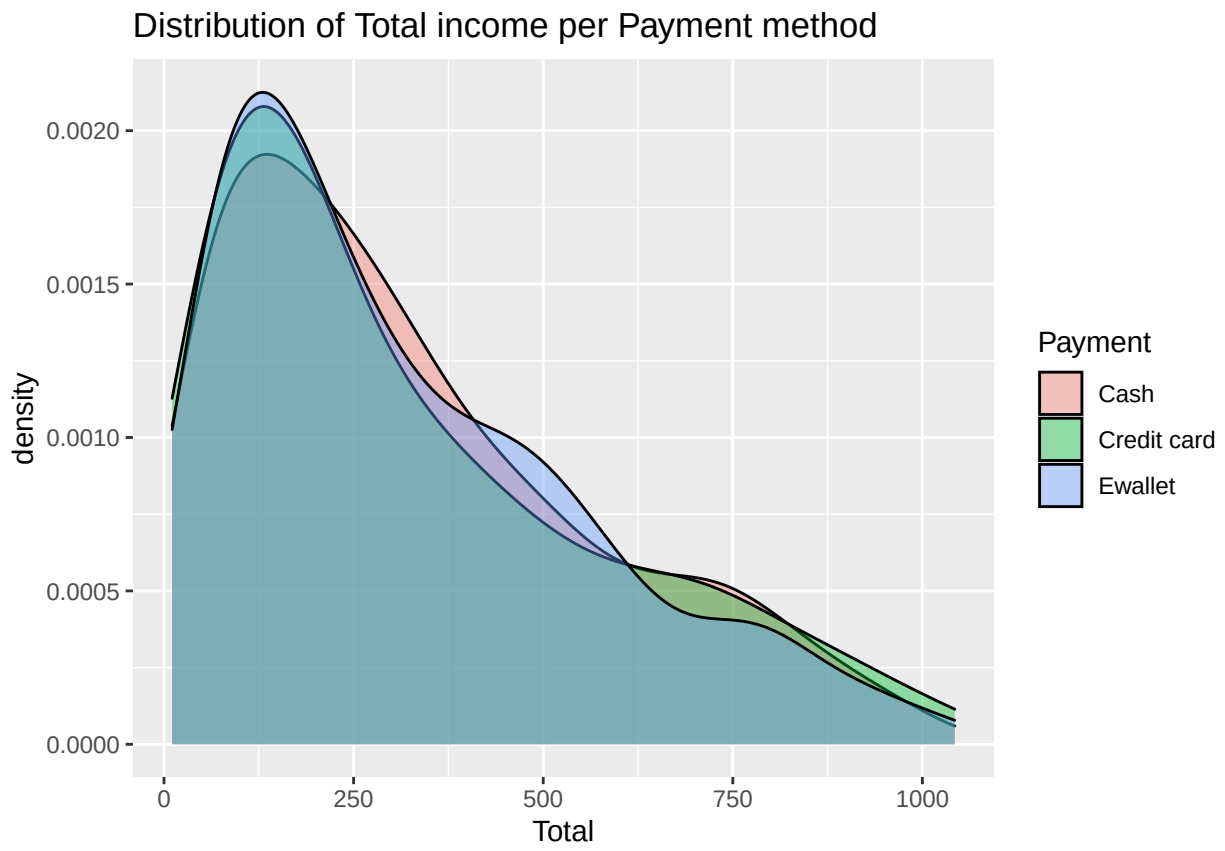
Normal customers seem to have a greater influence on total than members.

```
#Distribution of Total income per Branch
ggplot(sales,
aes(x = Total,
fill = Branch)) +
geom_density(alpha = 0.4) +
labs(title = "Distribution of Total income per Branch")
```

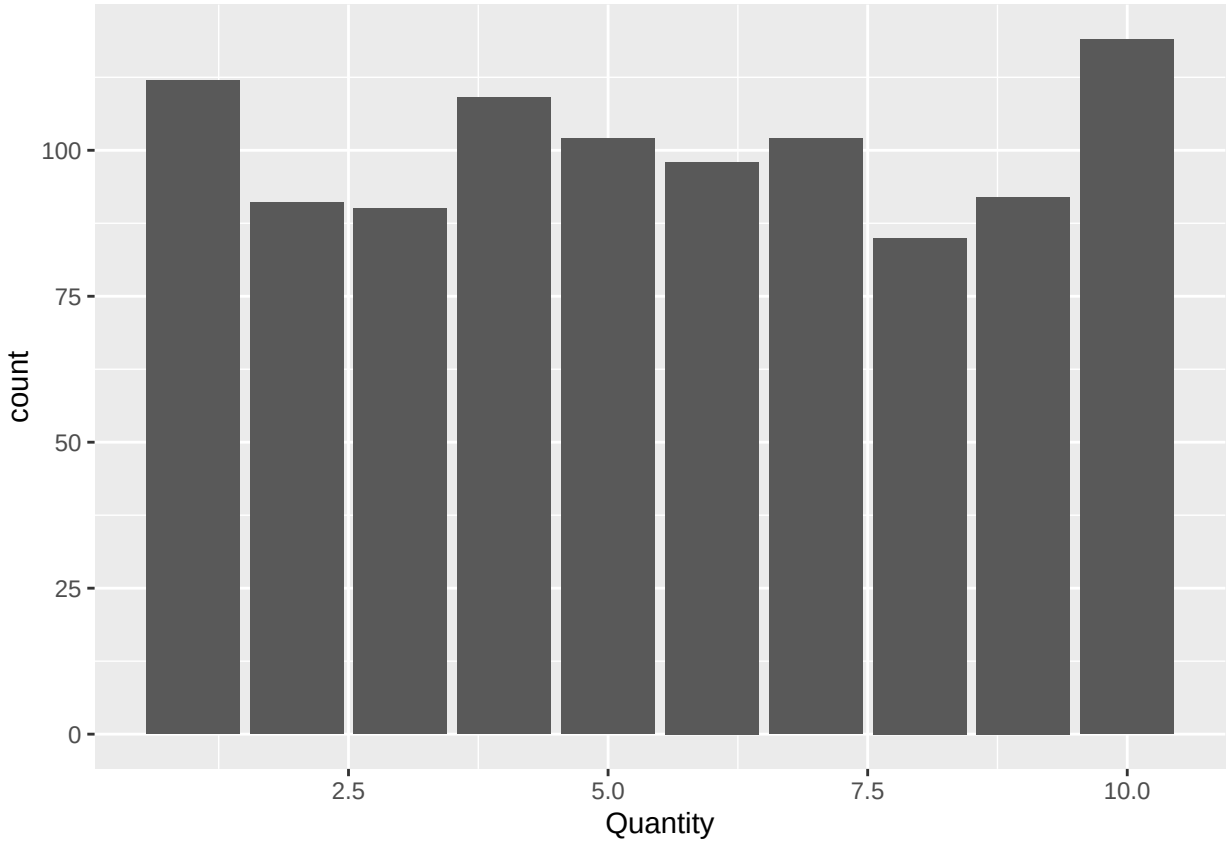


Branch A contributes more to total and Branch C contributes the leastBranch A contributes more to total and Branch C contributes the least

```
#Distribution of Total per Payment method
ggplot(sales,
aes(x = Total,
fill = Payment)) +
geom_density(alpha = 0.4) +
labs(title = "Distribution of Total income per Payment method")
```




```
#What quantity was mostly purchased in the store
ggplot(sales, aes(x = Quantity)) +
geom_bar()
```



Most people purchased 10items, followed by those who purchased 1 item

```
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
#Get the correlation matrix
nums <- subset(sales, select = -c(Branch, Customer.type,Gender,Product.line,Date, Time, Payment))
res = cor(nums)
#Plotting a correlation plot
corrplot(res, method="color",addCoef.col = "black",
tl.col="black", tl.srt=45)
```

There is perfect correlation between Tax, Cogs and gross income. There is also high correlation between Unit Price and Tax,cogs and gross.income and Total.

Dimensionality Reduction

PCA

Feature Engineering

- All variables to be used for dimensionality reduction should be numerical variables, hence we will convert our factor categories to numerics.We will also drop the date and time columns.

```
#First we will make a copy of our sales dataset for future use
data <- sales
#Dropping columns for date and time
data <- subset(data, select = -c(Date, Time))
head(data)
```

```
##   Branch Customer.type Gender           Product.line Unit.price Quantity
## 1      A        Member Female      Health and beauty      74.69        7
## 2      C        Normal Female Electronic accessories      15.28        5
## 3      A        Normal   Male     Home and lifestyle      46.33        7
## 4      A        Member   Male      Health and beauty      58.22        8
## 5      A        Normal   Male      Sports and travel      86.31        7
## 6      C        Normal   Male Electronic accessories      85.39        7
```

```
##       Tax     Payment   cogs gross.income Rating    Total
## 1 26.1415     Ewallet 522.83      26.1415    9.1 548.9715
## 2  3.8200        Cash  76.40       3.8200    9.6  80.2200
## 3 16.2155 Credit card 324.31      16.2155    7.4 340.5255
## 4 23.2880     Ewallet 465.76      23.2880    8.4 489.0480
## 5 30.2085     Ewallet 604.17      30.2085    5.3 634.3785
## 6 29.8865     Ewallet 597.73      29.8865    4.1 627.6165
```

```
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)
```

```
#Converting factor columns to numeric
data$Branch <- as.integer(data$Branch)
data$Customer.type <- as.numeric(data$Customer.type)
data$Gender <- as.numeric(data$Gender)
data$Product.line <- as.numeric(data$Product.line)
data$Payment <- as.numeric(data$Payment)
data$Quantity <- as.numeric(data$Quantity)
head(data)
```

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

```
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 4.1.1
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
#Performing pca
data.pca <- prcomp(data[,c(1:11)],center = TRUE,scale. = TRUE)
summary(data.pca)
```

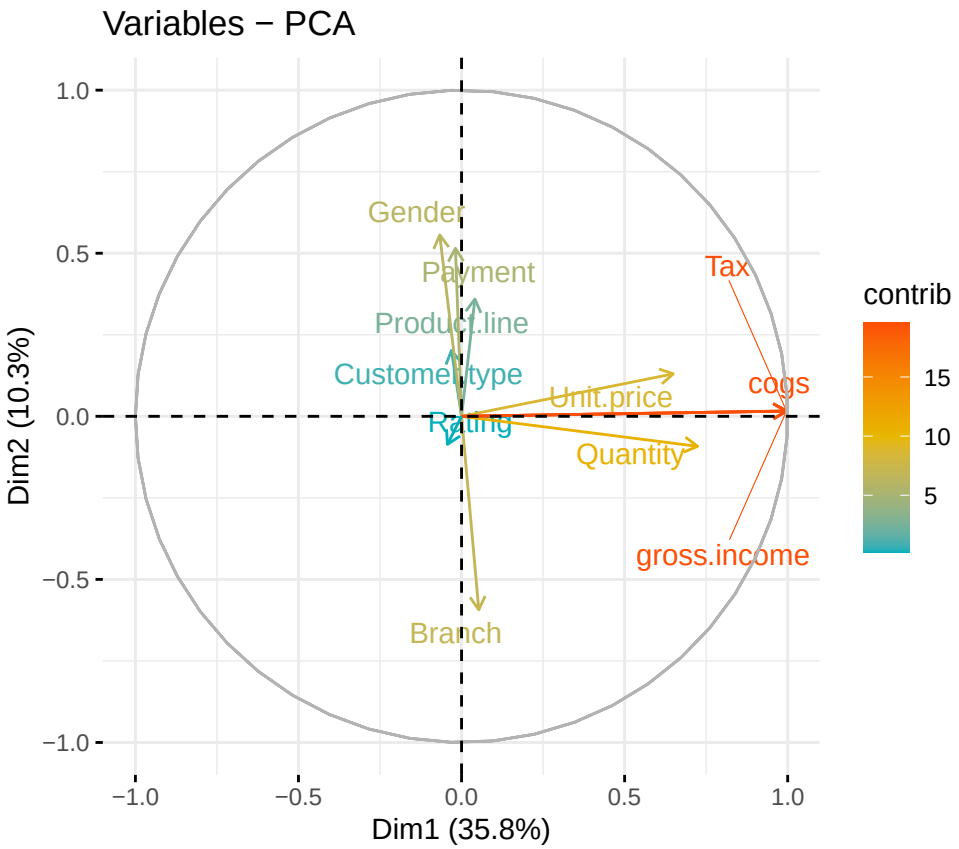
```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5    PC6     PC7
## Standard deviation     1.9836 1.0631 1.03159 1.00991 0.99289 0.9771 0.96270
```

```
## Proportion of Variance 0.3577 0.1027 0.09674 0.09272 0.08962 0.0868 0.08425
## Cumulative Proportion  0.3577 0.4604 0.55719 0.64991 0.73953 0.8263 0.91058
##                            PC8     PC9      PC10      PC11
## Standard deviation     0.94823 0.29062 2.736e-16 1.109e-16
## Proportion of Variance 0.08174 0.00768 0.000e+00 0.000e+00
## Cumulative Proportion  0.99232 1.00000 1.000e+00 1.000e+00
```

```
str(data.pca)
```

```
## List of 5
##  $ sdev    : num [1:11] 1.984 1.063 1.032 1.01 0.993 ...
##  $ rotation: num [1:11, 1:11] 0.0267 -0.0155 -0.0338 0.0206 0.3273 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:11] "Branch" "Customer.type" "Gender" "Product.line" ...
##   .. ..$ : chr [1:11] "PC1" "PC2" "PC3" "PC4" ...
##  $ center  : Named num [1:11] 1.99 1.5 1.5 3.45 55.67 ...
##   ..- attr(*, "names")= chr [1:11] "Branch" "Customer.type" "Gender" "Product.line" ...
##  $ scale   : Named num [1:11] 0.818 0.5 0.5 1.715 26.495 ...
##   ..- attr(*, "names")= chr [1:11] "Branch" "Customer.type" "Gender" "Product.line" ...
##  $ x       : num [1:1000, 1:11] 1.79 -2.05 0.11 1.29 2.43 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:1000] "1" "2" "3" "4" ...
##   .. ..$ : chr [1:11] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"
```

```
#Graph of variables
fviz_pca_var(data.pca,
col.var = "contrib", # Color by contributions to the PC
gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
repel = TRUE # Avoid text overlapping
)
```



Gross income, Tax and cogs contribute highly to the first PC whereas Gender, Payment mostly contribute to the second PC

```
# Eigenvalues
eig.val <- get_eigenvalue(data.pca)
eig.val
```

```
##          eigenvalue variance.percent cumulative.variance.percent
## Dim.1  3.934732e+00     3.577029e+01                     35.77029
## Dim.2  1.130132e+00     1.027393e+01                     46.04423
## Dim.3  1.064187e+00     9.674426e+00                     55.71865
## Dim.4  1.019927e+00     9.272061e+00                     64.99071
## Dim.5  9.858334e-01     8.962122e+00                     73.95283
## Dim.6  9.548085e-01     8.680077e+00                     82.63291
## Dim.7  9.267825e-01     8.425296e+00                     91.05821
## Dim.8  8.991345e-01     8.173950e+00                     99.23216
## Dim.9  8.446266e-02     7.678424e-01                    100.00000
## Dim.10 7.484119e-32     6.803745e-31                    100.00000
## Dim.11 1.229717e-32     1.117924e-31                    100.00000
```

We have obtained 11 principal components. Our first PC, PC1 explains 35.7% Variation, our second, PC2 explains 46%. The first 8 PCs gives us a variability proportion of upto 100%.

Feature Selection

- Using Filter Method Using the filter method, we will check for correlation between variables. We will then remove variables that are highly correlated as that is a sign of redundancy.

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.1
```

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

```
## Warning: package 'lattice' was built under R version 4.1.1
```

```
#Separating target variable with independent variables
df <- data[-12]
# Calculating the correlation matrix
correlationMatrix <- cor(df)
# Find attributes that are highly correlated
highlyCorrelated <- findCorrelation(correlationMatrix, cutoff= 0.75)
# Highly correlated attributes
highlyCorrelated
```

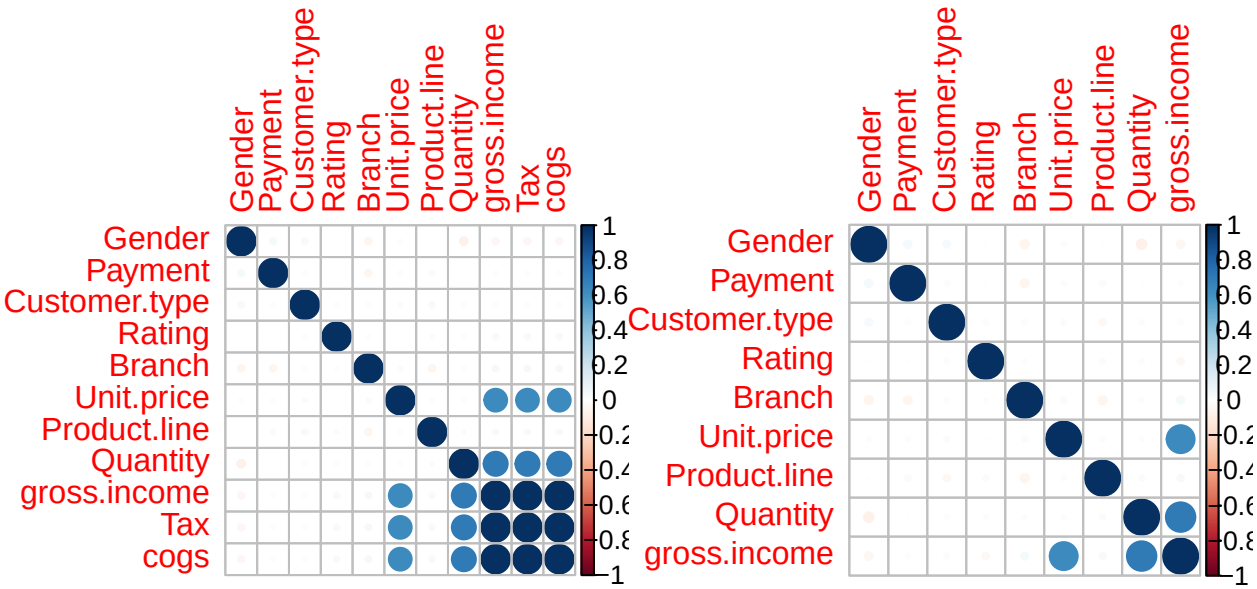
```
## [1] 7 9
```

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

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

Tax and Cogs are highly correlated.

```
# Removing the highly correlated features
df.feat<-df[-highlyCorrelated]
# Performing a graphical comparison
par(mfrow = c(1, 2))
corrplot(correlationMatrix, order = "hclust")
corrplot(cor(df.feat), order = "hclust")
```



Conclusion

The following features will be used for analysis:

- Gender
- Payment
- Customer type
- Rating
- Branch
- Unit price
- Product line
- Quantity
- Gross Income