12/9/2021

E-Commerce When a customer makes a purchase on an online retail website, E-commerce data is collected, which informs businesses about their products, customers, stores, sales, purchases, and price.The US market are the second largest market for E-commerce and it provides a huge amount of data on daily basis. This data gives the insights on how the business strategies are applied to gain more customers day-by-day. While working on this data set we have found some interesting facts on the discounts, their retail price and many more.

With sales, ratings, retail price we have performed different analysis such as clustering, text analysis, sentiment analysis. We also worked on some probabilities by comparing the sales provided in each month with many other columns acquired from the data set.

Data Preprocessing.

Loading of the Libraries that are used in the project:

```
#Libraries
library(tidyr)
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(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

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

```
library(ggplot2)
library(modeest)
```

```
## Warning: package 'modeest' was built under R version 4.1.2
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.1.2
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 4.1.2
```

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

```
library(cluster)
library(clValid)
```

```
## Warning: package 'clValid' was built under R version 4.1.2
```

```
library(cluster)
library(fpc)
```

```
## Warning: package 'fpc' was built under R version 4.1.2
```

```
library(plotrix)
library(magrittr)
```

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

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

```
library(knitr)
library(tidytext)
```

```
## Warning: package 'tidytext' was built under R version 4.1.2
```

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 4.1.2
```

```
## Loading required package: RColorBrewer
```

```
library(crqa)
```

```
## Warning: package 'crqa' was built under R version 4.1.2
```

```
## Registered S3 methods overwritten by 'FSA':
##   method       from
##   confint.boot car
##   hist.boot    car
```

```
library(lemon)
```

```
## Warning: package 'lemon' was built under R version 4.1.2
```

```
##
## Attaching package: 'lemon'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     CoordCartesian, element_render
```

```
library(nonlinearTseries)
```

```
## Warning: package 'nonlinearTseries' was built under R version 4.1.2
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
##
## Attaching package: 'nonlinearTseries'
```

```
## The following object is masked from 'package:grDevices':
##
##     contourLines
```

This are the data sets that we have used in our project:

```
data <- read.csv('US  E-commerce.csv')

head(data)
```

	Order.Date <chr>	Row… <int>	Order.ID <chr>	Ship.Mode <chr>	Customer.ID <chr>	Segment <chr>	Country <chr>
1	1/1/2020	849	CA-2017-107503	Standard Class	GA-14725	Consumer	United States
2	1/1/2020	4010	CA-2017-144463	Standard Class	SC-20725	Consumer	United States
3	1/1/2020	6683	CA-2017-154466	First Class	DP-13390	Home Office	United States
4	1/1/2020	8070	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
5	1/1/2020	8071	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
6	1/1/2020	8072	CA-2017-151750	Standard Class	JM-15250	Consumer	United States

6 rows | 1-8 of 20 columns

```
colnames(data)
```

```
##  [1] "Order.Date"   "Row.ID"       "Order.ID"     "Ship.Mode"    "Customer.ID"
##  [6] "Segment"      "Country"      "City"         "State"        "Postal.Code"
## [11] "Region"       "Product.ID"   "Category"     "Sub.Category" "Product.Name"
## [16] "Sales"        "Quantity"     "Discount"     "Profit"
```

```
fd <- read.csv('flipkart_data.csv')

head(fd)
```

	uniq_id <chr>	crawl_timestamp <chr>
1	c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000
2	7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000
3	f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000
4	0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000
5	bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000
6	c2a17313954882c1dba461863e98adf2	2016-03-25 22:59:23 +0000

6 rows | 1-3 of 16 columns

```
colnames(fd)
```

```
##  [1] "uniq_id"                "crawl_timestamp"
##  [3] "product_url"            "product_name"
##  [5] "product_category_tree"  "pid"
##  [7] "retail_price"           "discounted_price"
##  [9] "image"                  "is_FK_Advantage_product"
## [11] "description"            "product_rating"
## [13] "overall_rating"         "brand"
## [15] "product_specifications"
```

In the data preprocessing we performed different types such as removed the null values, converted the data type of some columns as per the requirement, we extracted the required data from a particular column as per the necessity, rounded the numbers after getting the results upto the required decimal points, plot the graphs on the multiple operations performed, performed correlation for the discount and rating given in the data set.

Here are the data preprocessing that we have done which will be required to perform the operations.

```
#Removing of the null values from the data sets.
data <- drop_na(data)
data
```

Order.Date <chr>	Row… <int>	Order.ID <chr>	Ship.Mode <chr>	Customer.ID <chr>	Segment <chr>	Country <chr>
1/1/2020	849	CA-2017-107503	Standard Class	GA-14725	Consumer	United States
1/1/2020	4010	CA-2017-144463	Standard Class	SC-20725	Consumer	United States
1/1/2020	6683	CA-2017-154466	First Class	DP-13390	Home Office	United States
1/1/2020	8070	CA-2017-151750	Standard Class	JM-15250	Consumer	United States

Order.Date <chr>	Row… <int>	Order.ID <chr>	Ship.Mode <chr>	Customer.ID <chr>	Segment <chr>	Country <chr>
1/1/2020	8071	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
1/1/2020	8072	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
1/1/2020	8073	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
1/1/2020	8074	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
1/1/2020	8075	CA-2017-151750	Standard Class	JM-15250	Consumer	United States
1/1/2020	8076	CA-2017-151750	Standard Class	JM-15250	Consumer	United States

1-10 of 3,312 rows | 1-7 of 19 columns Previous 1 2 3 4 5 6 … 332 Next

```
colnames(data)
```

```
##  [1] "Order.Date"   "Row.ID"       "Order.ID"     "Ship.Mode"    "Customer.ID"
##  [6] "Segment"      "Country"      "City"         "State"        "Postal.Code"
## [11] "Region"       "Product.ID"   "Category"     "Sub.Category" "Product.Name"
## [16] "Sales"        "Quantity"     "Discount"     "Profit"
```

```
fd <- drop_na(fd)
fd
```

uniq_id <chr>	crawl_timestamp <chr>
c2d766ca982eca8304150849735ffef9	2016-03-25 22:59:23 +0000
7f7036a6d550aaa89d34c77bd39a5e48	2016-03-25 22:59:23 +0000
f449ec65dcbc041b6ae5e6a32717d01b	2016-03-25 22:59:23 +0000
0973b37acd0c664e3de26e97e5571454	2016-03-25 22:59:23 +0000
bc940ea42ee6bef5ac7cea3fb5cfbee7	2016-03-25 22:59:23 +0000
c2a17313954882c1dba461863e98adf2	2016-03-25 22:59:23 +0000
ce5a6818f7707e2cb61fdcdbba61f5ad	2016-03-25 22:59:23 +0000
8542703ca9e6ebdf6d742638dfb1f2ca	2016-03-25 22:59:23 +0000
29c8d290caa451f97b1c32df64477a2c	2016-03-25 22:59:23 +0000
4044c0ac52c1ee4b28777417651faf42	2016-03-25 22:59:23 +0000

1-10 of 10,000 rows | 1-2 of 15 columns Previous 1 2 3 4 5 6 … 1000 Next

```
colnames(fd)
```

```
##  [1] "uniq_id"                "crawl_timestamp"
##  [3] "product_url"            "product_name"
##  [5] "product_category_tree"  "pid"
##  [7] "retail_price"           "discounted_price"
##  [9] "image"                  "is_FK_Advantage_product"
## [11] "description"            "product_rating"
## [13] "overall_rating"         "brand"
## [15] "product_specifications"
```

```
#Appending a new column of month from date time column
data <- mutate(data, month = month(dmy(Order.Date)))
colnames(data)
```

```
##  [1] "Order.Date"   "Row.ID"       "Order.ID"     "Ship.Mode"    "Customer.ID"
##  [6] "Segment"      "Country"      "City"         "State"        "Postal.Code"
## [11] "Region"       "Product.ID"   "Category"     "Sub.Category" "Product.Name"
## [16] "Sales"        "Quantity"     "Discount"     "Profit"       "month"
```

```
# Finding the average of sales column according to their month
avg_sale_per_month <- data %>%
    group_by(month) %>%
    summarise( mean = mean(Sales))
avg_sale_per_month
```

month <dbl>	mean <dbl>
1	283.6863
2	189.7302
3	247.3628
4	179.9090
5	182.8971
6	216.2519
7	200.2850
8	289.5454
9	191.4306
10	260.9964

1-10 of 12 rows Previous **1** 2 Next

```
#Calculating the frequency of different ship modes in the given data set
count_Ship.Mode <- data.frame(table(data$Ship.Mode))
count_Ship.Mode
```

Var1 <fct>	Freq <int>
First Class	572
Same Day	186
Second Class	657
Standard Class	1897

4 rows

```
#Finding out the probaility of each ship modes
count_Ship.Mode = mutate(count_Ship.Mode ,prob = count_Ship.Mode$Freq/sum(count_Ship.Mode$Freq)*
100)
count_Ship.Mode
```

Var1 <fct>	Freq <int>	prob <dbl>
First Class	572	17.270531
Same Day	186	5.615942
Second Class	657	19.836957
Standard Class	1897	57.276570

4 rows

```
#Rounding the values of result of acquired probability upto 2 decimals
count_Ship.Mode$prob <- round(count_Ship.Mode$prob, digits = 2)
count_Ship.Mode
```

Var1 <fct>	Freq <int>	prob <dbl>
First Class	572	17.27
Same Day	186	5.62
Second Class	657	19.84
Standard Class	1897	57.28

4 rows

```
# Overall mean of the sales for 2020
M <- mean(data$Sales)
M
```

```
## [1] 221.3814
```

```
#Removing rows the where No rating available was present from the dataset
fd <-fd[!grepl("No rating available", fd$product_rating),]
```

```
#Converting the product rating data type into numeric
fd$product_rating <- as.numeric(fd$product_rating)
```

```
#Appending a new column as discount which will be discount = (retail_price - discounted_price)/r
etail_price
fd <- mutate(fd,discount = (retail_price - discounted_price)/retail_price)
```

```
#Grouping all the brands present in the data set , taking the correlation of the discount and th
e product_rating , taking the mean of the discount column and taking the mean of the product_rat
ing
relation <- fd  %>%
  group_by(brand) %>%
  summarise(correlation = cor(discount, product_rating), avg_discount = mean(discount), avg_rati
ng = mean(product_rating)) %>%
  drop_na()
```

```
## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero
```

```
## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero

## Warning in cor(discount, product_rating): the standard deviation is zero
```

```
relation
```

brand <chr>	correlation <dbl>	avg_discount <dbl>	avg_rating <dbl>
	-1.464686e-02	0.32842172	3.774484
3a Autocare	-1.000000e+00	0.49032679	4.500000
Aeoss	4.113486e-01	0.41299073	3.200000
Alysa	-1.000000e+00	0.66143189	4.500000
Apple	1.000000e+00	0.02948529	4.800000
APS	-1.000000e+00	0.57148600	3.000000
Asus	3.010696e-01	0.31040435	3.973684
Beige	4.285402e-01	0.59057293	4.260000
Being Nawab	1.000000e+00	0.60521042	2.000000
Belkin	1.131233e-01	0.10361853	3.637500

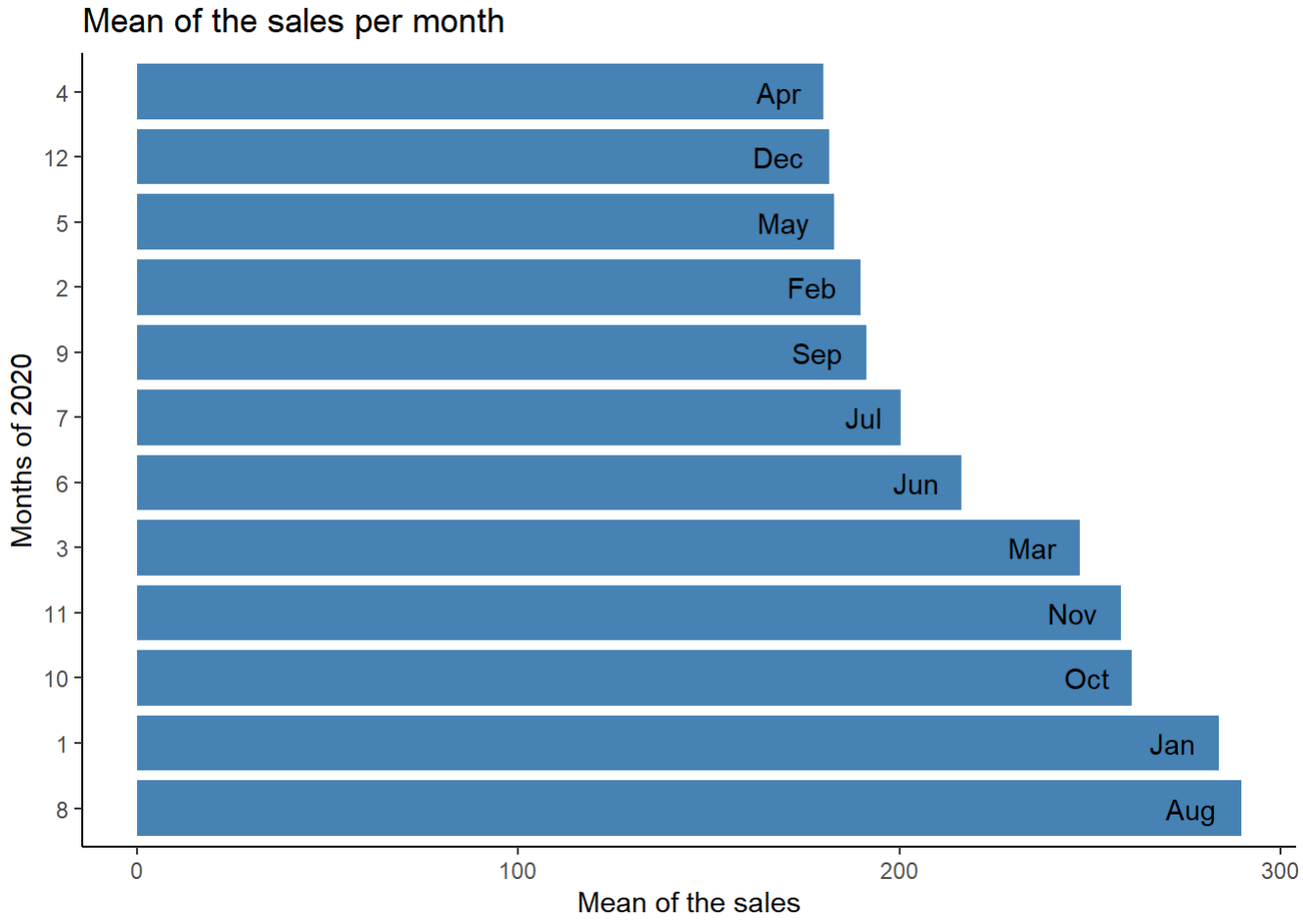
1-10 of 89 rows Previous **1** 2 3 4 5 6 … 9 Next

Here are the Business questions that we have done on the data set.

Probability :

Question 1. Which month have the highest sales in 2020?

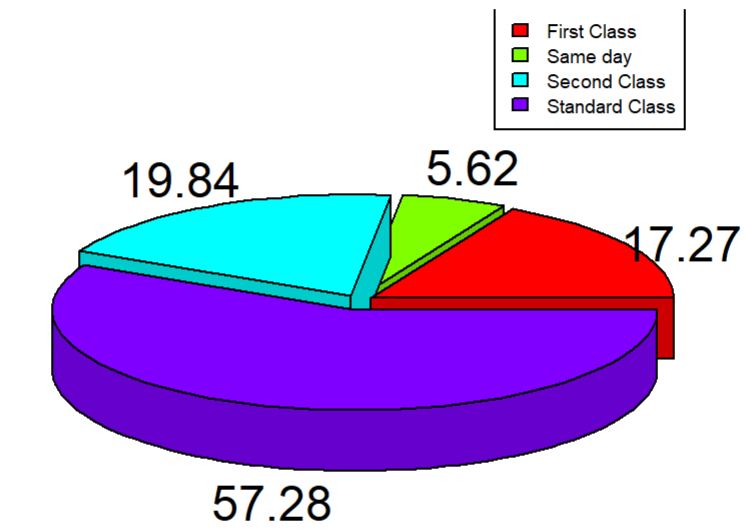
```
p1 <- avg_sale_per_month%>%
  ggplot(aes(x=reorder(month,-mean), y =mean,fil=month.abb[month]))+       #using the ggplot to
 plot the bar graph
  geom_bar(stat='identity', width= 0.85, fill='steelblue')+
  geom_text(aes(label=month.abb[month]),position=position_dodge(width=0.9),hjust=1.5)+
  labs(title="Mean of the sales per month",x="Months of 2020", y="Mean of the sales")+
  #scale_fill_manual(values=c('blue','steelblue'))+
  theme_classic()+
  coord_flip()

p1
```



As we can observe that, January and August have the much higher sales compared to the other months in 2020.

Question 2. Which Shipping Mode is much preferred by the customers?

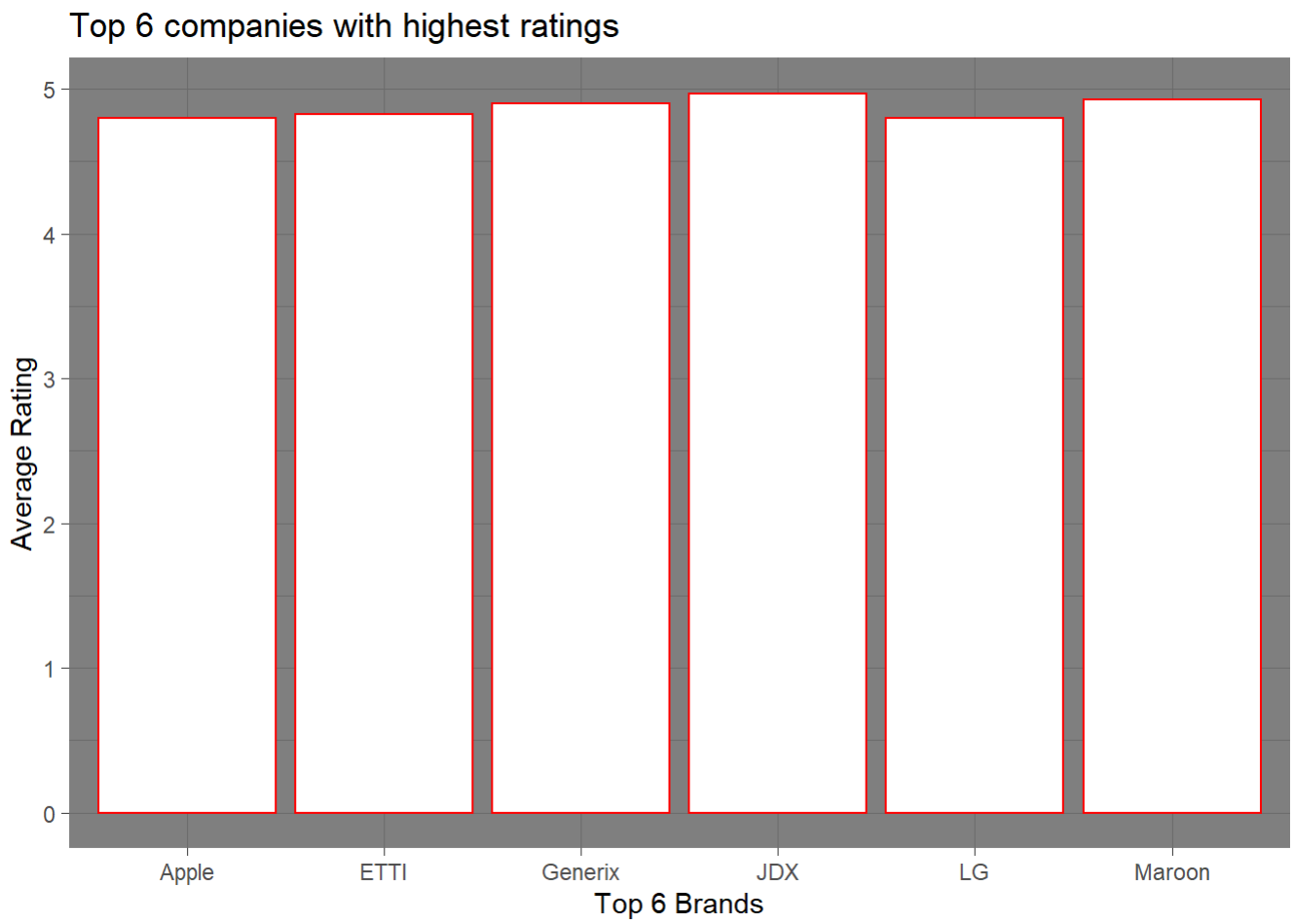
```
pie3D(count_Ship.Mode$prob, labels = count_Ship.Mode$prob,                    #plotting the 3D
 pie graph
      explode = 0.05,
    main= " Probability of Ship Modes for any order",
    col = rainbow(4))
legend("topright", c("First Class","Same day","Second Class","Standard Class"), cex = 0.6, fill
 = rainbow(4))
```



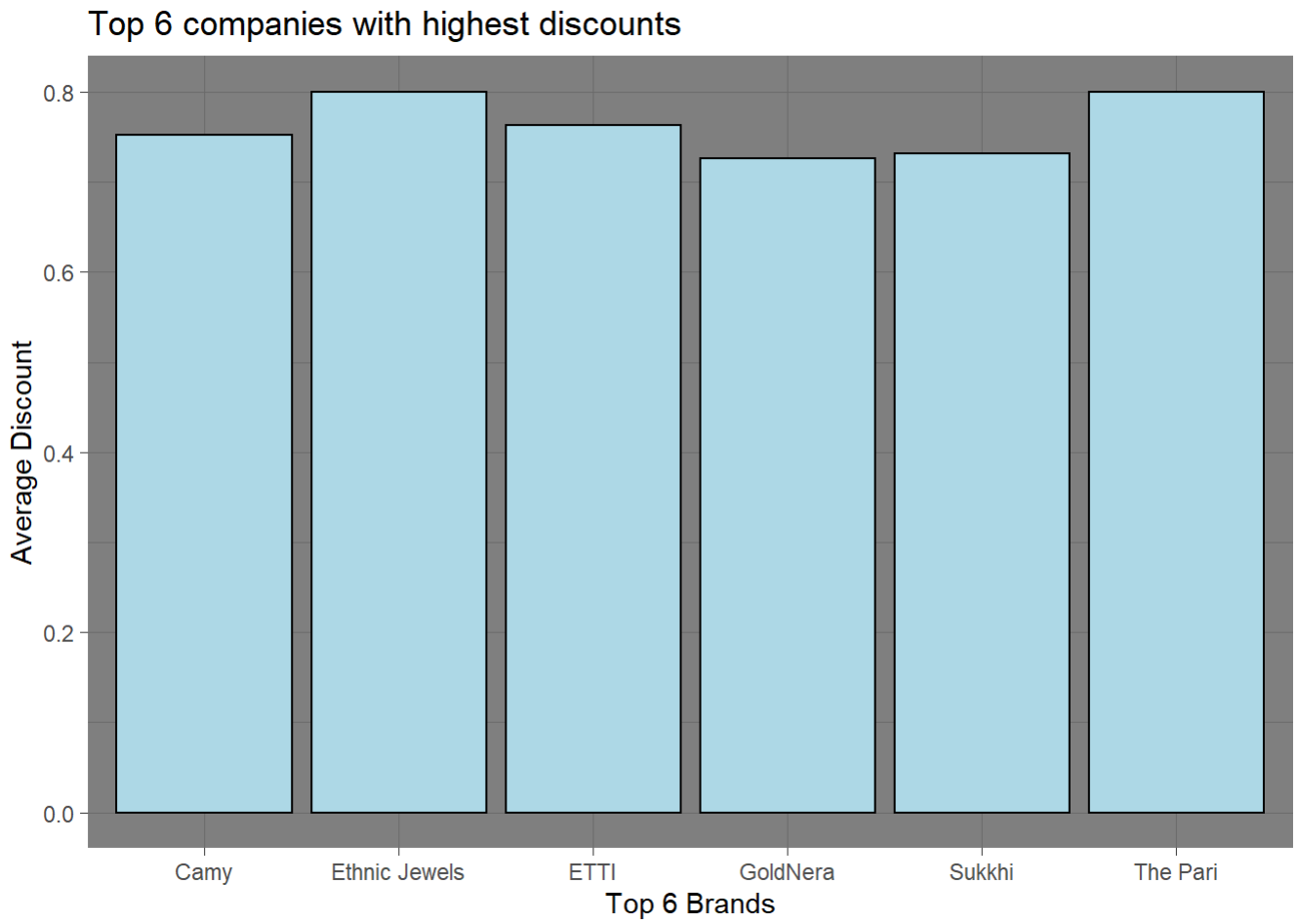
As we can see the most used shipping mode for the deliveries is the Standard Class the percentage of 57.28%.

Question 3. a>Top 6 companies with highest discounts b>Top 6 companies with highest ratings c> Correlation graph plotting of discounts and the ratings of the companies

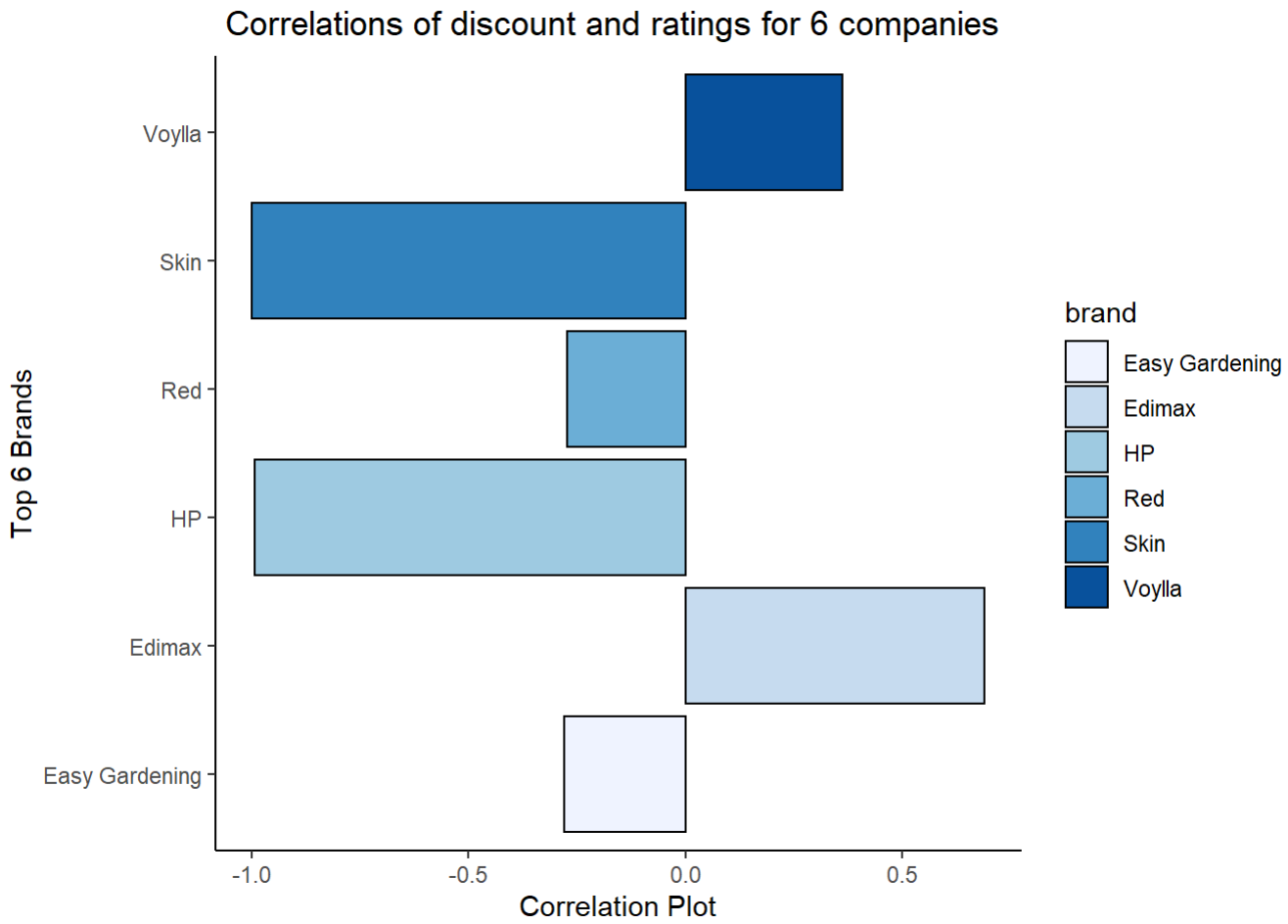
```
rating_company <- relation %>%
  arrange(desc(avg_rating)) %>%                          #sorting the average rating in descendi
ng order
  slice(1:6) %>%                                        #selecting the top 6 rows
  ggplot(aes(x = brand, y = avg_rating))+
  geom_bar(stat = 'identity',color= "red", fill = "white")+
  labs(title="Top 6 companies with highest ratings", x= "Top 6 Brands", y=" Average Rating ")+
  theme_dark()

rating_company
```




```
discount_company <- relation %>%
  arrange(desc(avg_discount)) %>%          #sorting the average discount in descending order
  slice(1:6) %>%                              #selecting the top 6 rows
  ggplot(aes(x = brand, y = avg_discount))+
  geom_bar(stat = 'identity', fill= "lightblue", color = "black")+
  labs(title="Top 6 companies with highest discounts", x= "Top 6 Brands", y= "Average Discount"
 )+
  theme_dark()

discount_company
```




```
relation  <- relation[!grepl(1, relation$correlation),]

correlation_company <- relation %>%
  arrange(desc(correlation)) %>%                                    #sorting the correla
tion in descending order
  slice(1:7) %>%                                                      #selecting th
e top 6 rows
  ggplot(aes(x = brand, y = correlation, fill = brand))+
  geom_bar(stat = 'identity', color = "black" )+
  labs(title=" Correlations of discount and ratings for 6 companies", x="Top 6 Brands", y="Corre
lation Plot")+
  theme_classic()+
  scale_fill_brewer(palette = "Blues")+
  coord_flip()

correlation_company
```

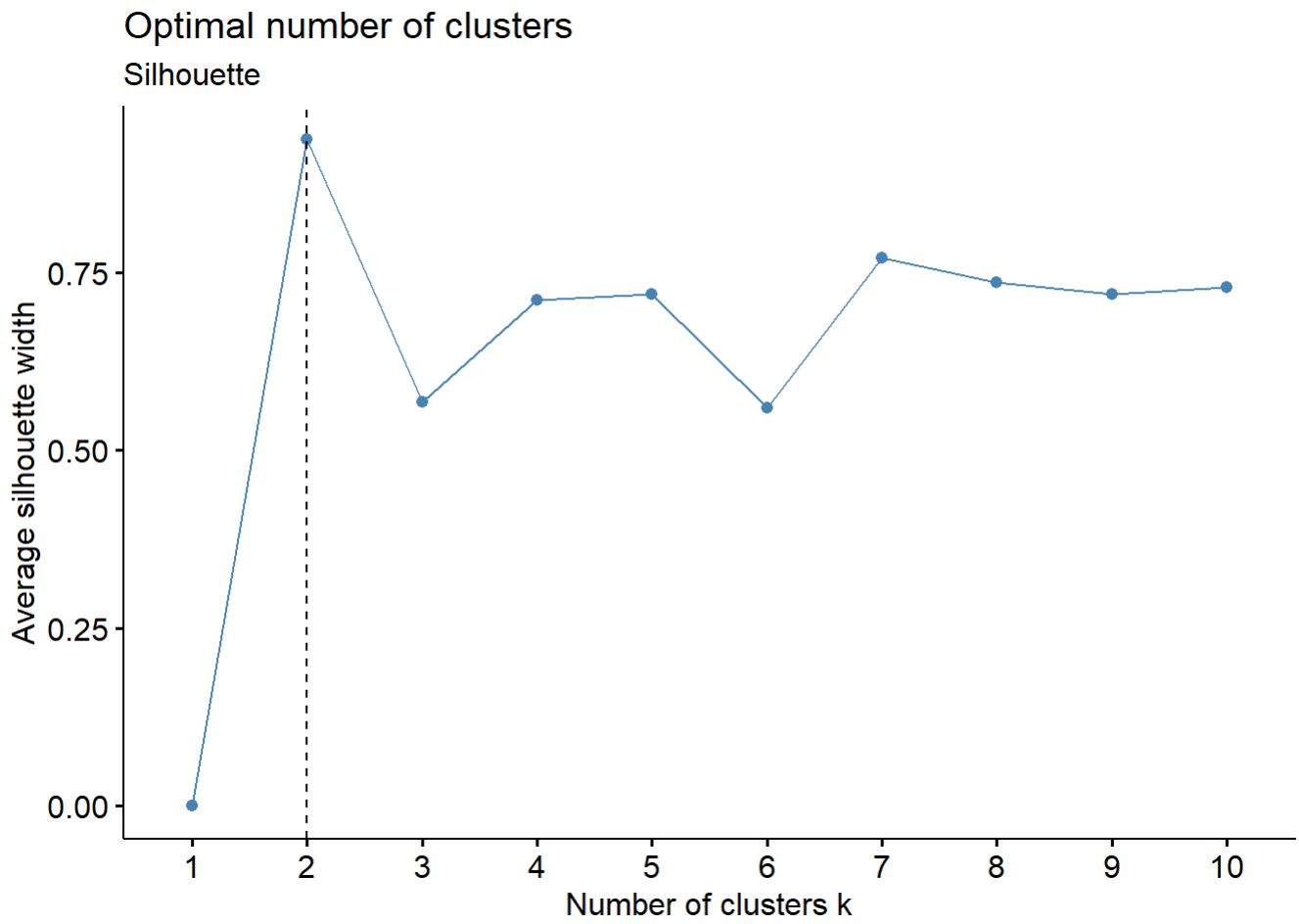


As we can say that ratings of the companies directly implies to the discount given by the company.In this graph we can see that the companies Edimax and Voylla has positive correlation that means the company is giving good discount with the good quality of products. However we are getting negative correlation for companies Easy Gardening, HP, Red, Skin that means customers are compensating with quality of product because of the discount.

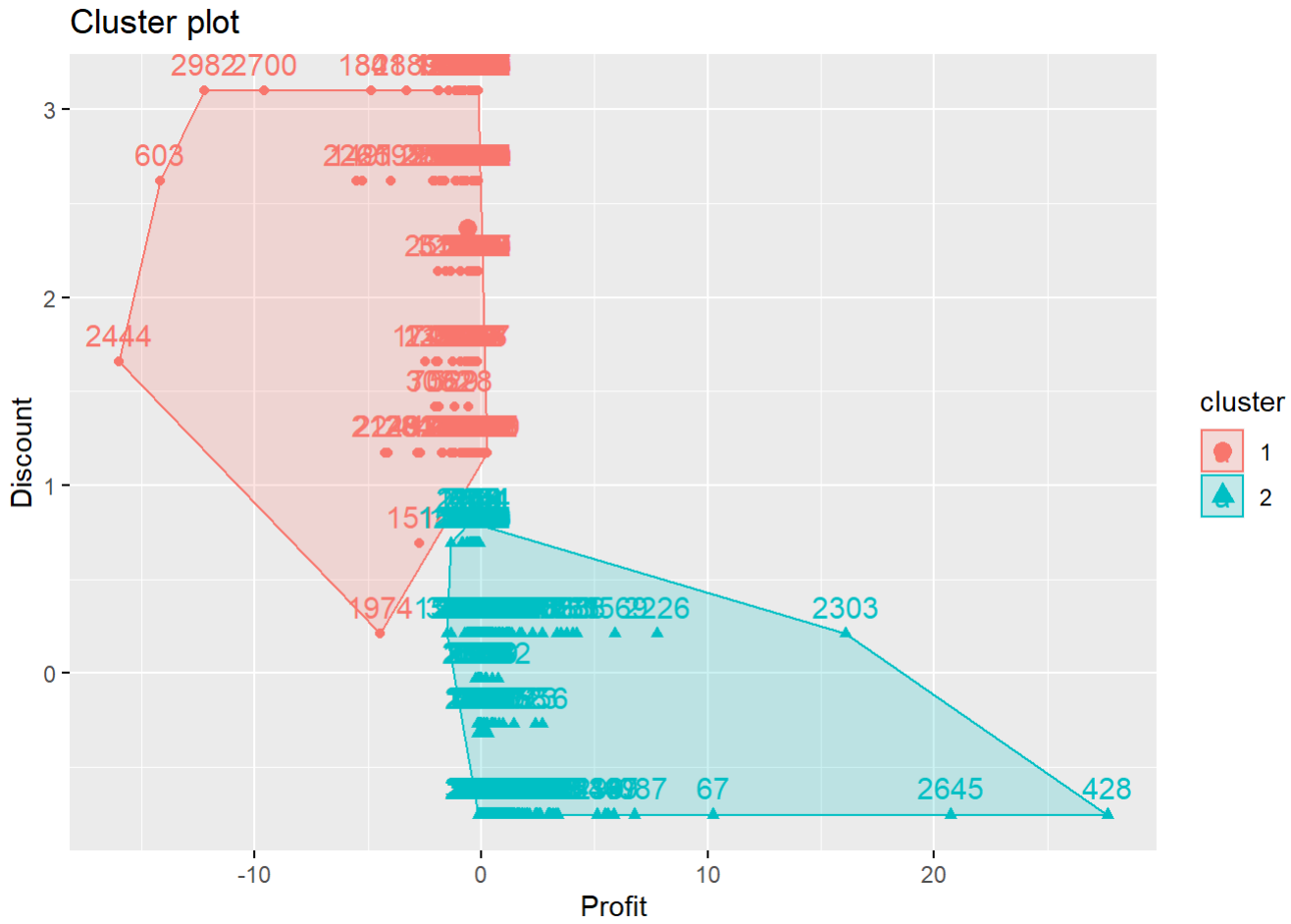
Clustering

Question 4. How the company decides to give the discount on the products to increase their profit using Silhouette Method?

```
y <- data %>%
  select(Profit, Discount)                    #selecting the profit and discount columns

y <- scale(y)             #scaling y



set.seed(123)

fviz_nbclust(y, kmeans, method = "silhouette") + geom_vline(xintercept = 2, linetype = 2)+
#finding the optimal numbers of cluster using silhouette method
  labs(subtitle = "Silhouette")
```



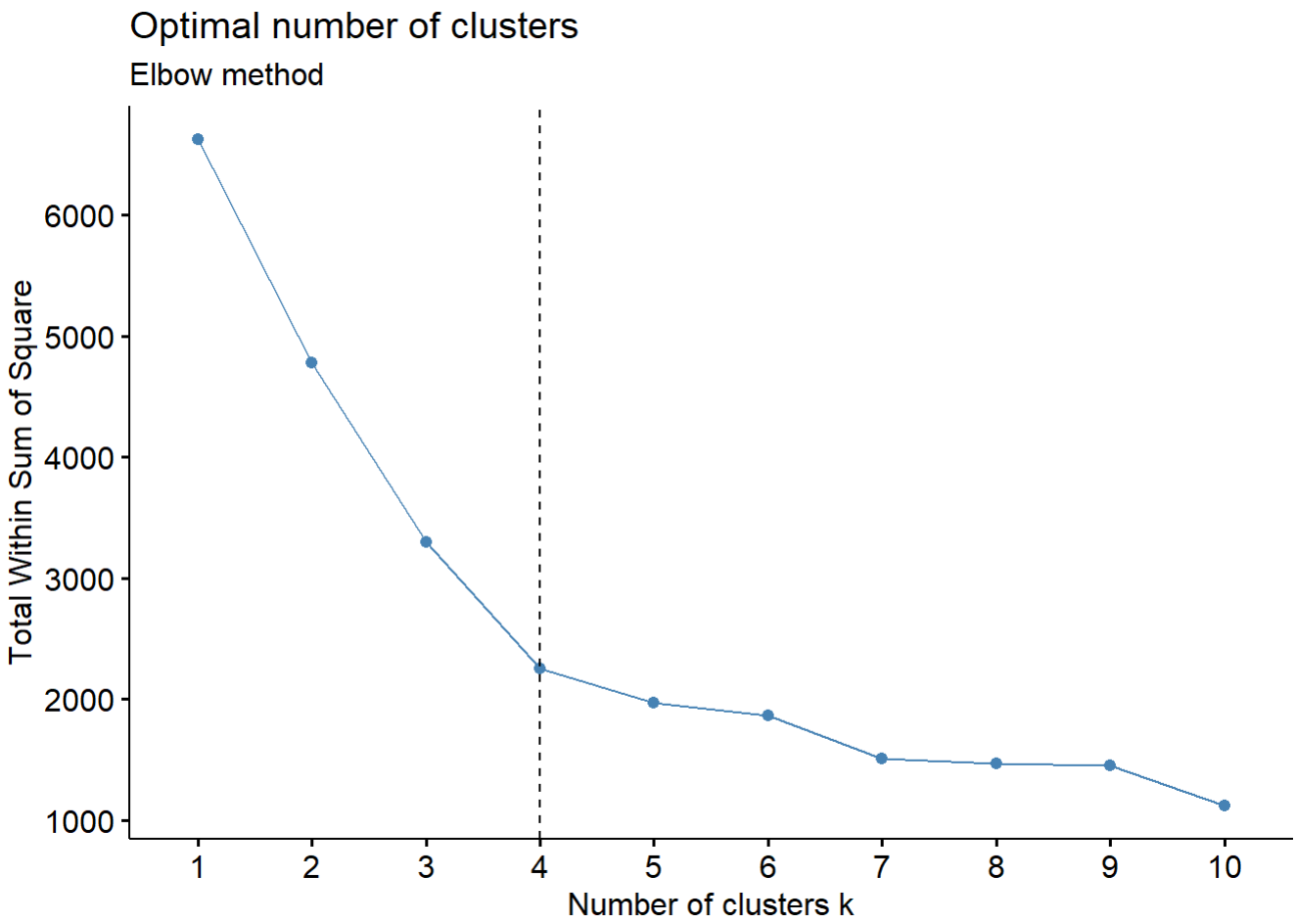

```
km <- kmeans(y, 2, nstart = 25)


Fviz <- fviz_cluster(km,y)
Fviz
```



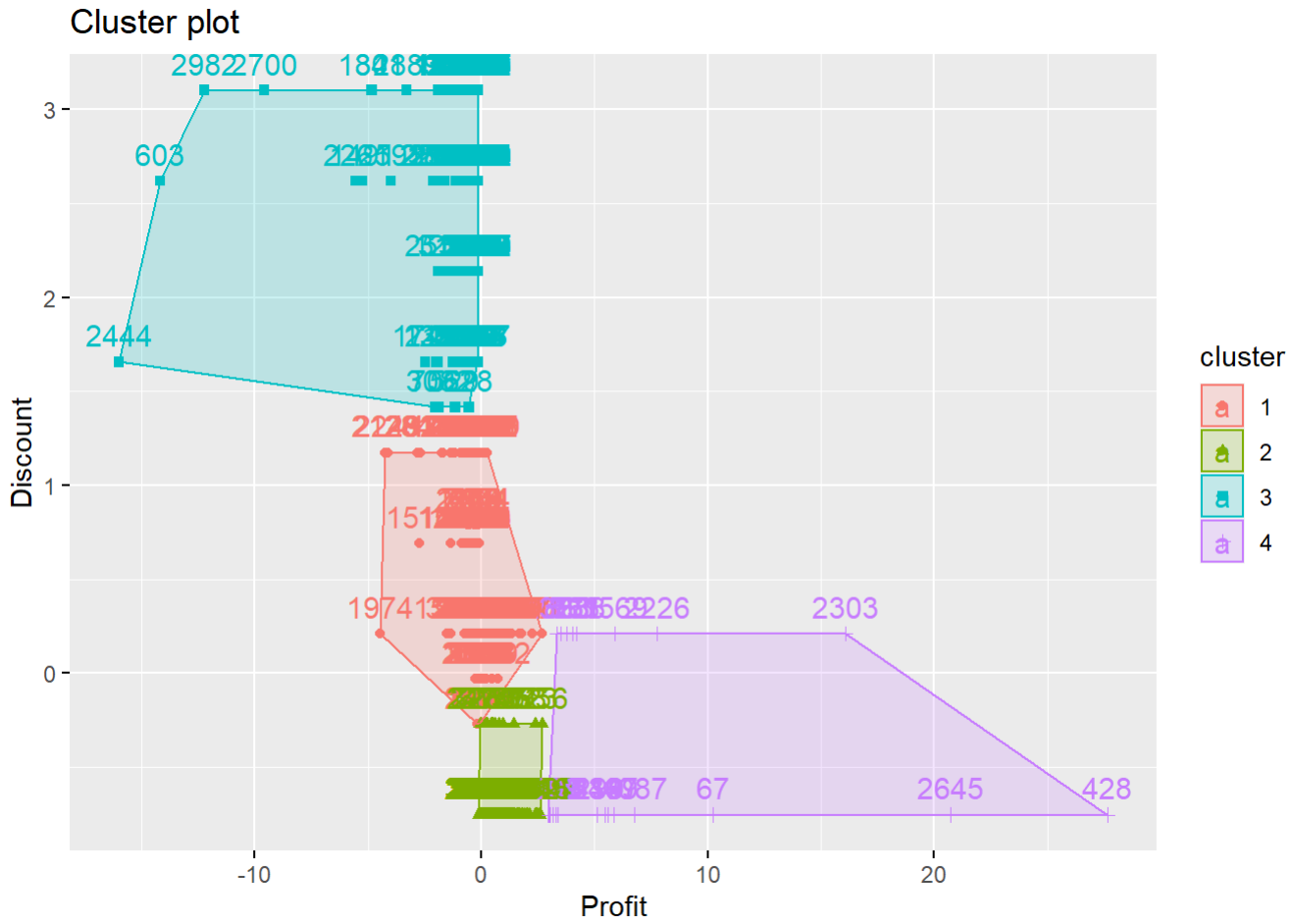
Using the Silhouette Method, we have clustered the Profit and Discount of the US E-commerce into 2 clusters.

Question 5. How the company decides to give the discount on the products to increase their profit using Elbow method?

```
y <- data %>%
  drop_na() %>%
  select(Profit, Discount)

y <- scale(y)

set.seed(123)

fviz_nbclust(y, clara, method = "wss") + geom_vline(xintercept = 4, linetype = 2)+
  labs(subtitle = "Elbow method")
#finding the optimal numbers of cluster using elbow method
```




```
km <- clara(y, 4)



Fviz <- fviz_cluster(km,y)
Fviz
```

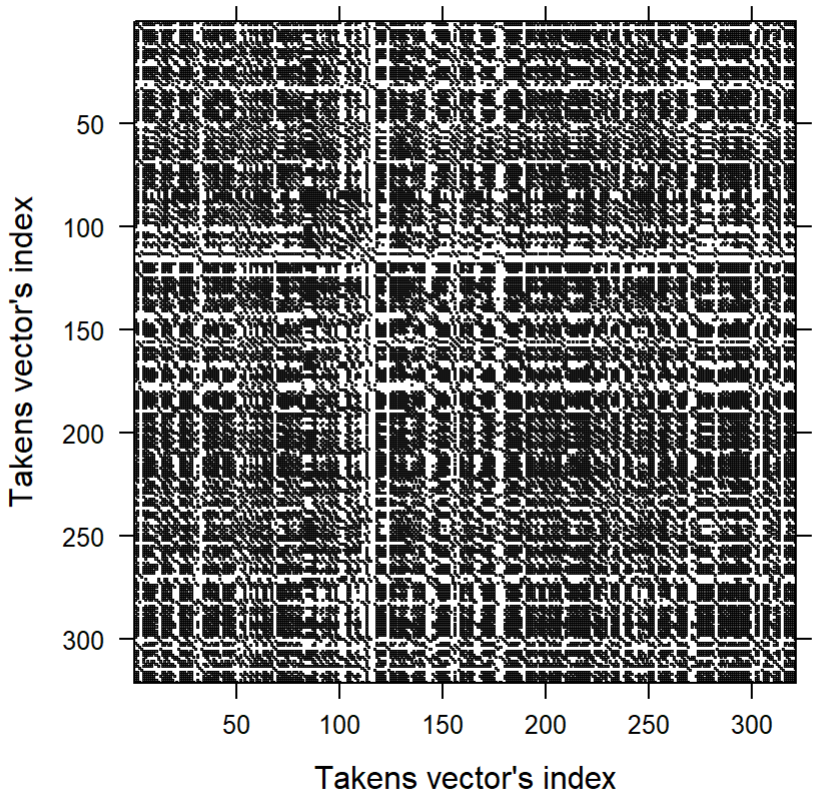


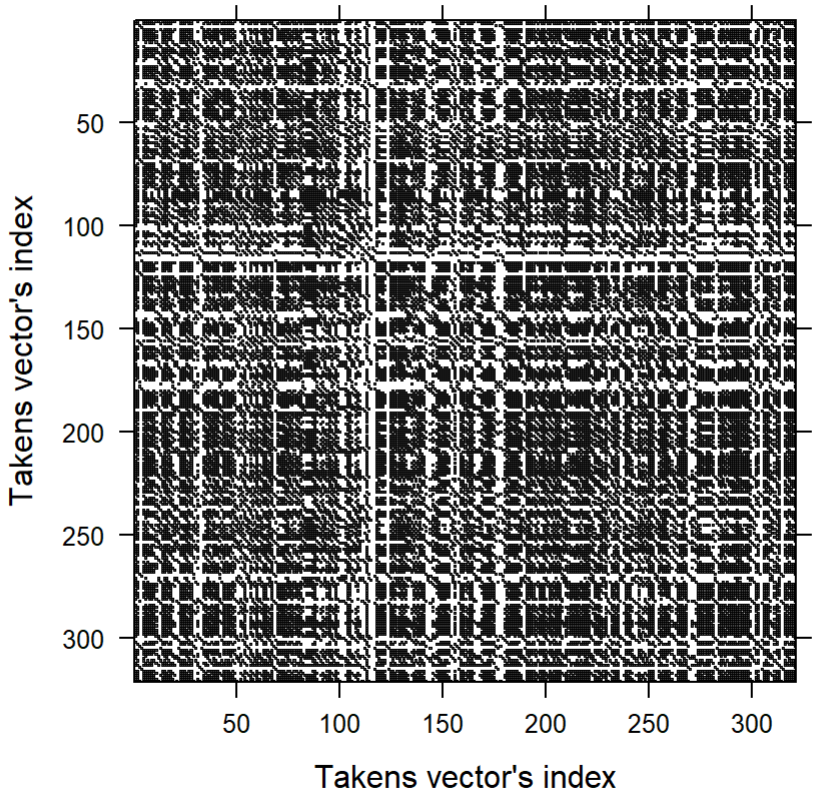
Using the Elbow Method, we have clustered the Profit and Discount of the US E-commerce into 4 clusters.

Time Series Analysis:

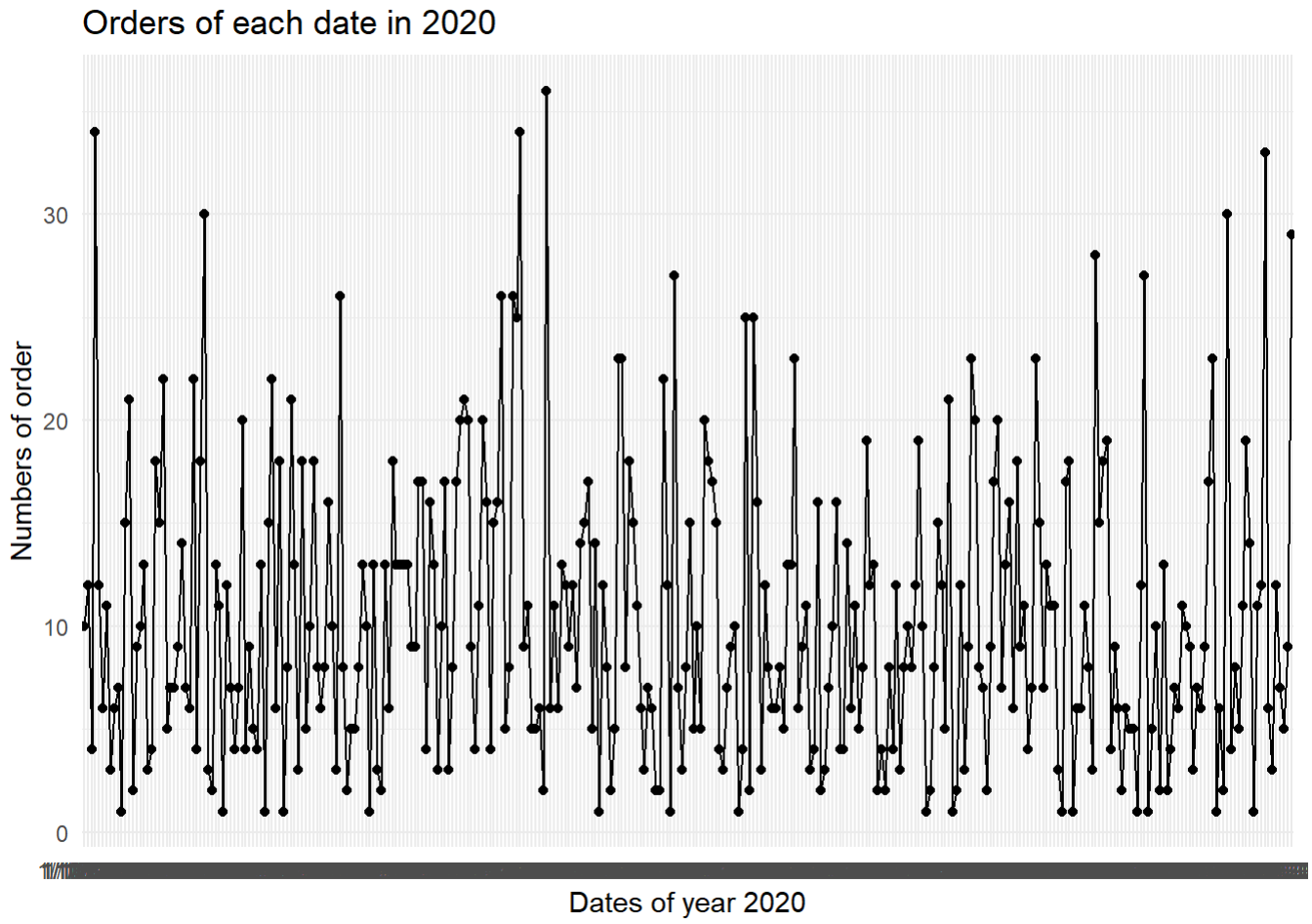
Question 6: How much orders are there on a each date ?

```
orders <- data.frame(table(data$Order.Date))


rqa.analysis <- rqa(time.series = orders$Freq, embedding.dim=2, time.lag=1,
                    radius=10,lmin=2,do.plot=FALSE,distanceToBorder=2)

plot(rqa.analysis)
```


Recurrence plot
Takens vector's index
50
100
150
200
250
300
Takens vector's index
Dimensions: 321 x 321

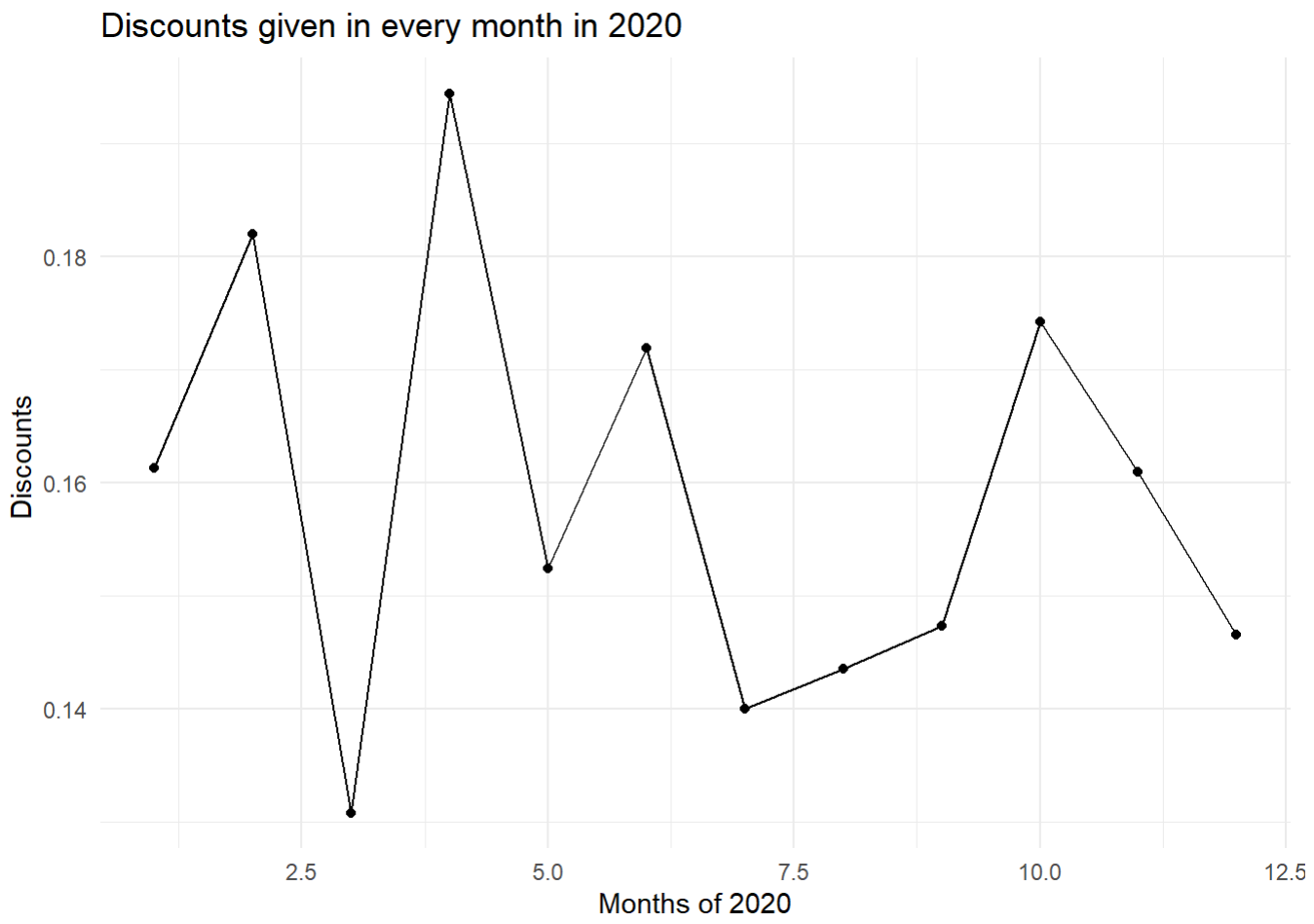

Recurrence plot
Takens vector's index
50
100
150
200
250
300
Takens vector's index
Dimensions: 321 x 321

```
ggplot(data=orders, aes(x=Var1, y=Freq, group=1)) +
  geom_line()+
  geom_point()+
  labs(title="Orders of each date in 2020", x="Dates of year 2020", y="Numbers of order")+
   theme_minimal()
```



Question 7: How much discount have been given in each month during year 2020?

```
discount_d <- data %>%
  group_by(month) %>%
  summarise(avg_discount_per_month = mean(Discount))


ggplot(data=discount_d, aes(x=order(month,-avg_discount_per_month), y=avg_discount_per_month, gr
oup=1)) +
  geom_line()+
  geom_point()+
  labs(title="Discounts given in every month in 2020", x="Months of 2020", y="Discounts ")+
  theme_minimal()
```



From the above results we can say that the companies try to sale the products with good discount only in months when the sales of the company is down.

Text Analysis :

Question 8: Propotion of positive and negative words in description

```
description <- tibble(fd$description, text = fd$description)

description <- description %>%
  unnest_tokens(word, text)
#unnest_token() - to make a tidy data frame of all the words in description

description <- description %>% anti_join(stop_words)
```

```
## Joining, by = "word"
```

```
rd <- data.frame(word=c('additional', 'alisha', 'Solid', 'Specifications', 'Features','of',
                        'for', 'the', 'and','only', 'in','flipkart.com', 'your','on', 'a', 'at',
'this',
                        'rs','to', 'with', 'is'))
#most common words in description

description <- description %>% anti_join(rd)                                    #removing
 most common words
```

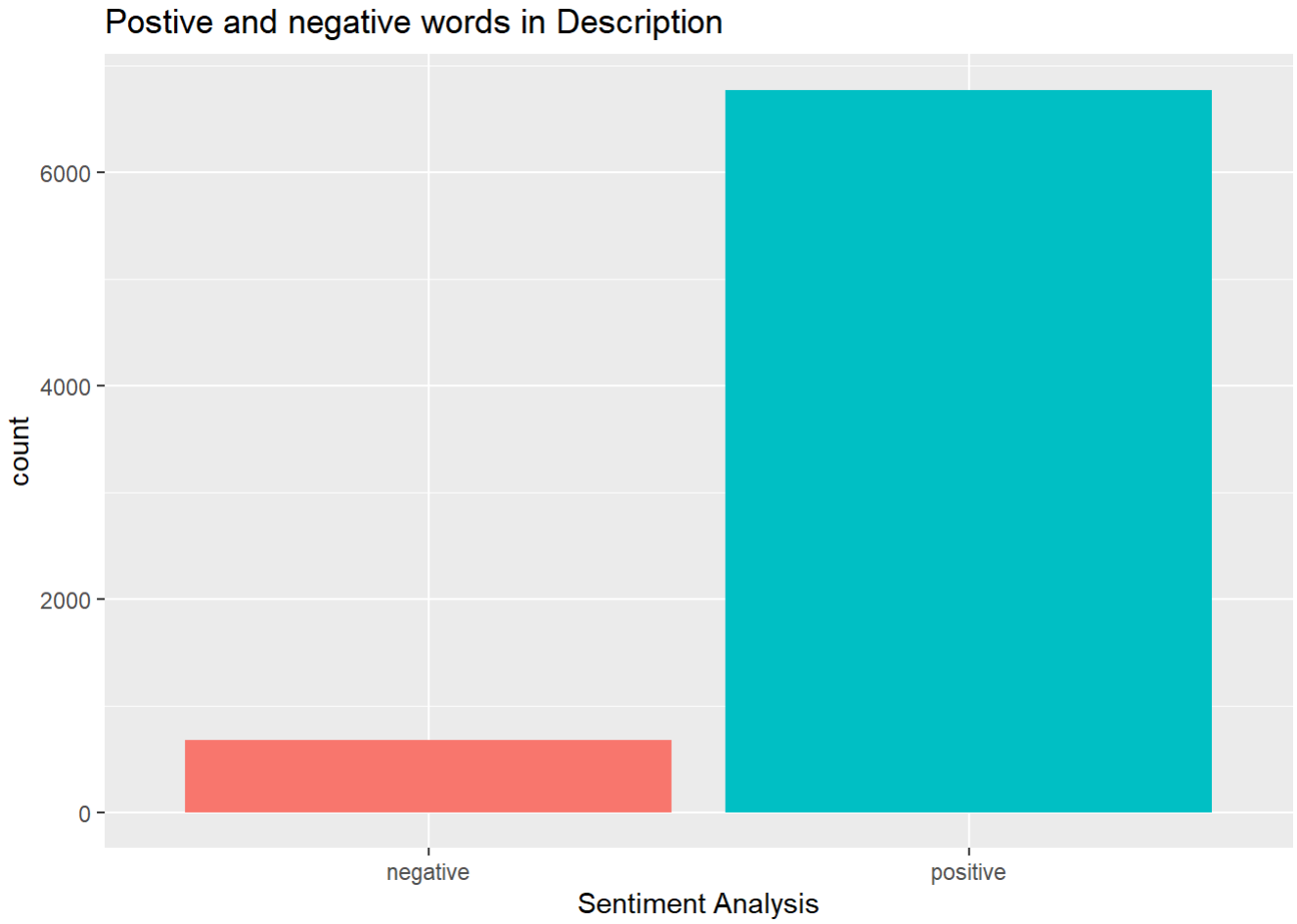
```
## Joining, by = "word"
```

```
pos_neg <- description %>% inner_join(get_sentiments('bing'))                    #find the sentim
ent analysis of the words
```

```
## Joining, by = "word"
```

```
ggplot(pos_neg,aes(x=sentiment,fill=sentiment))+
  geom_bar()+
  labs(title ="Postive and negative words in Description", x= "Sentiment Analysis")+
  guides(fill = F)
```

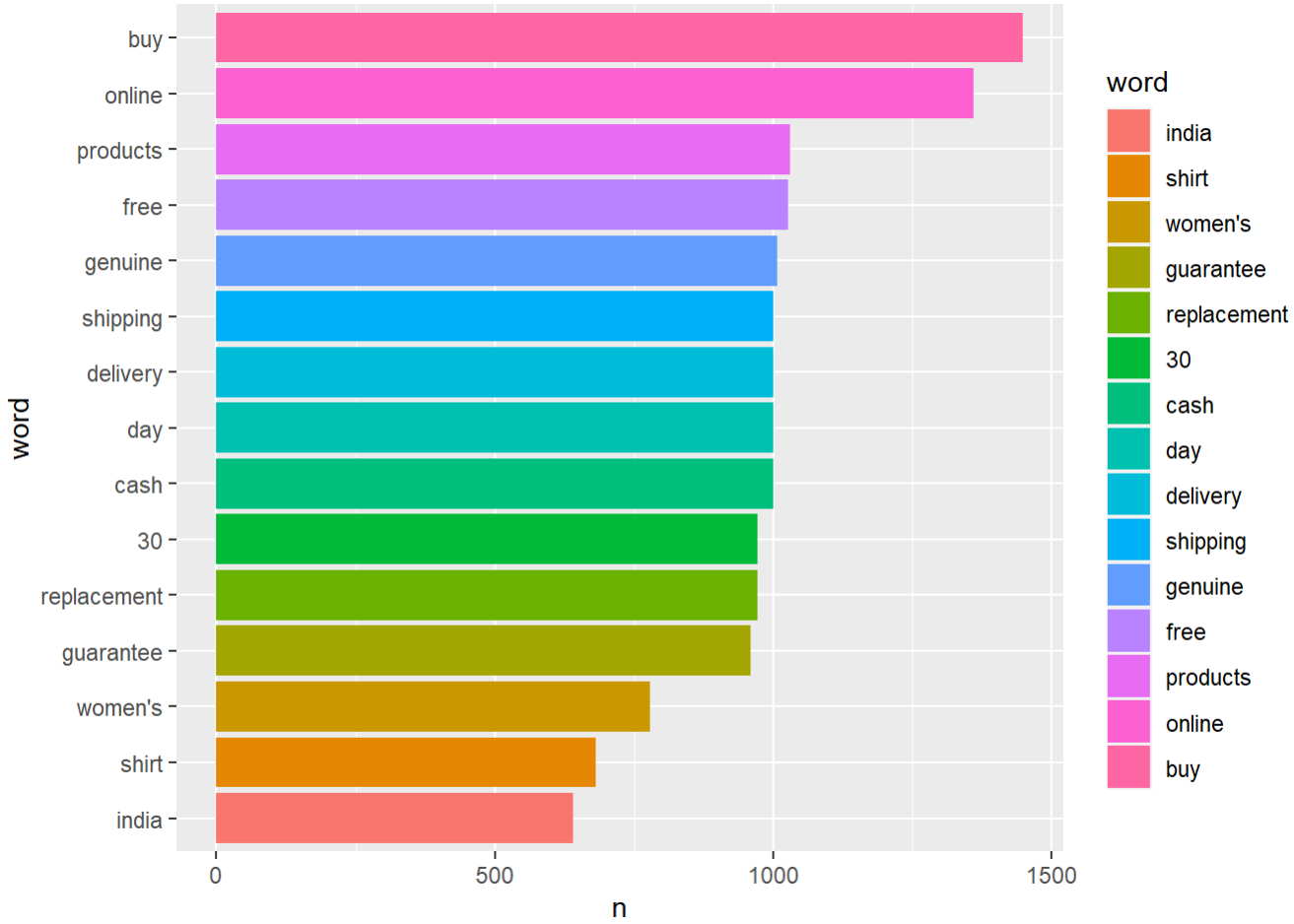
```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



From the above plot we can say that there are more positive words in the description than the negative words.

Question 9: The most common 15 words used in description

```
description %>%
  count(word, sort = TRUE) %>%
  filter(n > 500) %>%  slice(1:15) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word, fill= word)) +
  geom_col()
```




```
  labs(title = "The 15 most common keywords used in description of the product",y ="Words used i
n description", x="count" )+
  theme_minimal()
```

```
## NULL
```

```
wordcloudData1 =
  description%>%  group_by(word)%>%
  summarize(freq = n())%>%
  arrange(desc(freq))%>%
  ungroup()%>%
  data.frame()

set.seed(617)
colors = c("lightblue","lightgreen","red","orange","green")
wordcloud(words = wordcloudData1$word,wordcloudData1$freq,scale=c(2,0.5),max.words = 100,colors=
colors)
```

The most common words used in description to attract the attention of the customer are buy, online, free, genuine, replacement and guarantee.

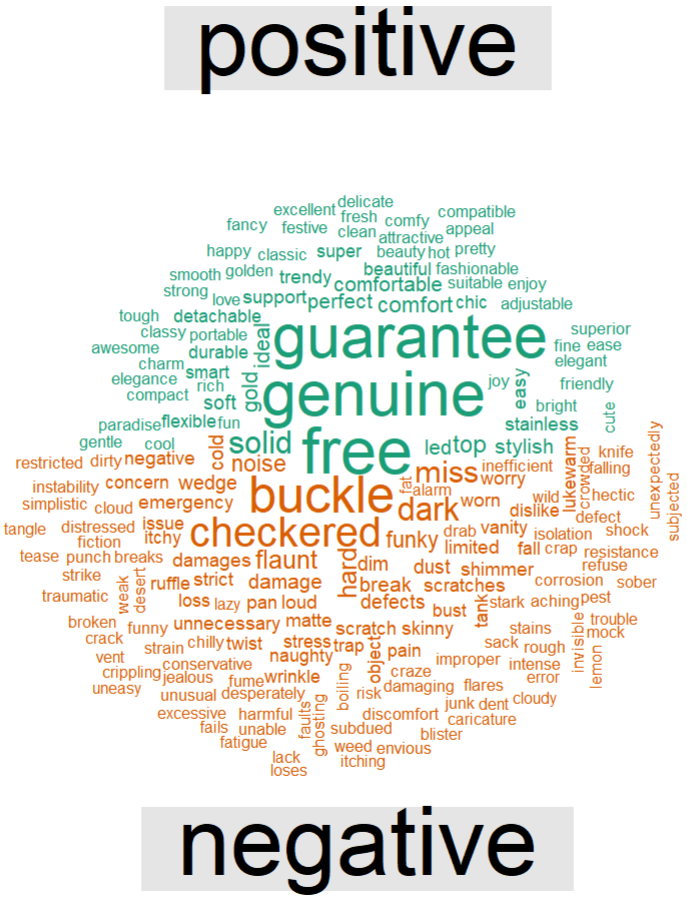
Sentiment Analysis:

Question 10:Segregating the positive and the negative words for the product description.

```
wordcloudData2 =
  description %>%  inner_join(get_sentiments('bing'))%>%
  count(sentiment,word,sort=T)%>%
  spread(key=sentiment,value = n,fill=0)%>%
  data.frame()
```

```
## Joining, by = "word"
```

```
rownames(wordcloudData2) = wordcloudData2[,'word']
wordcloudData2 = wordcloudData2[,c('positive','negative')]

set.seed(617)
comparison.cloud(term.matrix = wordcloudData2,scale = c(2,0.5),max.words = 200, rot.per=0.15)
```



The green color shows the positive words and the most highlighted words are guarantee, genuine and free. While in negative are in orange color and the most highlighted words are lost and matte.

Conclusion: For the above conclusion we can say that : a) When the sales are down the business companies apply more and more sales on the products to increase the sales of their company. b) The most preferred ship mode is the Standard Ship Mode as it is cheap or freely available for the customers. c) There are more sales in the festive season mainly in month of November and December. e) There are more positive words in the description as compared to negative words to attract the customers.