Assignment 2

Dixitha Kasturi

2/8/2022

Loading Data:

```
setwd('C:/Users/kastu/Desktop/Syracuse/Spring22/IST707-AML/Week2/Assignment')
library(tidyverse)
```

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

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

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

```
market <- read_csv('supermarket_sales.csv')
```

```
## Rows: 1000 Columns: 17

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (8): Invoice ID, Branch, City, Customer type, Gender, Product line, Dat...
## dbl  (8): Unit price, Quantity, Tax 5%, Total, cogs, gross margin percentage...
## time (1): Time

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dim(market)
```

```
## [1] 1000   17
```

The supermarket dataset has 1000 records of purchases with 18 different attributes for each purchase. We will be exploring some of these attributes in detail to understand and get insights from the data that could be further used to understand any underlying conditions or situations that are resulting in the numbers that are reflected.

Exploratory Data Analysis:

Overall it is a clean dataset with not a lot of transformations to do.

```
str(market)
```

```
## spec_tbl_df [1,000 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Invoice ID            : chr [1:1000] "750-67-8428" "226-31-3081" "631-41-3108" "123-19-1176" ...
##  $ Branch                : chr [1:1000] "A" "C" "A" "A" ...
##  $ City                  : chr [1:1000] "Yangon" "Naypyitaw" "Yangon" "Yangon" ...
##  $ Customer type         : chr [1:1000] "Member" "Normal" "Normal" "Member" ...
##  $ Gender                : chr [1:1000] "Female" "Female" "Male" "Male" ...
##  $ Product line          : chr [1:1000] "Health and beauty" "Electronic accessories" "Home and lifes
##  $ Unit price            : num [1:1000] 74.7 15.3 46.3 58.2 86.3 ...
##  $ Quantity              : num [1:1000] 7 5 7 8 7 7 6 10 2 3 ...
##  $ Tax 5%                : num [1:1000] 26.14 3.82 16.22 23.29 30.21 ...
##  $ Total                 : num [1:1000] 549 80.2 340.5 489 634.4 ...
##  $ Date                  : chr [1:1000] "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ Time                  : 'hms' num [1:1000] 13:08:00 10:29:00 13:23:00 20:33:00 ...
##   ..- attr(*, "units")= chr "secs"
##  $ Payment               : chr [1:1000] "Ewallet" "Cash" "Credit card" "Ewallet" ...
##  $ cogs                  : num [1:1000] 522.8 76.4 324.3 465.8 604.2 ...
##  $ gross margin percentage: num [1:1000] 4.76 4.76 4.76 4.76 4.76 ...
##  $ gross income          : num [1:1000] 26.14 3.82 16.22 23.29 30.21 ...
##  $ Rating                : num [1:1000] 9.1 9.6 7.4 8.4 5.3 4.1 5.8 8 7.2 5.9 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   `Invoice ID` = col_character(),
##   ..   Branch = col_character(),
##   ..   City = col_character(),
##   ..   `Customer type` = col_character(),
##   ..   Gender = col_character(),
##   ..   `Product line` = col_character(),
##   ..   `Unit price` = col_double(),
##   ..   Quantity = col_double(),
##   ..   `Tax 5%` = col_double(),
##   ..   Total = col_double(),
##   ..   Date = col_character(),
##   ..   Time = col_time(format = ""),
##   ..   Payment = col_character(),
##   ..   cogs = col_double(),
##   ..   `gross margin percentage` = col_double(),
##   ..   `gross income` = col_double(),
##   ..   Rating = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

- We don't necessarily see the need to transform any attribute. Customer type, Gender, Product line, Payment, Branch are all character type, but we don't need to convert them into factors despite them having repetitive values, is because factors in R are nothing but ordinal data. And we don't need order/hierarchy for any of the above mentioned attributes. So i am not performing any transformation.

```
summary(market)
```

```
##   Invoice ID           Branch              City           Customer type
##  Length:1000        Length:1000        Length:1000        Length:1000
```

```
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     Gender          Product line         Unit price       Quantity
##  Length:1000        Length:1000        Min.   :10.08   Min.   : 1.00
##  Class :character   Class :character   1st Qu.:32.88   1st Qu.: 3.00
##  Mode  :character   Mode  :character   Median :55.23   Median : 5.00
##                                        Mean   :55.67   Mean   : 5.51
##                                        3rd Qu.:77.94   3rd Qu.: 8.00
##                                        Max.   :99.96   Max.   :10.00
##      Tax 5%            Total             Date              Time
##  Min.   : 0.5085   Min.   :  10.68   Length:1000        Length:1000
##  1st Qu.: 5.9249   1st Qu.: 124.42   Class :character   Class1:hms
##  Median :12.0880   Median : 253.85   Mode  :character   Class2:difftime
##  Mean   :15.3794   Mean   : 322.97                      Mode  :numeric
##  3rd Qu.:22.4453   3rd Qu.: 471.35
##  Max.   :49.6500   Max.   :1042.65
##    Payment               cogs        gross margin percentage  gross income
##  Length:1000        Min.   : 10.17   Min.   :4.762           Min.   : 0.5085
##  Class :character   1st Qu.:118.50   1st Qu.:4.762           1st Qu.: 5.9249
##  Mode  :character   Median :241.76   Median :4.762           Median :12.0880
##                     Mean   :307.59   Mean   :4.762           Mean   :15.3794
##                     3rd Qu.:448.90   3rd Qu.:4.762           3rd Qu.:22.4453
##                     Max.   :993.00   Max.   :4.762           Max.   :49.6500
##      Rating
##  Min.   : 4.000
##  1st Qu.: 5.500
##  Median : 7.000
##  Mean   : 6.973
##  3rd Qu.: 8.500
##  Max.   :10.000
```

```
library(Hmisc)
```

```
## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
describe(market)
```

```
## market
##
##  17  Variables      1000  Observations
## ---------------------------------------------------------------------------
## Invoice ID
##        n  missing distinct
##     1000        0     1000
##
## lowest : 101-17-6199 101-81-4070 102-06-2002 102-77-2261 105-10-6182
## highest: 894-41-5205 895-03-6665 895-66-0685 896-34-0956 898-04-2717
## ---------------------------------------------------------------------------
## Branch
##        n  missing distinct
##     1000        0        3
##
## Value          A     B     C
## Frequency    340   332   328
## Proportion 0.340 0.332 0.328
## ---------------------------------------------------------------------------
## City
##        n  missing distinct
##     1000        0        3
##
## Value       Mandalay Naypyitaw    Yangon
## Frequency        332       328       340
## Proportion     0.332     0.328     0.340
## ---------------------------------------------------------------------------
## Customer type
##        n  missing distinct
##     1000        0        2
##
## Value      Member Normal
## Frequency     501    499
## Proportion  0.501  0.499
## ---------------------------------------------------------------------------
## Gender
##        n  missing distinct
##     1000        0        2
##
## Value      Female   Male
## Frequency     501    499
## Proportion  0.501  0.499
## ---------------------------------------------------------------------------
## Product line
##        n  missing distinct
##     1000        0        6
##
## lowest : Electronic accessories Fashion accessories    Food and beverages     Health and beauty
## highest: Fashion accessories    Food and beverages     Health and beauty      Home and lifestyle
##
## Value      Electronic accessories    Fashion accessories     Food and beverages
```

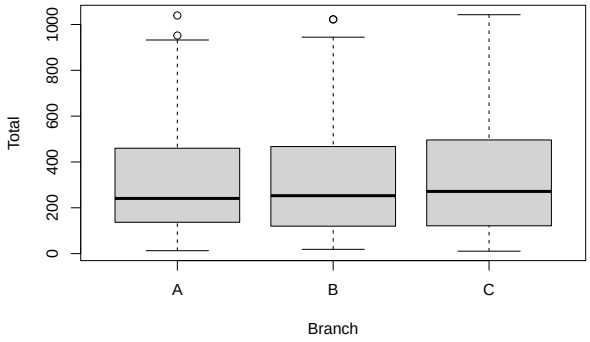
```
## Frequency                     170                   178                   174
## Proportion                  0.170                 0.178                 0.174
##
## Value           Health and beauty     Home and lifestyle      Sports and travel
## Frequency                     152                   160                   166
## Proportion                  0.152                 0.160                 0.166
## ---------------------------------------------------------------------------
## Unit price
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      943        1    55.67     30.6    15.28    19.31
##      .25      .50      .75      .90      .95
##    32.88    55.23    77.94    93.12    97.22
##
## lowest : 10.08 10.13 10.16 10.17 10.18, highest: 99.82 99.83 99.89 99.92 99.96
## ---------------------------------------------------------------------------
## Quantity
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0       10     0.99     5.51     3.36        1        1
##      .25      .50      .75      .90      .95
##        3        5        8       10       10
##
## lowest :  1  2  3  4  5, highest:  6  7  8  9 10
##
## Value          1     2     3     4     5     6     7     8     9    10
## Frequency    112    91    90   109   102    98   102    85    92   119
## Proportion 0.112 0.091 0.090 0.109 0.102 0.098 0.102 0.085 0.092 0.119
## ---------------------------------------------------------------------------
## Tax 5%
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      990        1    15.38    12.89    1.956    3.243
##      .25      .50      .75      .90      .95
##    5.925   12.088   22.445   34.234   39.166
##
## lowest :  0.5085  0.6045  0.6270  0.6390  0.6990
## highest: 48.6900 48.7500 49.2600 49.4900 49.6500
## ---------------------------------------------------------------------------
## Total
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      990        1      323    270.7    41.07    68.10
##      .25      .50      .75      .90      .95
##   124.42   253.85   471.35   718.91   822.50
##
## lowest :   10.6785   12.6945   13.1670   13.4190   14.6790
## highest: 1022.4900 1023.7500 1034.4600 1039.2900 1042.6500
## ---------------------------------------------------------------------------
## Date
##        n  missing distinct
##     1000        0       89
##
## lowest : 1/1/2019  1/10/2019 1/11/2019 1/12/2019 1/13/2019
## highest: 3/5/2019  3/6/2019  3/7/2019  3/8/2019  3/9/2019
## ---------------------------------------------------------------------------
## Time [secs]
##        n  missing distinct
```

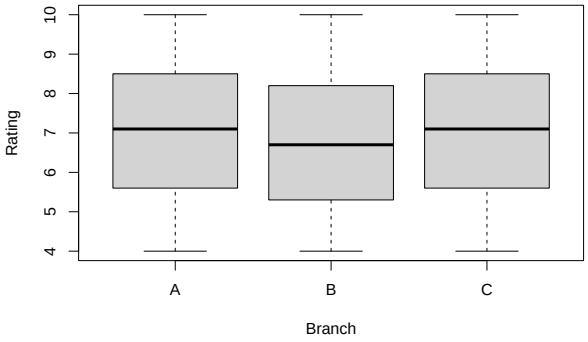
```
##     1000        0      506
##
## lowest : 10:00:00 10:01:00 10:02:00 10:03:00 10:04:00
## highest: 20:52:00 20:54:00 20:55:00 20:57:00 20:59:00
## ---------------------------------------------------------------------------
## Payment
##        n  missing distinct
##     1000        0        3
##
## Value             Cash Credit card     Ewallet
## Frequency          344         311         345
## Proportion       0.344       0.311       0.345
## ---------------------------------------------------------------------------
## cogs
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      990        1    307.6    257.8    39.11    64.86
##      .25      .50      .75      .90      .95
##   118.50   241.76   448.91   684.68   783.33
##
## lowest :  10.17  12.09  12.54  12.78  13.98, highest: 973.80 975.00 985.20 989.80 993.00
## ---------------------------------------------------------------------------
## gross margin percentage
##        n  missing distinct     Info     Mean      Gmd
##     1000        0        1        0    4.762        0
##
## Value      4.761905
## Frequency      1000
## Proportion        1
## ---------------------------------------------------------------------------
## gross income
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      990        1    15.38    12.89    1.956    3.243
##      .25      .50      .75      .90      .95
##    5.925   12.088   22.445   34.234   39.166
##
## lowest :  0.5085  0.6045  0.6270  0.6390  0.6990
## highest: 48.6900 48.7500 49.2600 49.4900 49.6500
## ---------------------------------------------------------------------------
## Rating
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0       61        1    6.973    1.985    4.295    4.500
##      .25      .50      .75      .90      .95
##    5.500    7.000    8.500    9.400    9.700
##
## lowest :  4.0  4.1  4.2  4.3  4.4, highest:  9.6  9.7  9.8  9.9 10.0
## ---------------------------------------------------------------------------
```

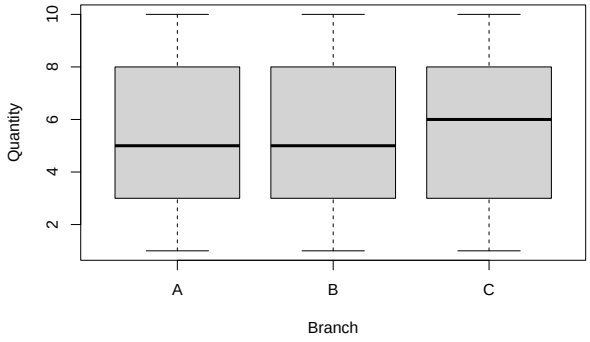
- Describe gives us complete picture of the attributes from the missing values, disntinct values , descriptive statistics and quantile values.
- There are no missing values in any of the attributes
- Almost equal number of orders from males and females, Members and normal customers
- From the descriptive statistics especially for the numeric attributes, we see that the values are the same for Tax and Grossincome. An interesting observation is that the lowest rating was 4 on a scale

of 1-10. For time(from 10am to 9pm) since the values are continuous, by discretizing it we can give a broader picture of our analysis related to time.

```
#Outliers:

boxplot(Total ~ Branch, data = market)
boxplot(Rating ~ Branch, data = market)
boxplot(Quantity ~ Branch, data = market)
```







- Just a few outliers in the total cost from branch A and B. These values dont need to be excluded from the dataset as they are less in number and they can be justified as high purchases.

```
#Cleaning column names for easy use

colnames(market) <- c('InvoiceID','Branch','City','Customertype','Gender',
                      'Productline','Unitprice','Quantity','Tax','Total',
                      'Date','Time','Payment','cogs','grossmarginpercentage',
                      'grossincome','Rating')


# Discretizing time levels
market$time_levels <- cut(strptime(market$Time, format = "%H:%M:%S"),
                          breaks = strptime(c( "10:00:00","12:00:00","16:00:00",
                                               "19:00:00","21:00:00")
                                            , format = "%H:%M:%S"),
```

```
   labels = c("Morning", "Afternoon","Evening","Night"))


# Getting data by branches

branchA <- market[which(market$Branch == 'A'),]
branchB <- market[which(market$Branch == 'B'),]
branchC <- market[which(market$Branch == 'C'),]
```

Analysis :

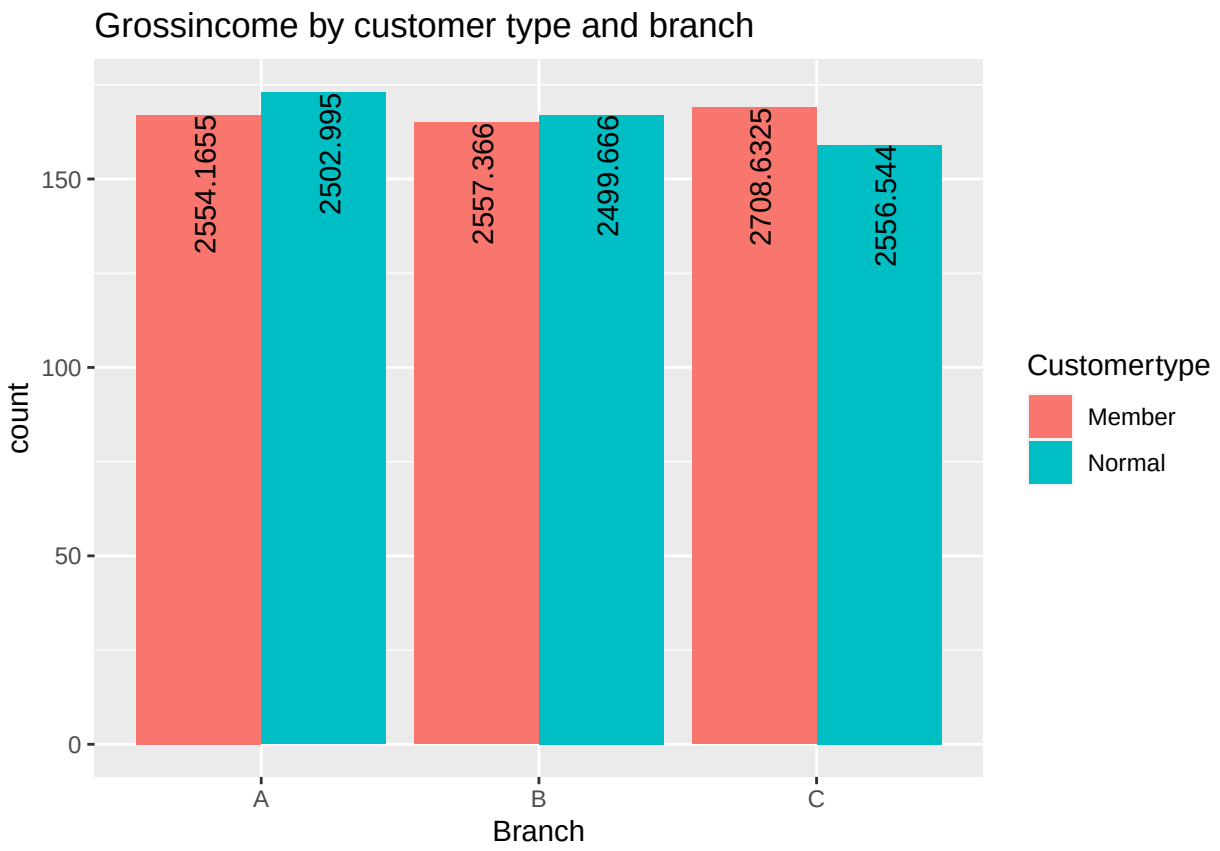
```
library(dplyr)
library(ggplot2)
```

- Grossincome by customertype and branch - All three branches had similar gross income generated by customers who were members and normal type

```
# Grouping data by branch and customer type
tab1 <- market %>% group_by(Branch,Customertype) %>% summarise(count = n(),
                                                               grossincome = sum(grossincome),Quantity =
```

```
## `summarise()` has grouped output by 'Branch'. You can override using the `.groups` argument.
```

```
ggplot(data = tab1,aes(x= Branch, y=count, fill=Customertype)) +
  geom_bar(stat="identity", position=position_dodge()) +
  ggtitle("Grossincome by customer type and branch")+
  geom_text(aes(label = grossincome), position = position_dodge(width = 0.9),
            vjust = 0.8,size = 4, hjust = 1, angle=  90)
```



- Gross income and tax summed upto the same, which was first observed in descriptive statistics.

```
grossincome_B <- market %>% group_by(Branch) %>% summarise(grossincome = sum(grossincome),Quantity = su
grossincome_B
```

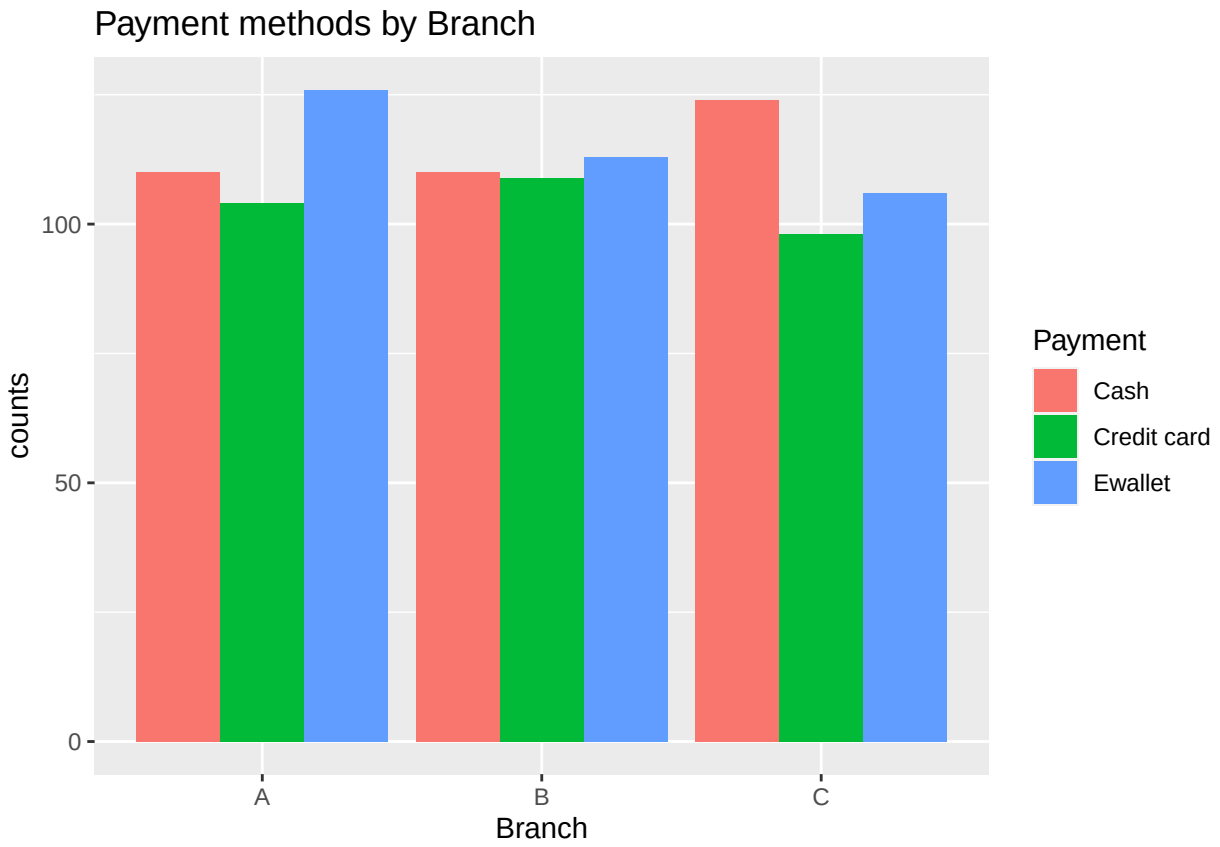
```
## # A tibble: 3 x 4
##   Branch grossincome Quantity   Tax
##   <chr>        <dbl>    <dbl> <dbl>
## 1 A            5057.     1859 5057.
## 2 B            5057.     1820 5057.
## 3 C            5265.     1831 5265.
```

Payment methods used by branch:

- At Branch A, Ewallet high Ewallet payments were made, while at Branch B all 3 payment methods were equally used. On the contrary Branch C had high cash payments, this does in a way indicate that change should be checked often.

```
# Grouping data by payment method and branch with plot
ggplot(market %>% group_by(Payment, Branch) %>% summarise(counts = n()),
       aes(x= Branch, y= counts,fill = Payment)) +
  geom_bar(stat="identity",position=position_dodge()) +
  ggtitle("Payment methods by Branch")
```

```
## `summarise()` has grouped output by 'Payment'. You can override using the `.groups` argument.
```



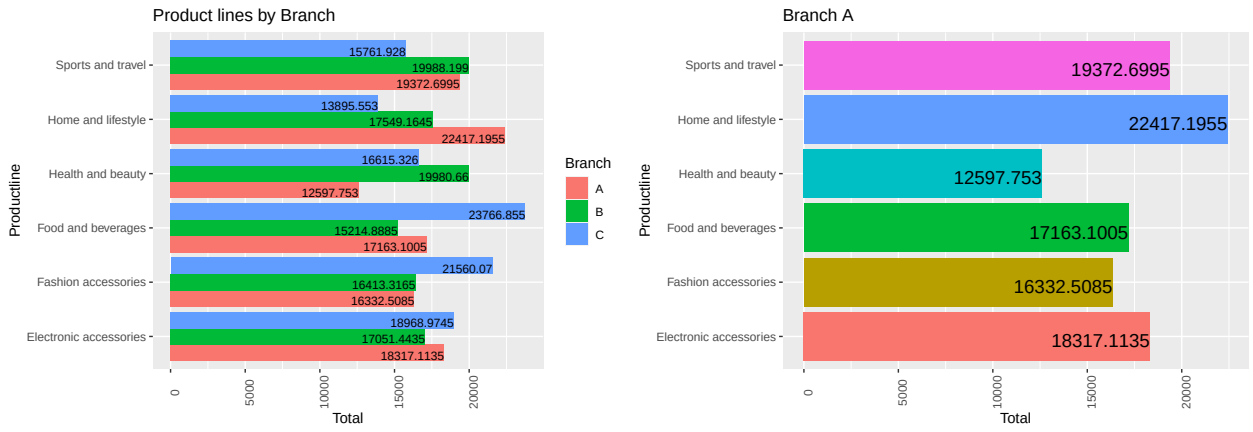
By Branch findings:

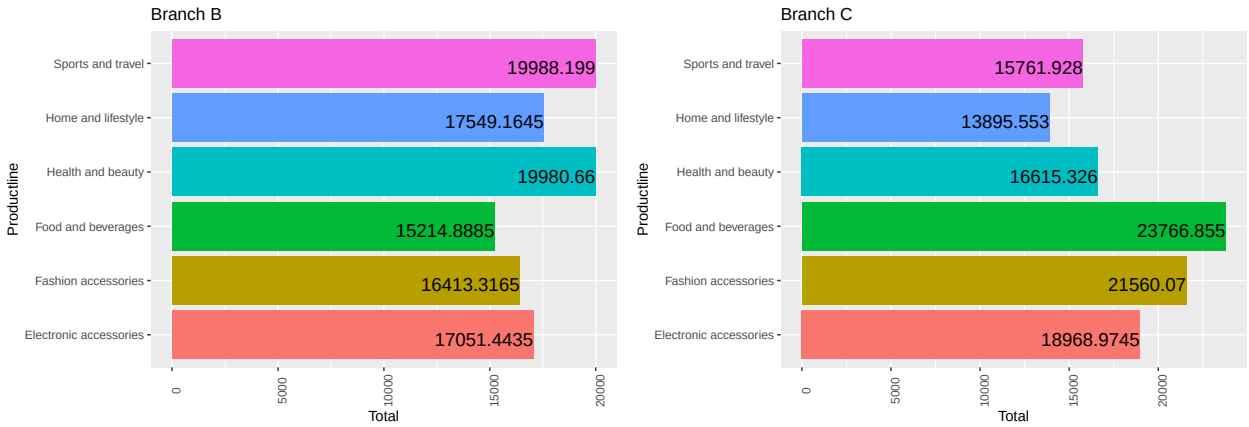
```
# Grouping branch and product categories and plotting the data

BP <-  market %>% group_by(Branch, Productline) %>% summarise(
  counts = n(),Total = sum(Total))
```

```
## `summarise()` has grouped output by 'Branch'. You can override using the `.groups` argument.
```

```
ggplot(data= BP,aes(y= Productline, x= Total,fill = Branch)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Product lines by Branch") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total), position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1)


# Branch A
ggplot(data= branchA %>% group_by(Productline) %>% summarise(counts = n(),
                                                             Total = sum(Total))
       , aes(x =Total,y=Productline, fill = Productline)) +
  geom_bar(stat="identity",position=position_dodge()) +
  ggtitle("Branch A")+
```

```
  theme(axis.text.x =  element_text(angle = 90), legend.position = "none")+
  geom_text(aes(label = Total), position = position_dodge(width = 0.9),
            vjust = 0.8,size = 5, hjust = 1)


#Branch B
ggplot(data= branchB %>% group_by(Productline) %>% summarise(
  counts = n(), Total = sum(Total))
       , aes(y=Productline, x =Total, fill = Productline)) +
  geom_bar(stat="identity",position=position_dodge()) +
  ggtitle("Branch B")+
  theme(axis.text.x =  element_text(angle = 90), legend.position = "none")+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 5, hjust = 1)


#Branch C
ggplot(data= branchC %>% group_by(Productline) %>% summarise(
  counts = n(), Total = sum(Total))
       , aes(y=Productline, x = Total, fill = Productline)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Branch C")+
  theme(axis.text.x =  element_text(angle = 90), legend.position = "none")+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 5, hjust = 1)
```





- Branch A generated high total amounts in home and lifestyle($22417.19)
- Branch B generated high total amounts in Sports and travel($19988.19). Health and beauty($19980.66)
- Branch C generated high total amounts in Food and Beverages($23766.85) Fashion accessories ($21560.07) Electronic accessories($18968.97)

By Gender findings:

```
#Grouping data by Branch and Gender
genderdata <- market %>% group_by(Branch, Gender) %>% summarise(
  Quantity = sum(Quantity), count = n(), grossincome = sum(grossincome))
```

```
## `summarise()` has grouped output by 'Branch'. You can override using the `.groups` argument.
```

```
ggplot(data = genderdata,aes(x= Branch, y= Quantity,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge()) +
  ggtitle("Grossincome by Gender and Branch") +
  geom_text(aes(label = grossincome), position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)

# Payment methods
ggplot(market %>% group_by(Gender,Payment) %>% summarise(counts = n()),
       aes(x= Gender, y= counts,fill = Payment)) +
  ggtitle("Payment by gender")+
  geom_bar(stat="identity",position=position_dodge())
```

```
## `summarise()` has grouped output by 'Gender'. You can override using the `.groups` argument.
```

```
# Productlines
ggplot(market %>% group_by(Gender,Productline) %>% summarise(
  Total = sum(Total),count = n()),
  aes(x= Productline, y= Total,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Productlines by Gender") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 2.5, hjust = 1, angle  = 90)
```

‘summarise()‘ has grouped output by ’Gender’. You can override using the ‘.groups‘ argument.

```
# Branch A
ggplot(branchA %>% group_by(Gender,Productline) %>% summarise(
  Total = sum(Total),count = n()),
  aes(x= Productline, y= Total,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Productlines by Gender for Branch A") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)
```

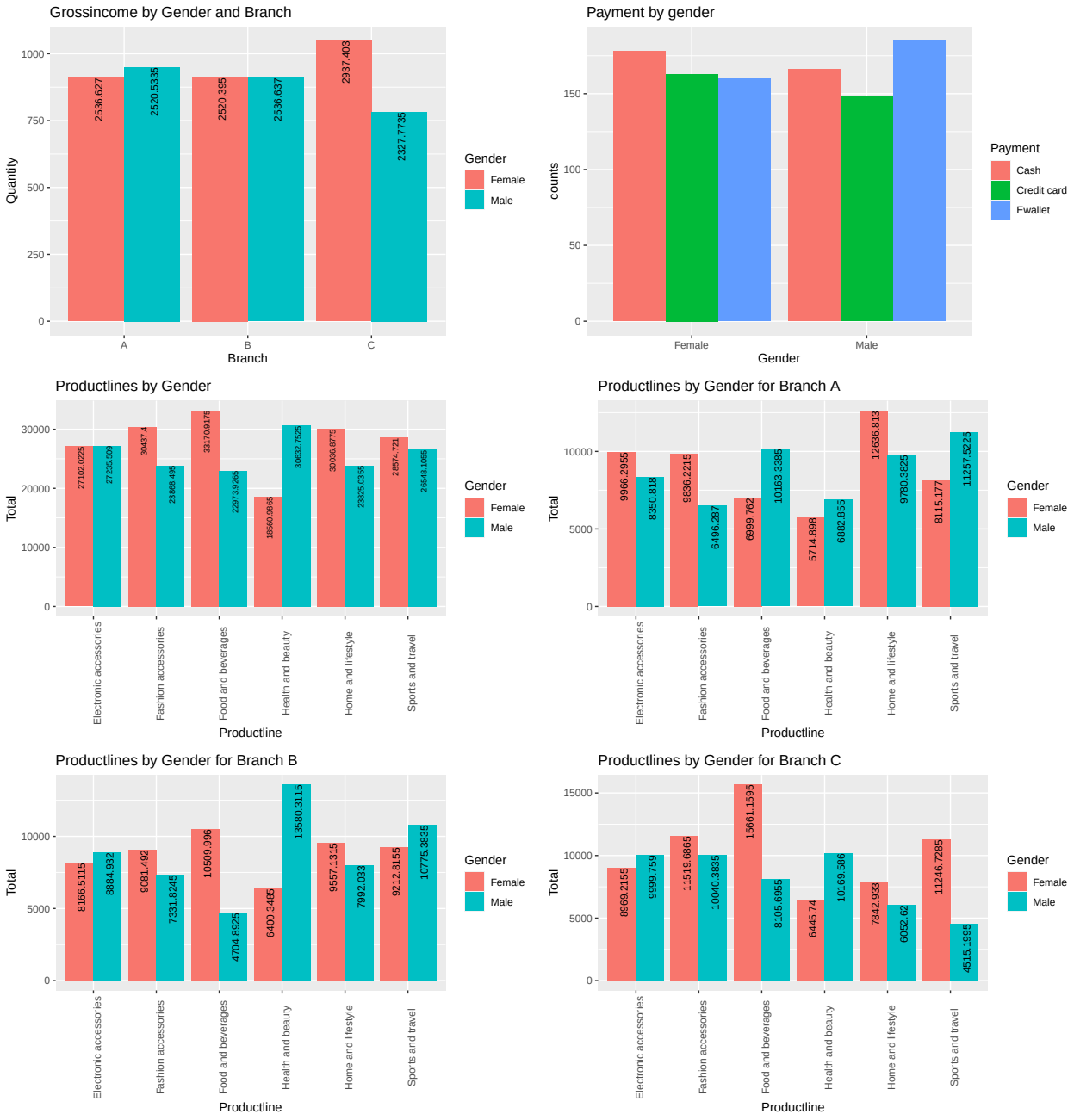
‘summarise()‘ has grouped output by ’Gender’. You can override using the ‘.groups‘ argument.

```
#Branch B
ggplot(branchB %>% group_by(Gender,Productline) %>% summarise(
  Total = sum(Total),count = n()),
  aes(x= Productline, y= Total,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Productlines by Gender for Branch B") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)
```

‘summarise()‘ has grouped output by ’Gender’. You can override using the ‘.groups‘ argument.

```
#Branch C
ggplot(branchC %>% group_by(Gender,Productline) %>% summarise(
  Total = sum(Total),count = n()),
  aes(x= Productline, y= Total,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Productlines by Gender for Branch C") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)
```

‘summarise()‘ has grouped output by ’Gender’. You can override using the ‘.groups‘ argument.



- Branch A : Males generated high Gross income($2520.53). Females purchased more electronic accessories, fashion accessories,home and lifestyle products while males purchased more in Food and Beverages, sports and travel and surprisingly health and beauty
- Branch B : Males and Females generated equal gross income. Females purchased more Food and Beverages, fashion accessories,home and lifestyle products while males purchased more in electronic accessories, sports and travel and surprisingly very high in health and beauty
- Branch C : Females generated high gross income($2937.4). Females purchased more Food and Beverages, fashion accessories,home and lifestyle, sports and travel products while males purchased more in electronic accessories, and surprisingly very high in health and beauty
- Females mostly preferred Cash payments while males preferred Ewallet

- Overall females purchased more products from Food and Beverages, fashion accessories,home and lifestyle, sports and travel products
- Overall males purchased more in health and beauty and a little more in electronic goods

Productlines by Customertype:

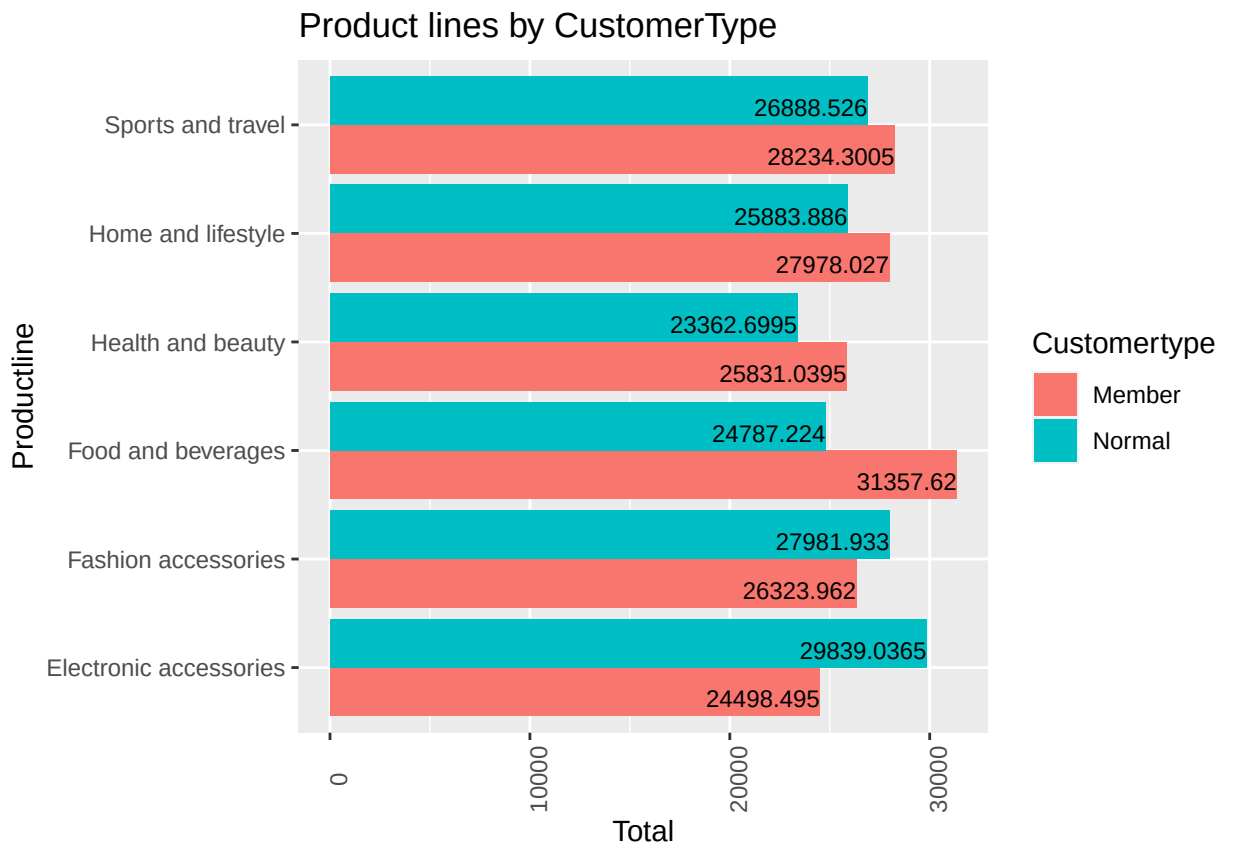
```
PC <-  market %>% group_by(Customertype, Productline) %>% summarise(
  counts = n(),Total = sum(Total))
```

```
## `summarise()` has grouped output by 'Customertype'. You can override using the `.groups` argument.
```

```
PC
```

```
## # A tibble: 12 x 4
## # Groups:   Customertype [2]
##    Customertype Productline            counts  Total
##    <chr>        <chr>                   <int>  <dbl>
##  1 Member       Electronic accessories     78 24498.
##  2 Member       Fashion accessories        86 26324.
##  3 Member       Food and beverages         94 31358.
##  4 Member       Health and beauty          73 25831.
##  5 Member       Home and lifestyle         83 27978.
##  6 Member       Sports and travel          87 28234.
##  7 Normal       Electronic accessories     92 29839.
##  8 Normal       Fashion accessories        92 27982.
##  9 Normal       Food and beverages         80 24787.
## 10 Normal       Health and beauty          79 23363.
## 11 Normal       Home and lifestyle         77 25884.
## 12 Normal       Sports and travel          79 26889.
```

```
ggplot(data= PC,aes(y= Productline, x= Total,fill = Customertype)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Product lines by CustomerType") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Total),
            position = position_dodge(width = 0.9),
            vjust = 0.9,size = 3, hjust = 1)
```



- Overall, customers who are existing members in contributed more to Sports and travel, home and lifestyle, health and beauty, food and beverages
- Normal customers, who were not members, had high sales in Fashion accessories and electronic accessories.

Timeperiod of Sales

```
Time_B<- market %>% group_by(Branch,time_levels) %>% summarise(
  counts = n(), grossincome = sum(grossincome), Total = sum(Total))
```

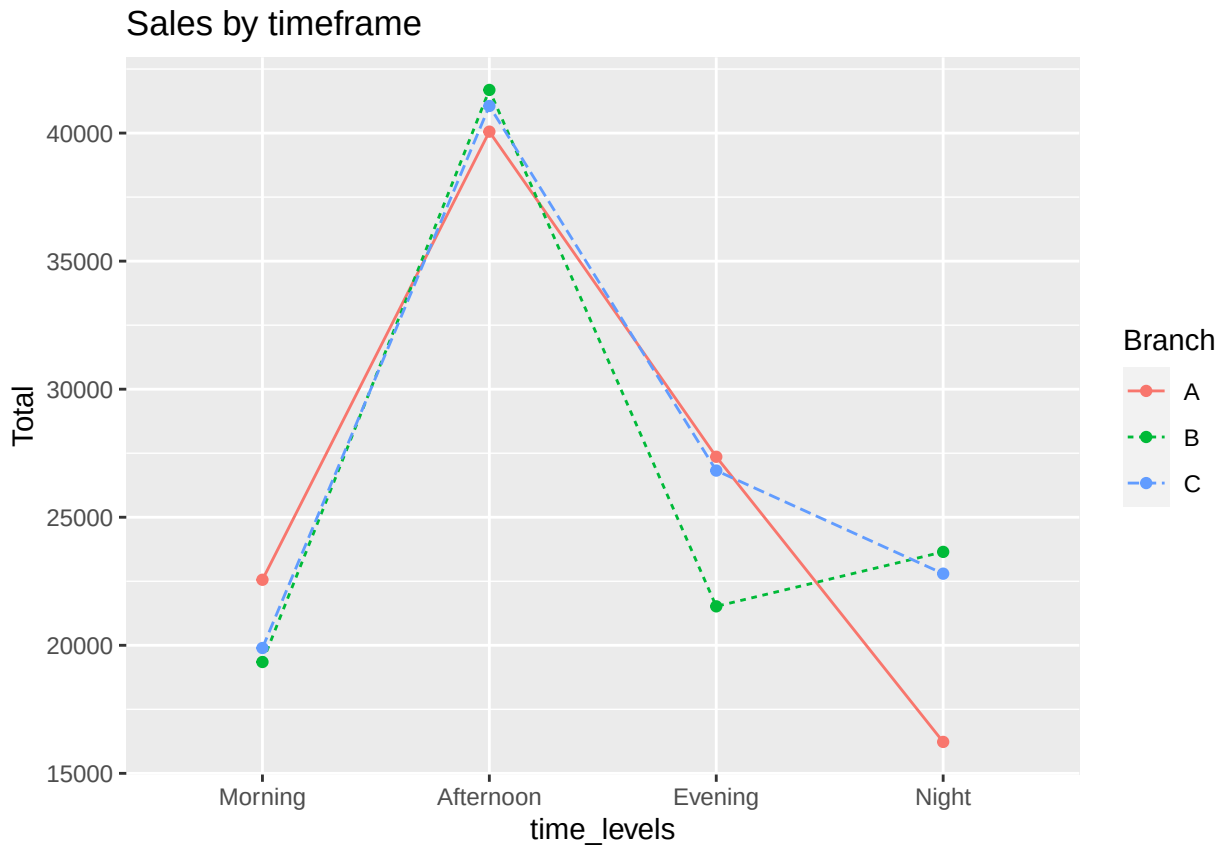
```
## `summarise()` has grouped output by 'Branch'. You can override using the `.groups` argument.
```

```
Time_B
```

```
## # A tibble: 12 x 5
## # Groups:   Branch [3]
##    Branch time_levels counts grossincome  Total
##    <chr>  <fct>        <int>       <dbl>  <dbl>
##  1 A      Morning         73       1074. 22558.
##  2 A      Afternoon      126       1907. 40055.
##  3 A      Evening         92       1303. 27360.
##  4 A      Night           49        773. 16227.
##  5 B      Morning         59        921. 19348.
```

```
##  6 B      Afternoon      125       1985. 41684.
##  7 B      Evening         72       1025. 21520.
##  8 B      Night           76       1126. 23647.
##  9 C      Morning         59        947. 19893.
## 10 C      Afternoon      126       1955. 41059.
## 11 C      Evening         80       1277. 26821.
## 12 C      Night           63       1086. 22796.
```

```
ggplot(data= Time_B, aes(x=time_levels, y=Total, group=Branch)) +
  geom_line(aes(linetype=Branch, color = Branch))+
  geom_point(aes(color = Branch)) +
  ggtitle("Sales by timeframe")
```



- Discretized Timings of orders into 4 groups : 10 am to 12pm as morning, 12pm to 4pm as afternoon, 4pm to 7pm as evening and 7pm to 9pm as night
- In all 3 branches, the total sales and gross income were high during afternoon period and sales gradually increased from 9am to 12pm
- For Branch A : Sales continuously dipped after 4pm
- For Branch B : Sales dipped from 4pm to 7pm but increased after 7pm
- For Branch C : Sales decreased after 4pm but it was not a steep drop.

Rating analysis:

by store, by category,by gender, by customertype

```
rb<- market %>% group_by(Branch) %>% summarise(counts = n(),
                                                Averagerating = mean(Rating))
rb
```

```
## # A tibble: 3 x 3
##   Branch counts Averagerating
##   <chr>   <int>         <dbl>
## 1 A         340          7.03
## 2 B         332          6.82
## 3 C         328          7.07
```

```
rc <- market %>% group_by(Customertype) %>% summarise(
  counts = n(), Averagerating = mean(Rating))
rc
```

```
## # A tibble: 2 x 3
##   Customertype counts Averagerating
##   <chr>         <int>         <dbl>
## 1 Member          501          6.94
## 2 Normal          499          7.01
```

```
rg <- market %>% group_by(Gender) %>% summarise(
  counts = n(), Averagerating = mean(Rating))
rg
```

```
## # A tibble: 2 x 3
##   Gender counts Averagerating
##   <chr>   <int>         <dbl>
## 1 Female    501          6.96
## 2 Male      499          6.98
```

```
rbg <- market %>% group_by(Branch,Gender) %>% summarise(
  counts = n(), Averagerating = round(mean(Rating),3))
```

```
## `summarise()` has grouped output by 'Branch'. You can override using the `.groups` argument.
```

```
rbg
```

```
## # A tibble: 6 x 4
## # Groups:   Branch [3]
##   Branch Gender counts Averagerating
##   <chr>  <chr>   <int>         <dbl>
## 1 A      Female    161          6.84
## 2 A      Male      179          7.20
## 3 B      Female    162          6.88
## 4 B      Male      170          6.76
## 5 C      Female    178          7.16
## 6 C      Male      150          6.97
```

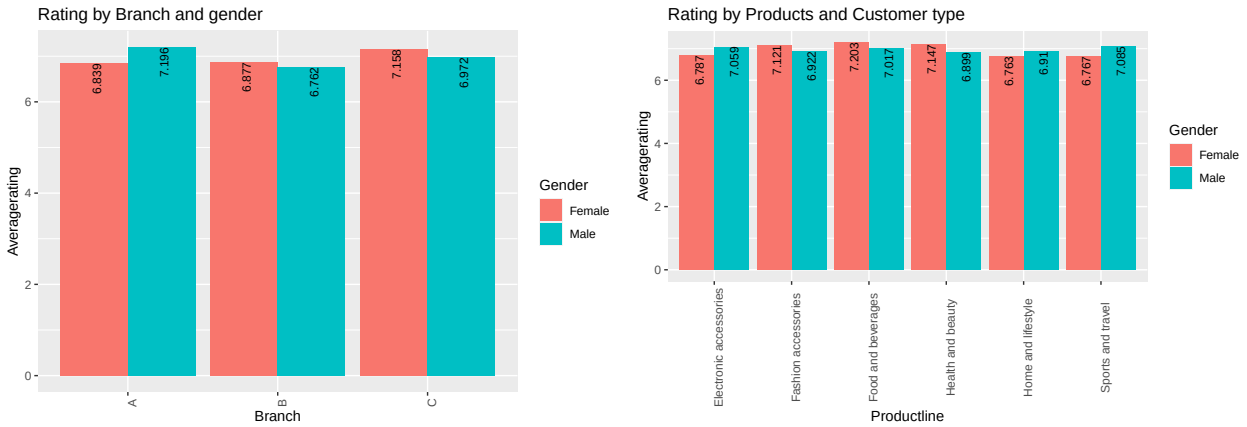
```
rgc <- market %>% group_by(Gender,Productline) %>% summarise(
  counts = n(), Averagerating = round(mean(Rating),3))
```

```
## `summarise()` has grouped output by 'Gender'. You can override using the `.groups` argument.
```

```
rgc
```

```
## # A tibble: 12 x 4
## # Groups:   Gender [2]
##    Gender Productline            counts Averagerating
##    <chr>  <chr>                   <int>         <dbl>
##  1 Female Electronic accessories     84          6.79
##  2 Female Fashion accessories        96          7.12
##  3 Female Food and beverages         90          7.20
##  4 Female Health and beauty          64          7.15
##  5 Female Home and lifestyle         79          6.76
##  6 Female Sports and travel          88          6.77
##  7 Male   Electronic accessories     86          7.06
##  8 Male   Fashion accessories        82          6.92
##  9 Male   Food and beverages         84          7.02
## 10 Male   Health and beauty          88          6.90
## 11 Male   Home and lifestyle         81          6.91
## 12 Male   Sports and travel          78          7.08
```

```
ggplot(data = rbg,aes(x= Branch, y= Averagerating,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Rating by Branch and gender") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Averagerating),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)


ggplot(data = rgc,aes(x= Productline, y= Averagerating,fill = Gender)) +
  geom_bar(stat="identity",position=position_dodge())+
  ggtitle("Rating by Products and Customer type") +
  theme(axis.text.x =  element_text(angle = 90))+
  geom_text(aes(label = Averagerating),
            position = position_dodge(width = 0.9),
            vjust = 0.8,size = 3, hjust = 1, angle  = 90)
```



- The average rating given for all 3 branches were more or less the same but branch C had the highest at 7.07
- Customers who were members surprisingly had low average rating at 6.94 compared to normal customers with average of 7
- Male and Female customers had more or less equal average rating overall but for individual branches, Males had high average rating for branch A while females had high average rating for branch B,C
- Females gave higher rating for health and beauty,fashion and accessories, food and beverages and categories where as males gave higher ratings in electronic accessories, home and lifestyle, sports and travel

NOTE : A lot of indepth analysis related to exact time of purchases wrt customer type, product category, ratings by categories in each branch etc can be explored heavily further.