VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127

CSE3020- Data Visualization

J Component-Final Project Report

Customer and Product Segmentation

By

Aayush Shukla	20BCE1500
Pranay Pratik	20BCE1751
Ambati sesha sai sahithya	20BCE1605

B. Tech CSE

Submitted to

Joshan Athanesious J

Assistant Professor, SCOPE, VIT, Chennai

School of Computer Science and Engineering

April 2022

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Bonafide

This Bonafide is to certify that this project report of 'Customer and Product Segmentationin e-Commerce' is the work of Pranay Pratik (20BCE1751), Aayush Shukla(20BCE1500) and Ambati sesha sai sahithya (20BCE1605) who carried out the project work for the subject Data Visualization (CSE3020) under my supervision for the winter semester 2022-23.

Joshan Athanesious J

Assistant Professor

SCOPE, VIT Chennai

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Joshan Athanesious J for** his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work and also for motivatingus to do the project on the topic.

Secondly, we take this opportunity to thank all the faculties of the school for their support and their wisdom imparted to us throughout the course.

Thirdly, we thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

In the end, we would like to thank our friends who believed in us throughout the project and displayed appreciation for my work.

Aayush Shukla 20BCE1500

Pranay Pratik 20BCE1751

Ambati sesha sai sahithya 20BCE1605

TABLE OF CONTENTS

ABSTRACT

"Customer segmentation is the process of putting customers into groups based on things they have in common so that businesses can market to each group in the best way possible." By using the right characteristics to set up the customer segment, companies can find the right customers for their targeted and relevant products.

Consumer classification is a measure of data quality. Most of the time, bad grouping is caused by bad data in the source networks. For individual clients, for example, characteristics like age, gender, and marital status are often used.

If these qualities are not kept in the right way, the segments will be wrong, and the information will be less useful. Users aren't likely to use parts if they don't trust the quality of the data.

Problems with data quality can also be caused by a lack of upkeep and regular scrubbing to keep the data accurate. Another common problem with customer division is that business users don't know what the terms mean and use them incorrectly.

There are many client groups that can be made to help with different business tasks. Users need to be taught to understand the many customer segments that have been made, the real data within the segments that show how they are grouped, and when to use the right customer segmentation for the right analysis.

INTRODUCTION

Management and maintenance of customer relationships have always been important for giving organisations the business data they need to build, manage, and grow valuable long-term relationships with customers. In this day and age, it's becoming more and more important for businesses to treat their customers as their most valuable commodity.

Organisations have a reason to put money into making customer acquisition, retention, and growth plans. Business intelligence is very important because it lets companies use their computer skills to learn more about their customers and create programmes to reach out to them. Using methods like k-means for clustering, customers with similar incomes are grouped together.

Customer segmentation helps the marketing team identify and reach out to different groups of customers who think differently and make purchases in different ways. Customer segmentation helps businesses figure out which customers are different from each other in terms of their tastes, expectations, wants, and other characteristics.

The main goal of customer segmentation is to put together groups of people with similar interests so that the marketing team can come up with a good marketing plan.

Clustering is a way to find patterns in huge amounts of raw, unorganised data. It is a repetitive process.

Clustering is a type of creative data mining that is used in many fields, such as machine learning, classification, and pattern recognition.

LITERATURE SURVEY

1)

TITLE-

Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining

AUTHOR-

Daqing Chen, Sai Laing Sain and Kun Guo

GIST-

Many small online retailers and new entrants to the online retail sector are keen to practice

data mining and consumer-centric marketing in their businesses yet technically lack the necessary knowledge and expertise to do so.

In this article a case study of using data mining techniques in customer-centric business intelligence for an online retailer is presented.

The main purpose of this analysis is to help the business better understand its customers and therefore conduct customer-centric marketing more effectively.

On the basis of the Recency, Frequency, and Monetary model, customers of the business have been segmented into various meaningful groups using the k-means clustering algorithm and decision tree induction, and the main characteristics of consumers in each segment have been clearly identified.

Accordingly, a set of recommendations is further provided to the business on consumer

centric marketing. SAS Enterprise Guide and SAS Enterprise Miner are used in the present study.

RESULT-

A case study has been presented in this article to demonstrate how customer-centric business intelligence for online retailers can be created by means of data mining techniques.

The distinct customer groups characterized in the case study can help the business better understand its customers in terms of their profitability, and accordingly, adopt appropriate marketing strategies for different consumers.

It has been shown in this analysis that there are two steps in the whole data mining process that are very crucial and the most time-consuming: data preparation and model interpretation and evaluation.

Further research for the business includes: conducting association analysis to establish customer buying patterns with regard to which products have been purchased together frequently by which customers and which customer groups; enhancing the merchant's web site to enable a consumer's shopping activities to be captured and tracked instantaneously and accurately; and predicting each customer's lifecycle value to quantify the level of diversity of each customer.

REFERENCE–

https://link.springer.com/content/pdf/10.1057/dbm.2012.17.pdf

2)

TITLE-

Integration K-Means Clustering Method and Elbow Method for Identification of the Best Customer Profile Cluster

AUTHOR-

M.A. Syakur, B.K. Khotimah, E.M.S Rochman, B.D. Satoto

GIST-

Clustering is a data mining technique used to analyse data that has variations and the number of lots. Clustering was process of grouping data into a cluster, so they contained data that is as similar as possible and different from other cluster objects.

SMEs Indonesia has a variety of customers, but SMEs do not have the mapping of these customers so they did not know which customers are loyal or otherwise.

Customer mapping is a grouping of customer profiling to facilitate analysis and policy of SMEs in the production of goods, especially batik sales. Researchers will use a combination of K-Means method with elbow to improve efficient and effective k-means performance in processing large amounts of data.

K-Means Clustering is a localized optimization method that is sensitive to the selection of the starting position from the midpoint of the cluster. So choosing the starting position from the midpoint of a bad cluster will result in K-Means Clustering algorithm resulting in high errors and poor cluster results.

The K-means algorithm has problems in determining the best number of clusters. So, Elbow looks for the best number of clusters on the K-means method. Based on the results obtained from the process in determining the best number of clusters with elbow method can produce the same number of clusters K on the amount of different data.

The result of determining the best number of clusters with elbow method will be the default for characteristic process based on case study. Measurement of k-means value of k-means has resulted in the best clusters based on SSE values on 500 clusters of batik visitors. The result shows the cluster has a sharp decrease is at K = 3, so K as the cut-off point as the best cluster.

RESULT-

The results obtained from the process in determining the best number of clusters with elbow and K Means methods that the determination of the best number of clusters can produce the same number of clusters K on the amount of different data. The result of determining the best number of clusters with elbow method will be the default for characteristic process based on case study.

REFERNCE –

https://iopscience.iop.org/article/10.1088/1757-899X/336/1/012017/pdf

3)

TITLE-

Customer segmentation and strategy development based on customer lifetime value: A case study

AUTHOR-

Su-YeonKim, Tae-SooJung, Eui-HoSuh

GIST-

Customer Relationship Management (CRM) has become a leading business strategy in highly competitive business environment. CRM can be viewed as 'Managerial efforts to manage business interactions with customers by combining business processes and technologies that seek to understand a company's customers.

Companies are becoming increasingly aware of the many potential benefits provided by CRM. Some potential benefits of CRM are as follows:

(1) Increased customer retention and loyalty,

(2) Higher customer profitability,

(3) Creation value for the customer,

(4) Customization of products and services,

(5) Lower process, higher quality products and services

In this paper, we propose a framework for analysing customer value and segmenting customers based on their value. After segmenting customers based on their value, strategies building according to customer segment will be illustrated through a case study on a wireless telecommunication company.

RESULT-

Since the increased importance is placed on customer satisfaction in today's business environment, many firms are focusing on the notion of customer loyalty and profitability to increasing market share and customer satisfaction. CRM, the core business concept to enhance customer relationship, is emerging as core competence of a firm. Building successful CRM of a firm starts from identifying customers' true value and loyalty since customer value can provide basic information to deploy more.

Li, Zeying [1] have proposed a method in which a retail supermarket was taken as research object, and data mining methods was used to retail enterprise customer segments, and then association rules obtained using Apriori algorithm were used to different groups of customers and get rules about customer characteristics to make customer characteristic analysis efficiently.

Finally, the author gave some references to the supermarket's marketing and management work, which helped in understanding it in detail. Data mining was used efficiently to deal with the large number of historical and current data, from the database to find some potential, useful and valuable information for the retail stores which help us target customers.

Wang, Zhenyu, Yi Zuo, Tieshan Li, CL Philip Chen, and Katsutoshi Yada [2] have

analyzed customer segmentation based on broad learning system which provides an alternative view of learning in deep structure.

Firstly, in addition to customer purchasing behavior, RFID (Radio Frequency Identification) data was also included, which can accurately represent the consumers' in-store behavior.

Secondly, this paper used Broad Learning System (BLS) to analyze the consumer segmentation. BLS is one of the finest machine learning techniques, and quite efficient and effective for classification tasks.

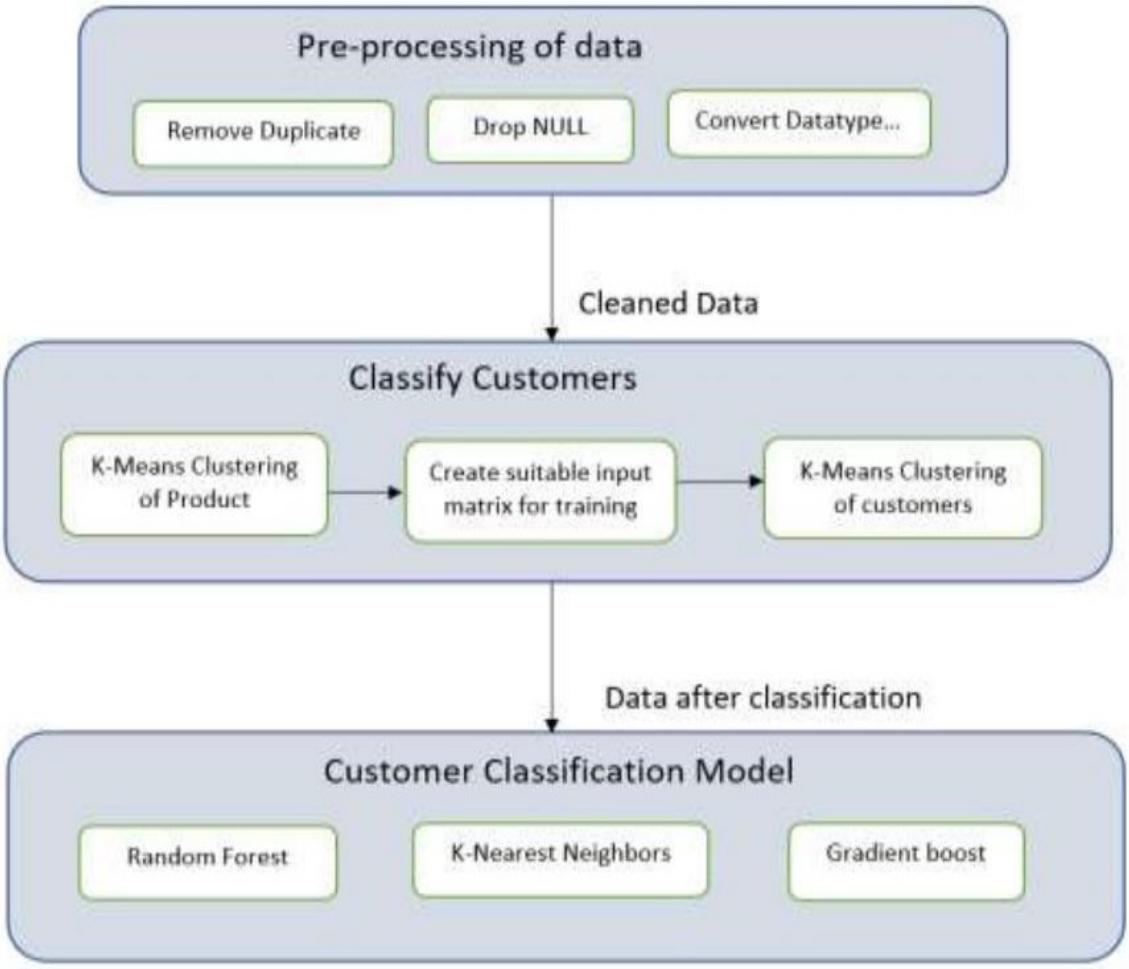
Thirdly, the customer behaviour data used in this paper was collected from a real-world supermarket in Japan. Customer segmentation was considered as a multi-label classification problem based on both of POS data and RFID data.

Kansal, Tushar, Suraj Bahuguna, Vishal Singh, and Tanupriya Choudhury [3] performed customer segmentation using K-means clustering. A python program was developed and the program was trained by applying standard scaler onto a dataset having two features of 200 training sample taken from local retail shop. Both the features are the average of the amount of shopping by customers and average of the customer's visit into the shop annually. By applying clustering, 5 segments of cluster were formed labelled as Careless, Careful, Standard, Target and Sensible customers. However, the authors got two new clusters on applying mean shift clustering labelled as High buyers and frequent visitors and High buyers and occasional visitors.

Bhade Kalyani, Vedanti Gulalkari, Nidhi Harwani and Sudhir N Dhage have proposed a systematic approach for targeting customers and providing maximum profit to the organizations. An important initial step was to analyze the data of sales acquired from the purchase history and determine the parameters that have the maximum correlation.

Based on respective clusters, proper resources can be assigned towards profitable customers using machine learning algorithms. K-Means clustering was used for customer segmentation and Singular Value Decomposition was used for providing appropriate recommendations to the customers. This paper also deals with the drawbacks of the recommender system like sparsity, cold start problem etc and how they can be overcome.

ARCHITECTURAL DESIGN



System architecture for customer segmentation.

The data has been extracted from the kaggle and converted to raw data and then the raw data is been refined to make various modules using R Studio and then the module in the form of rows are used for analysis.

But in the project, we directly got the inbuilt dataset from Kaggle that is the modules profiles with thousands of modules in rows and the module attributes in columns. We performed the analysis on the customer segmentation profiles and got the expected result.

Automatic collection of customer data across different customer touchpoints (sales channels, social media, customer surveys, customer service centers, etc.) and transaction channels (online and offline stores, marketplaces) for analysis and segmentation.

Typically, we are already collecting this sort of data when our customers enter their payment details upon check out, sign up for our newsletter, or voluntarily hand it over in order to receive a product, service, or incentive.

The aim of the quantitative data game is to understand the decision-making process of our customers as they interact with your company. What led them to discover our business? Which channel drives the most conversions?

Channel-specific tools are available throughout the customer lifecycle and should be tailored to measuring your marketing goals and strategy.

Obtaining high-quality descriptive data is no easy feat and requires additional ingenuity. Companies typically turn to in-depth questionnaires for their data collection, which dive into discovering seasonal growth and decline, buying behaviors, and lifespan of the customer cycle.

Consolidation of customer profiles across various data source systems to get a 360° view of each customer.

REQUIREMENTS

1) Jupyter Notebook
2) Python 3.8
3) MATPLOTLIB
4) Pandas
5) Numpy
6) Plotly

DATASETS

First dataset consisting of attributes of all app related information such as-

The dataset (Mall_Customers) comprises of

1. Customer Id
2. gender
3. age
4. annual income (k)
5. Spending Score (1-100)
6. Number of items bought
7. Invoice id, Branch, City
8. Customer type
9. product line
10. unit price
11. quantity
12. tax 5%
13. total
14. Date
15. time
16. payment
17. cogs
18. gross margin percentage
19. gross income
20. rating.

This dataset is taken from Kaggle.

CustomerID: Unique ID assigned to the customer

Gender: Gender of the customer

Age: Age of the customer

Annual Income (k$): Annual Income of the customer

Spending Score: Score assigned by the mall based on customer behavior and spending nature

Number.of.items.bought : Number of items bought by the customer.

Invoice id | Computer generated sales slip invoice identification number

Branch | Branch of supercenter (A/B/C).

City | Location of supercenters

Customer type | Type of customers,Members/Normal with or without member card.

Product line | General item categorization groups Electronic accessories/ Fashion accessories/

Food and beverages/Health and beauty/Home and lifestyle/Sports and travel

Unit price | Price of each product in $

Quantity | Number of products purchased by customer

Tax | 5% tax fee for customer buying

Total | Total price including tax

Date | Date of purchase (MM/DD/YYYY)

Time | Purchase time (10am to 9pm)

Payment | Payment used by customer for purchase (Cash/Credit card/Ewallet)

COGS | Cost of goods sold

Gross margin percentage| Gross margin percentage

Gross income | Gross income

Rating | Customer stratification rating on their overall shopping experience (On a scale of 1 to 10)

MODULES

1) Mall Customers Dataset

In this module, we have imported the dplyr package and read the mall_customers.csv. We have printed the total number of customers available in the dataset and performed analysis on product as well as customers.

2) Data cleaning

In this module, we cleaned the dataset by removing the unnecessary columns such as Current Version, Last Updated. We have removed the missing data from the date column and we have removed any missing values from the dataset and printed the structure of the dataset.

3) Correcting data types

In this module, we imported the numpy library and converted the installs column values to float. We also converted the price values to float. Because, we wanted the datatype of the two columns to be float for calculations and graphs. We printed the structure of the dataset.

4) Depicting the ratio between male and female customers

Customer Gender Visualization – In this, we will create a barplot and a piechart to show the gender distribution across our cutomer_data dataset.

5) Distribution of Age Group of customers

In this module, we will create histogram and boxplot to analyse the distribution of age group amongst all the customers in our dataset.

6)Descriptive analysis of Age based on the sales

We will perform visualization and analysis of age based on the sales, what products attracts what age group and how products can be segmented into various age groups.

7) Analysis of Annual income vs items purchased

In this section of the R project, we will create visualizations to analyze the annual income of the customers. We will plot a histogram and then we will proceed to examine this data using a density plot.

8) Comparison Analysis of total item sold Vs Branch

In this module, we will perform analysis on all the items sold in the branches, create visualisations and plots and getting data on what items are sold in each branch, total items sold in the branches.

9) Descriptive analysis of product line w.r.t each Branch

In this module, we will perform analysis on products sold in each branch and collect data on the selling of the products in each and every branch and create line graphs to analyse selling rate per branch.

10) Density plot of Rating Vs Branch

In this module, we will create visualizations and density plots between Rating of the products given by the customers and branch, this is a relationship between rating and branches.

11) Analysis of Product sold date Vs Product Line

In this module, we will perform analysis on the product sold date and the product line

12) Performance Metrics

We imported the matplotlib, numpy, sklearn libraries and for the columns Installs and Reviews, we fit the columns into the linear regression model and predicted the reviews for an app with total number of installs 10000.

IMPLEMENTATION AND PERFORMANCE ANALYSIS

```
str(Mall_Customers)
```

```
## 'data.frame':    200 obs. of  22 variables:
##  $ CustomerID             : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Gender                 : chr  "Male" "Male" "Female" "Female" ...
##  $ Age                    : int  19 21 20 23 31 22 35 23 64 30 ...
##  $ Annual.Income..k..     : int  15 15 16 16 17 17 18 18 19 19 ...
##  $ Spending.Score..1.100. : int  39 81 6 77 40 76 6 94 3 72 ...
##  $ Number.of.items.bought : int  15 15 16 16 17 17 18 18 19 19 ...
##  $ Invoice.ID             : chr  "750-67-8428" "226-31-3081" "631-41-3108" "123-19-1176" ...
##  $ Branch                 : chr  "A" "C" "A" "A" ...
##  $ City                   : chr  "Yangon" "Naypyitaw" "Yangon" "Yangon" ...
##  $ Customer.type          : chr  "Member" "Normal" "Normal" "Member" ...
##  $ Product.line           : chr  "Health and beauty" "Electronic accessories" "Home and lifestyle" "Health and beaut
...
##  $ Unit.price             : num  74.7 15.3 46.3 58.2 86.3 ...
##  $ Quantity               : int  7 5 7 8 7 7 6 10 2 3 ...
##  $ Tax.5.                 : num  26.14 3.82 16.22 23.29 30.21 ...
##  $ Total                  : num  549 80.2 340.5 489 634.4 ...
##  $ Date                   : chr  "1/5/2019" "3/8/2019" "3/3/2019" "1/27/2019" ...
##  $ Time                   : chr  "13:08" "10:29" "13:23" "20:33" ...
##  $ Payment                : chr  "Ewallet" "Cash" "Credit card" "Ewallet" ...
##  $ cogs                   : num  522.8 76.4 324.3 465.8 604.2 ...
##  $ gross.margin.percentage: num  4.76 4.76 4.76 4.76 4.76 ...
##  $ gross.income           : num  26.14 3.82 16.22 23.29 30.21 ...
##  $ Rating                 : num  9.1 9.6 7.4 8.4 5.3 4.1 5.8 8 7.2 5.9 ...
```

```
names(Mall_Customers)
```

```
##  [1] "CustomerID"              "Gender"
##  [3] "Age"                     "Annual.Income..k.."
##  [5] "Spending.Score..1.100."  "Number.of.items.bought"
##  [7] "Invoice.ID"              "Branch"
##  [9] "City"                    "Customer.type"
## [11] "Product.line"            "Unit.price"
## [13] "Quantity"                "Tax.5."
## [15] "Total"                   "Date"
## [17] "Time"                    "Payment"
## [19] "cogs"                    "gross.margin.percentage"
## [21] "gross.income"            "Rating"
```

```
dim(Mall_Customers)
```

```
## [1] 200  22
```

```
head(Mall_Customers)
```

```
##   CustomerID Gender Age Annual.Income..k.. Spending.Score..1.100.
## 1          1   Male  19                 15                     39
## 2          2   Male  21                 15                     81
## 3          3 Female  20                 16                      6
## 4          4 Female  23                 16                     77
## 5          5 Female  31                 17                     40
## 6          6 Female  22                 17                     76
##   Number.of.items.bought  Invoice.ID Branch      City Customer.type
## 1                     15 750-67-8428      A    Yangon        Member
## 2                     15 226-31-3081      C Naypyitaw        Normal
## 3                     16 631-41-3108      A    Yangon        Normal
## 4                     16 123-19-1176      A    Yangon        Member
## 5                     17 373-73-7910      A    Yangon        Normal
## 6                     17 699-14-3026      C Naypyitaw        Normal
##             Product.line Unit.price Quantity  Tax.5.    Total      Date  Time
## 1      Health and beauty      74.69        7 26.1415 548.9715  1/5/2019 13:08
## 2 Electronic accessories      15.28        5  3.8200  80.2200  3/8/2019 10:29
## 3     Home and lifestyle      46.33        7 16.2155 340.5255  3/3/2019 13:23
## 4      Health and beauty      58.22        8 23.2880 489.0480 1/27/2019 20:33
## 5      Sports and travel      86.31        7 30.2085 634.3785  2/8/2019 10:37
## 6 Electronic accessories      85.39        7 29.8865 627.6165 3/25/2019 18:30
##       Payment   cogs gross.margin.percentage gross.income Rating
## 1     Ewallet 522.83                4.761905      26.1415    9.1
## 2        Cash  76.40                4.761905       3.8200    9.6
## 3 Credit card 324.31                4.761905      16.2155    7.4
## 4     Ewallet 465.76                4.761905      23.2880    8.4
## 5     Ewallet 604.17                4.761905      30.2085    5.3
## 6     Ewallet 597.73                4.761905      29.8865    4.1
```

```
summary(Mall_Customers)
```

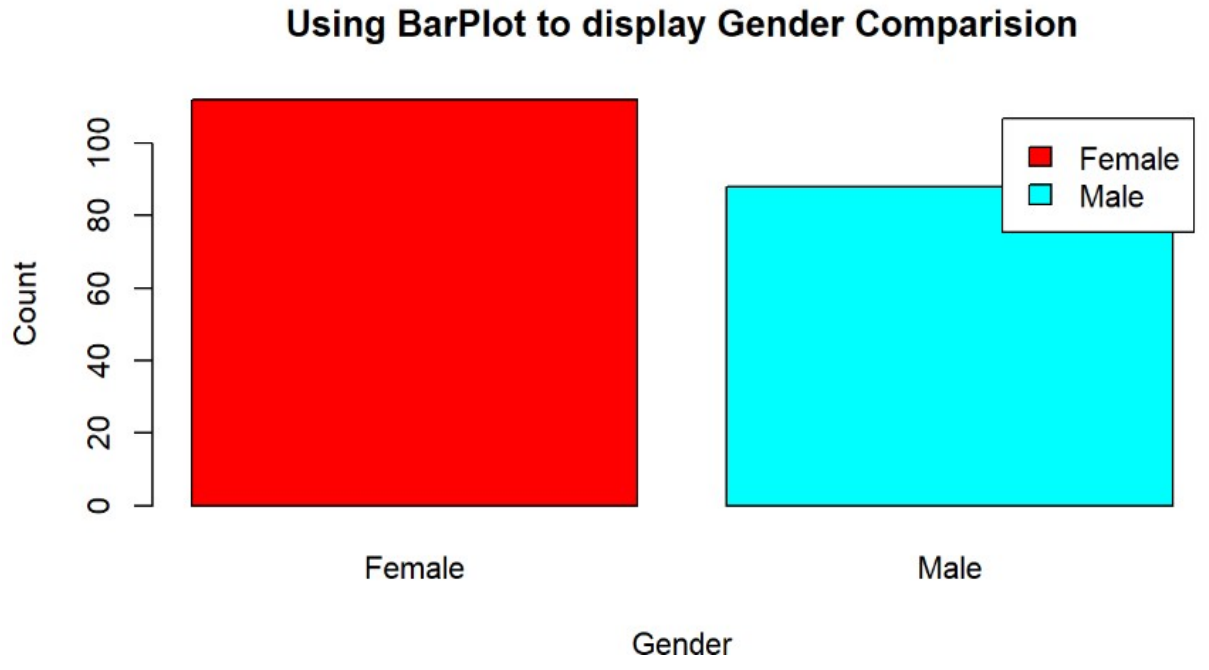
```
##    CustomerID        Gender               Age        Annual.Income..k..
##  Min.   :  1.00   Length:200         Min.   :18.00   Min.   : 15.00
##  1st Qu.: 50.75   Class :character   1st Qu.:28.75   1st Qu.: 41.50
##  Median :100.50   Mode  :character   Median :36.00   Median : 61.50
##  Mean   :100.50                      Mean   :38.85   Mean   : 60.56
##  3rd Qu.:150.25                      3rd Qu.:49.00   3rd Qu.: 78.00
##  Max.   :200.00                      Max.   :70.00   Max.   :137.00
##  Spending.Score..1.100. Number.of.items.bought  Invoice.ID
##  Min.   : 1.00          Min.   : 15.00         Length:200
##  1st Qu.:34.75          1st Qu.: 41.50         Class :character
##  Median :50.00          Median : 61.50         Mode  :character
##  Mean   :50.20          Mean   : 60.56
##  3rd Qu.:73.00          3rd Qu.: 78.00
##  Max.   :99.00          Max.   :137.00
##     Branch              City           Customer.type      Product.line
##  Length:200         Length:200         Length:200         Length:200
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    Unit.price       Quantity          Tax.5.            Total
##  Min.   :10.96   Min.   : 1.000   Min.   : 0.7715   Min.   :  16.2
##  1st Qu.:34.23   1st Qu.: 4.000   1st Qu.: 7.8464   1st Qu.: 164.8
##  Median :58.24   Median : 6.000   Median :15.3860   Median : 323.1
##  Mean   :58.44   Mean   : 6.055   Mean   :17.9922   Mean   : 377.8
##  3rd Qu.:82.14   3rd Qu.: 8.000   3rd Qu.:26.1340   3rd Qu.: 548.8
##  Max.   :99.96   Max.   :10.000   Max.   :49.4900   Max.   :1039.3
##      Date               Time             Payment               cogs
##  Length:200         Length:200         Length:200         Min.   : 15.43
##  Class :character   Class :character   Class :character   1st Qu.:156.93
##  Mode  :character   Mode  :character   Mode  :character   Median :307.72
##                                                           Mean   :359.84
##                                                           3rd Qu.:522.68
##                                                           Max.   :989.80
##  gross.margin.percentage  gross.income         Rating
##  Min.   :4.762           Min.   : 0.7715   Min.   : 4.000
##  1st Qu.:4.762           1st Qu.: 7.8464   1st Qu.: 5.675
##  Median :4.762           Median :15.3860   Median : 7.100
##  Mean   :4.762           Mean   :17.9922   Mean   : 7.017
##  3rd Qu.:4.762           3rd Qu.:26.1340   3rd Qu.: 8.400
##  Max.   :4.762           Max.   :49.4900   Max.   :10.000
```

Customer Analysis

Customer Gender Visualization

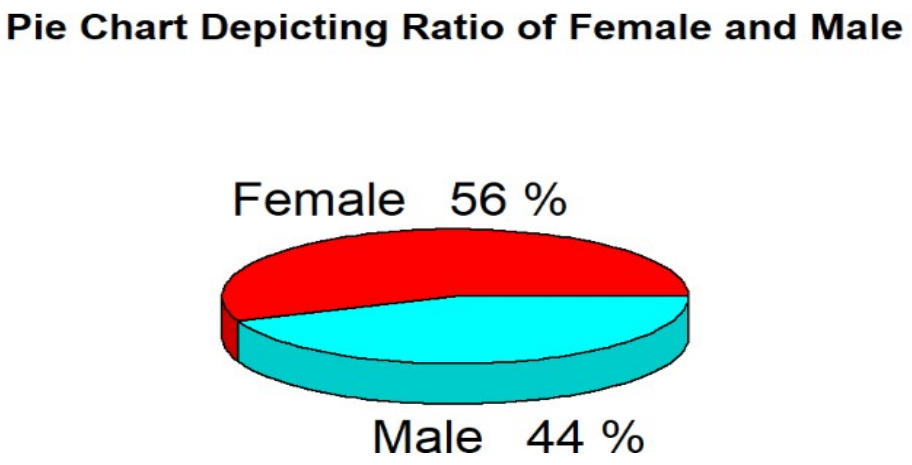
In this, we will create a barplot and a piechart to show the gender distribution across our customer_data dataset.

```
a=table(Mall_Customers$Gender)
barplot(a,main="Using BarPlot to display Gender Comparision",
        ylab="Count",
        xlab="Gender",
        col=rainbow(2),
        legend=rownames(a))
```



From the above barplot, we observe that the number of females is higher than the males. Now, let us visualize a pie chart to observe the ratio of male and female distribution. (Learnt from online “plotrix library”)

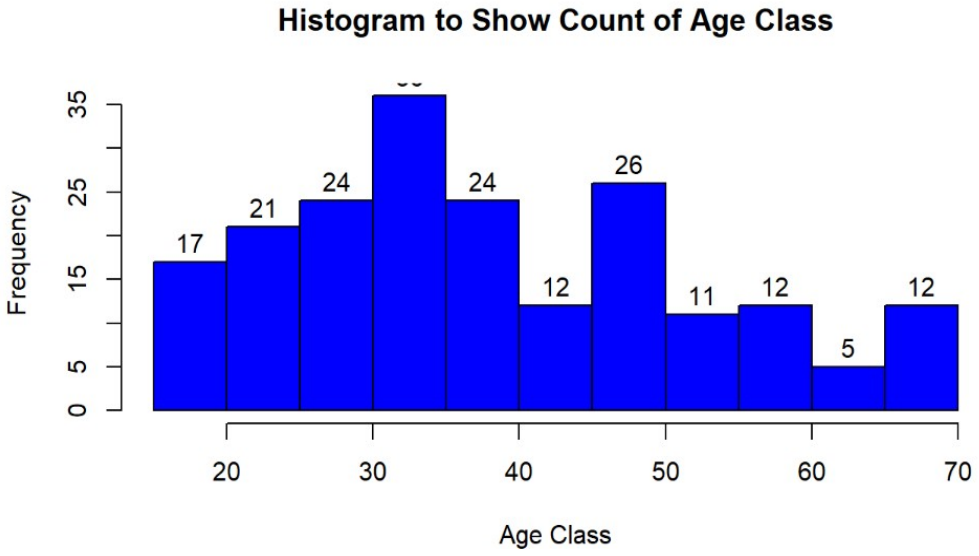
```
pct=round(a/sum(a)*100)
lbs=paste(c("Female","Male")," ",pct,"%",sep=" ")
library(plotrix)
pie3D(a,labels=lbs,
   main="Pie Chart Depicting Ratio of Female and Male")
```



Visualization of Age Distribution

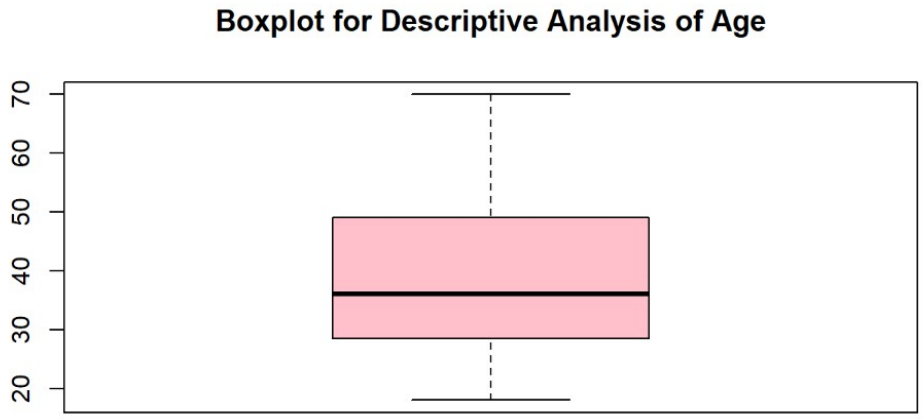
Let us plot a histogram to view the distribution to plot the frequency of customer ages. We will first proceed by taking summary of the Age variable.

```
hist(Mall_Customers$Age,
    col="blue",
    main="Histogram to Show Count of Age Class",
    xlab="Age Class",
    ylab="Frequency",
    labels=TRUE)
```



Barplot

```
boxplot(Mall_Customers$Age,
       col="pink",
       main="Boxplot for Descriptive Analysis of Age")
```



From the above two visualizations, we conclude that the maximum customer ages are between 30 and 35. The minimum age of customers is 18, whereas, the maximum age is 70.

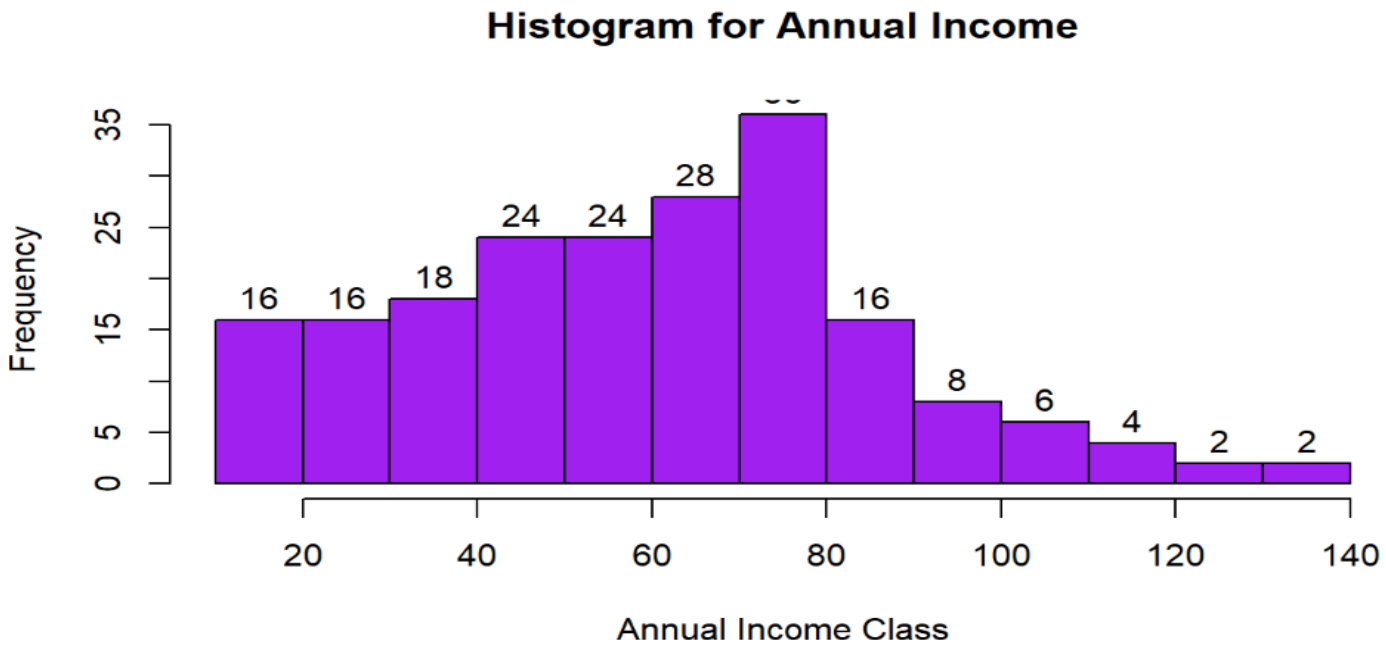
Analysis of the Annual Income of the Customers

In this section of the R project, we will create visualizations to analyze the annual income of the customers. We will plot a histogram and then we will proceed to examine this data using a density plot.

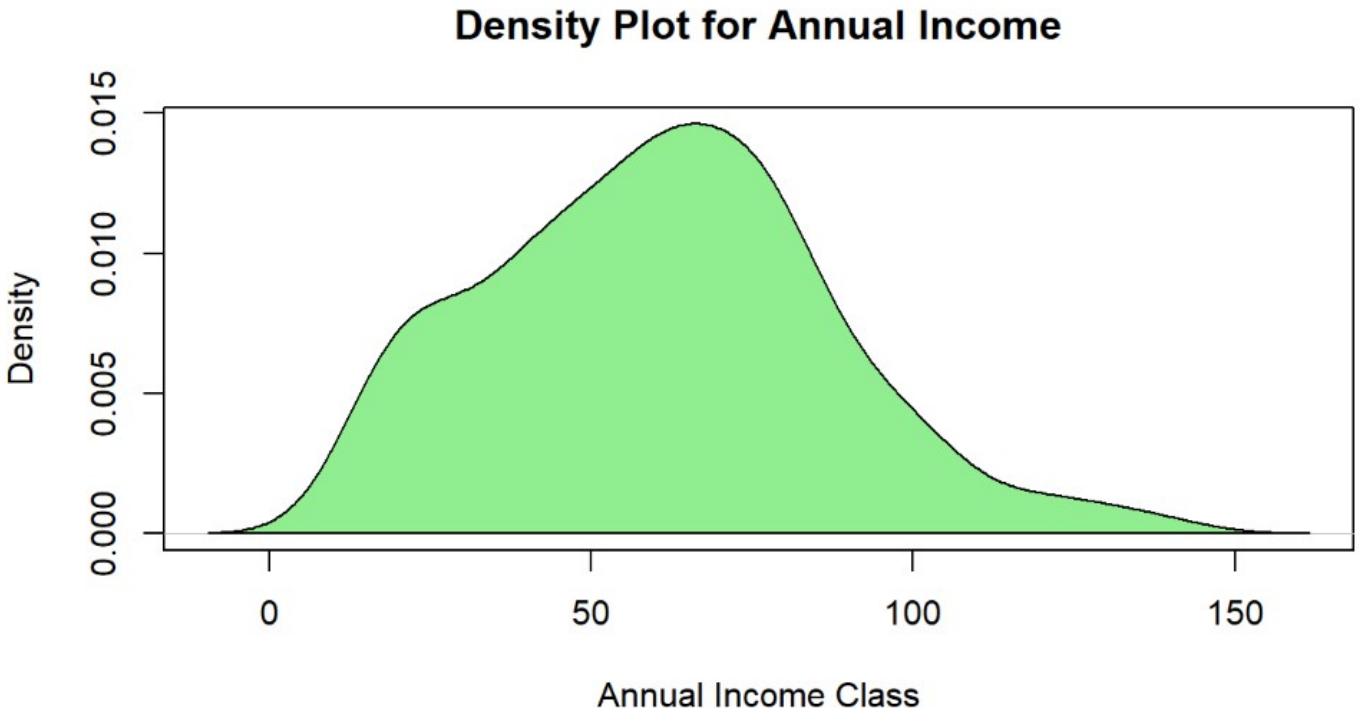
```
summary(Mall_Customers$Annual.Income..k..)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.00   41.50   61.50   60.56   78.00  137.00
```

```
hist(Mall_Customers$Annual.Income..k..,
  col="purple",
  main="Histogram for Annual Income",
  xlab="Annual Income Class",
  ylab="Frequency",
  labels=TRUE)
```



```
plot(density(Mall_Customers$Annual.Income..k..),
     col="yellow",
     main="Density Plot for Annual Income",
     xlab="Annual Income Class",
     ylab="Density")
polygon(density(Mall_Customers$Annual.Income..k..),
         col="lightgreen")
```



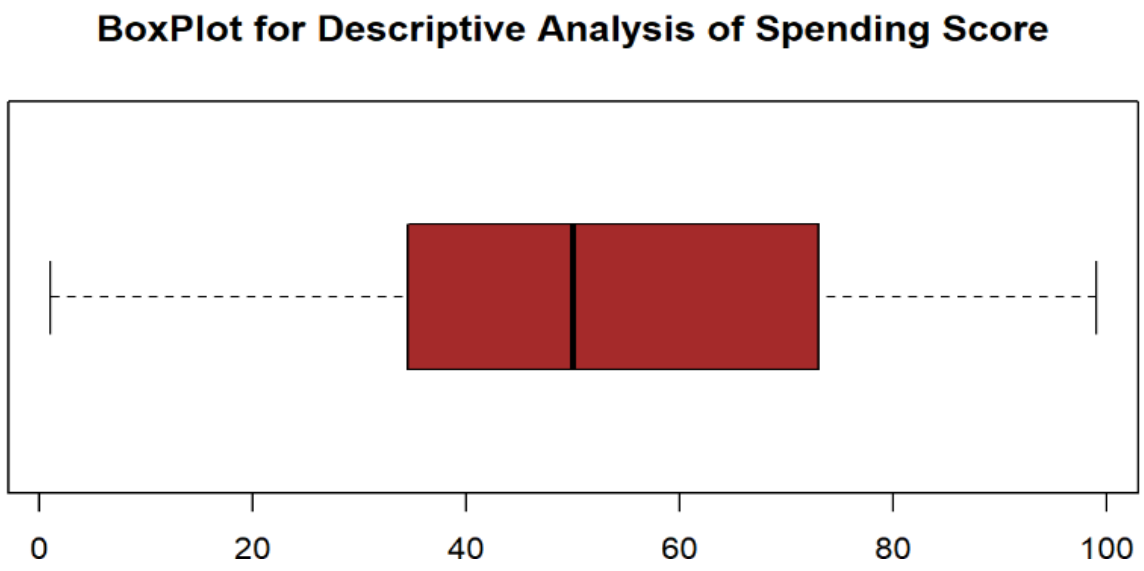
From the above descriptive analysis, we conclude that the minimum annual income of the customers is 15 and the maximum income is 137. People earning an average income of 70 have the highest frequency count in our histogram distribution. The average salary of all the customers is 60.56.

Analyzing Spending Score of the Customers

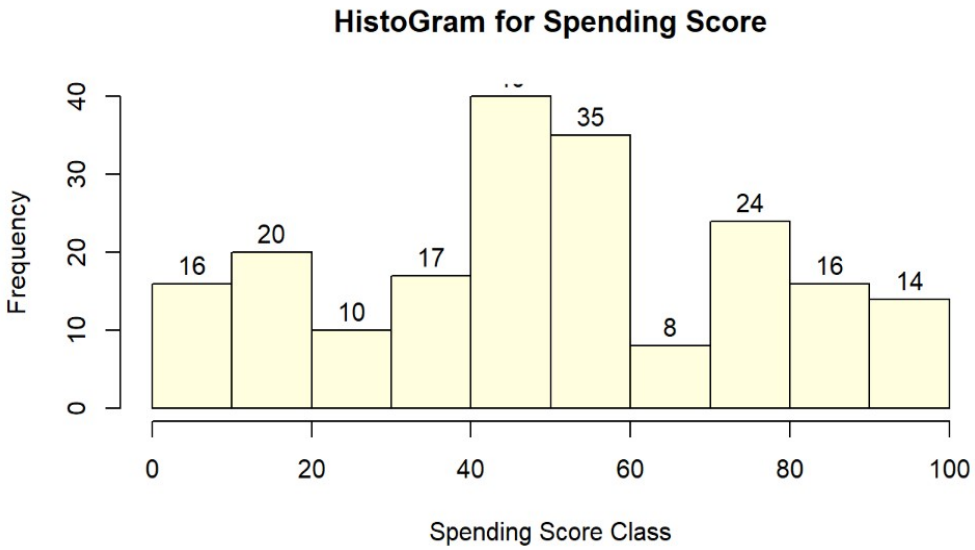
```
summary(Mall_Customers$Spending.Score..1.100.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   34.75   50.00   50.20   73.00   99.00
```

```
boxplot(Mall_Customers$Spending.Score..1.100.,
   horizontal=TRUE,
   col="brown",
   main="BoxPlot for Descriptive Analysis of Spending Score")
```



```
hist(Mall_Customers$Spending.Score..1.100.,
     main="HistoGram for Spending Score",
     xlab="Spending Score Class",
     ylab="Frequency",
     col="lightyellow",
     labels=TRUE)
```



The minimum spending score is 1, maximum is 99 and the average is 50.20. We can see Descriptive Analysis of Spending Score is that Min is 1, Max is 99 and avg. is 50.20. From the histogram, we conclude that customers between class 40 and 50 have the highest spending score among all the classes.

Product Analyzing

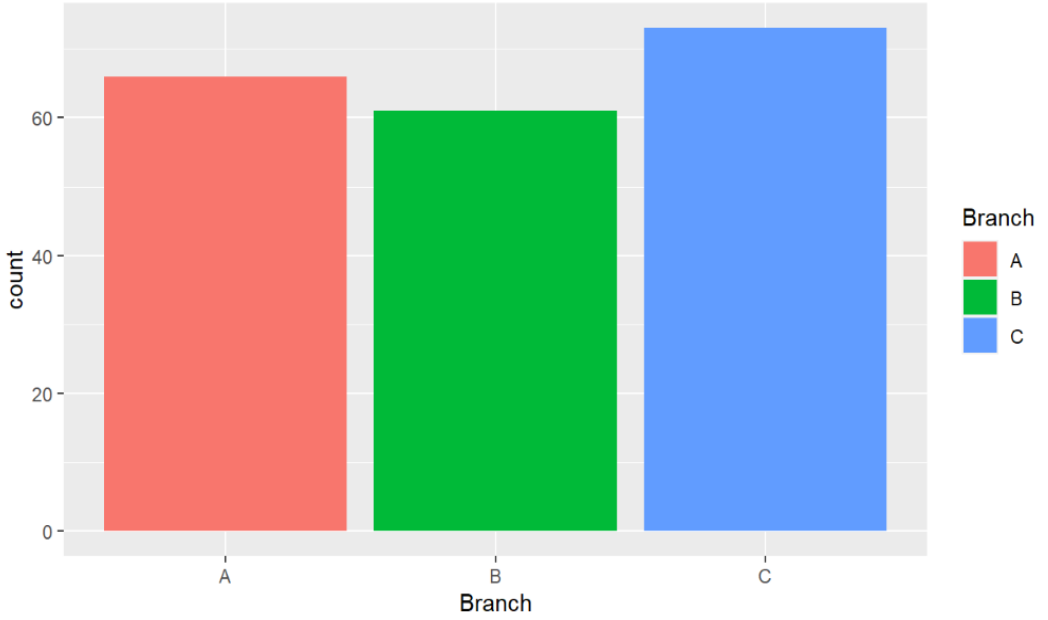
```
retail<-Mall_Customers
```

```
retail$Date <- gsub('/', '-', retail$Date)

r2 <- retail %>%
  mutate(DATE=mdy(Date))%>%
  select(everything())



retailA<- filter(r2, Branch=='A')
retailB<- filter(r2, Branch=='B')
retailC<- filter(r2, Branch=='C')

retailAsum <- group_by(retailA, DATE, Product.line) %>% select(everything())
retailAsum<- summarise(retailAsum, daytotal=sum(Total)) %>% select(everything())
```

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

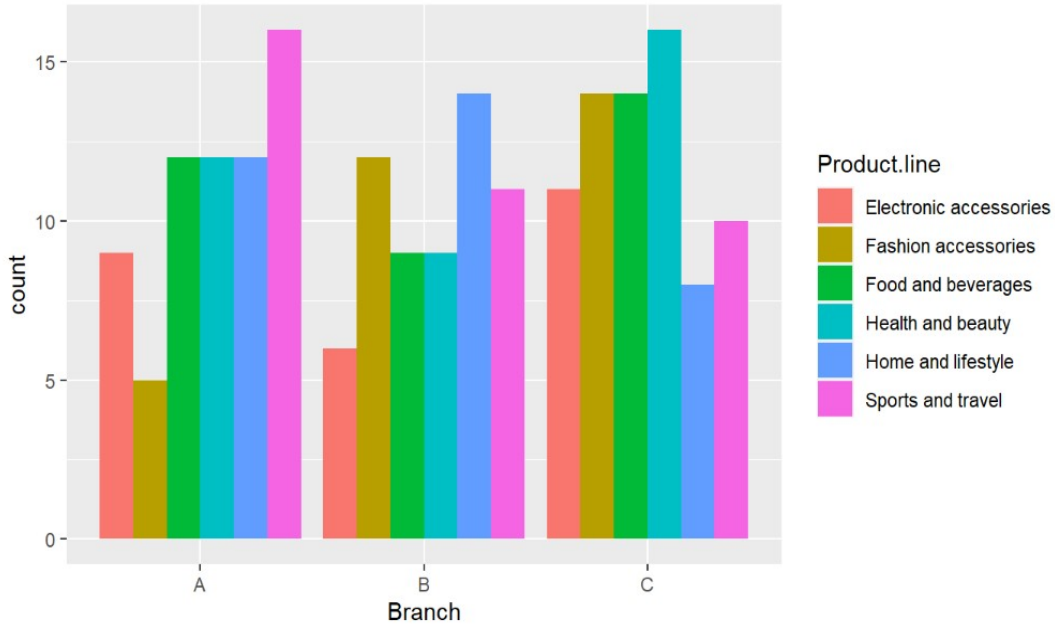
```
retailBsum <- group_by(retailB, DATE, Product.line) %>% select(everything())
retailBsum<- summarise(retailBsum, daytotal=sum(Total)) %>% select(everything())
```

```
library(ggplot2)
ggplot (data=retail)+
  geom_bar(mapping=aes(x=Branch,fill=Branch))
```



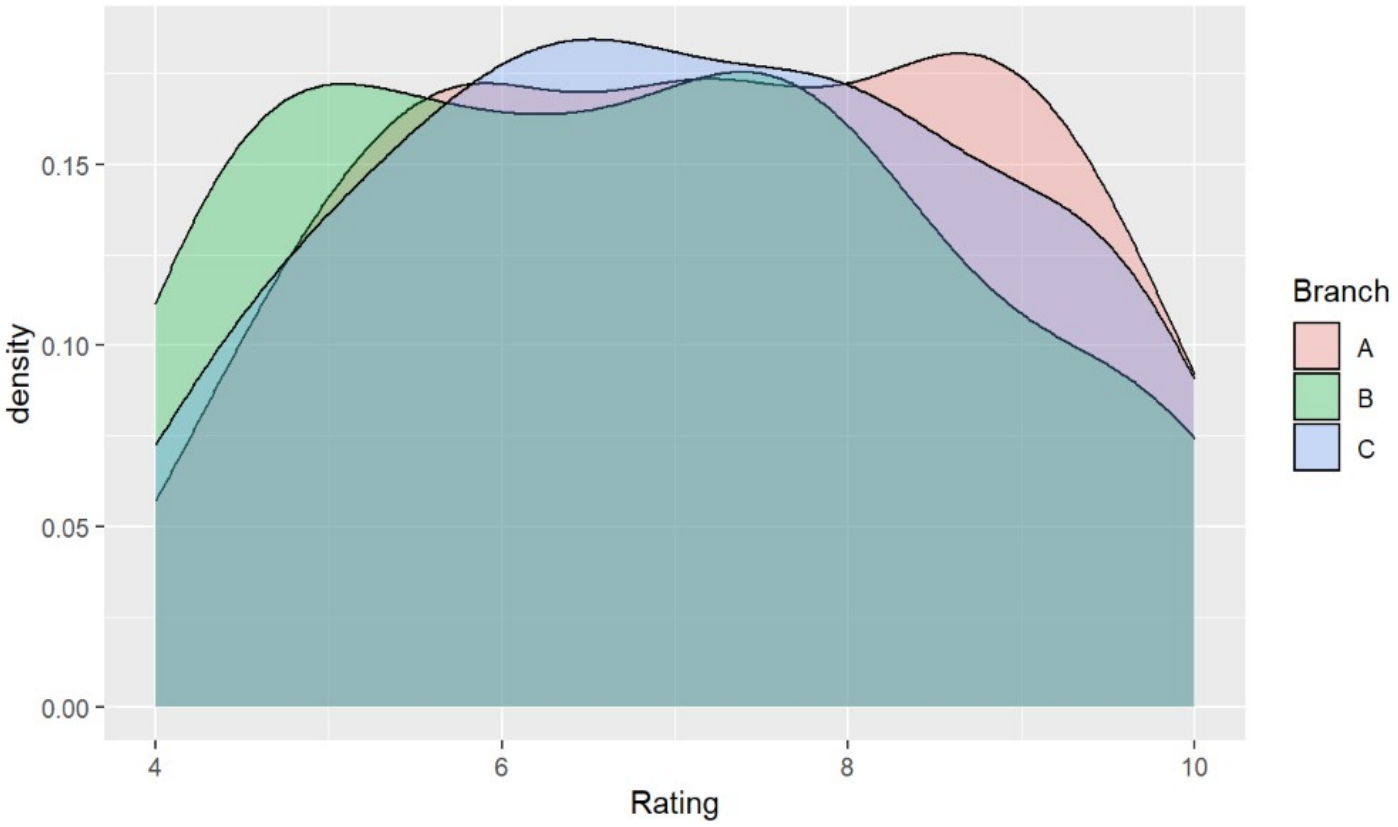
This graph shows the number of transactions per store over the period. It appears that store A sells slightly more than B, which sells more than C

```
ggplot (data=retail)+
  geom_bar(mapping=aes(x=Branch, fill=Product.line), position="dodge")
```

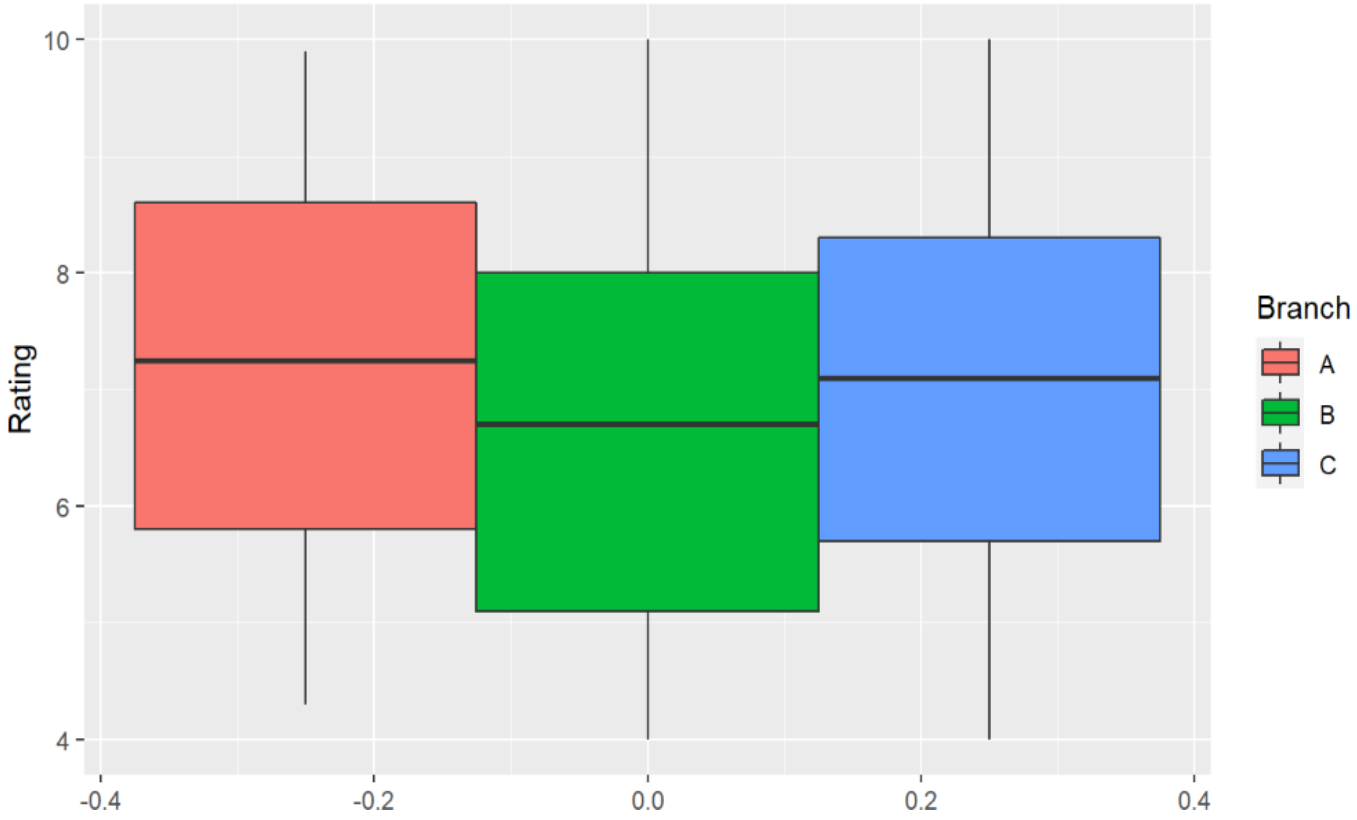


This graph shows quantity of sales per category. It appears that each store has a certain category that they sell more of than the other stores. This would benefit from futher analysis in order to determine the cause, e.g if the customer needs are different or if perhaps the products are not being displayed in an efficient manner.

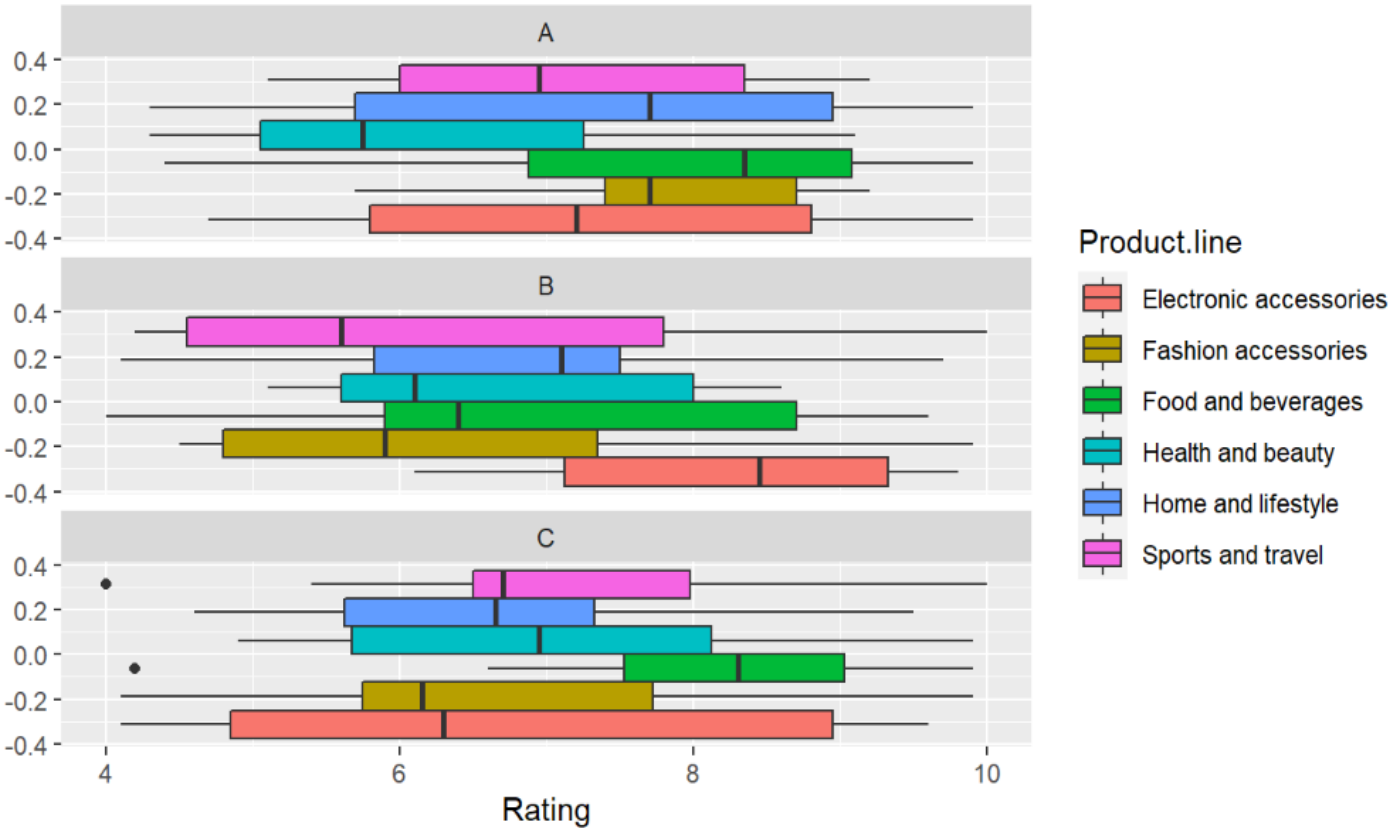
```
ggplot(data=retail) +
    geom_density(mapping=aes(x=Rating, fill=Branch), alpha=.3)
```



```
ggplot (data=retail)+
  geom_boxplot(mapping=aes(x=Rating, fill=Branch), position="dodge")+
    coord_flip()
```



```
ggplot (data=retail)+
  geom_boxplot(mapping=aes(x=Rating, fill=Product.line), position="dodge")+
    facet_wrap(~Branch, nrow=3)
```



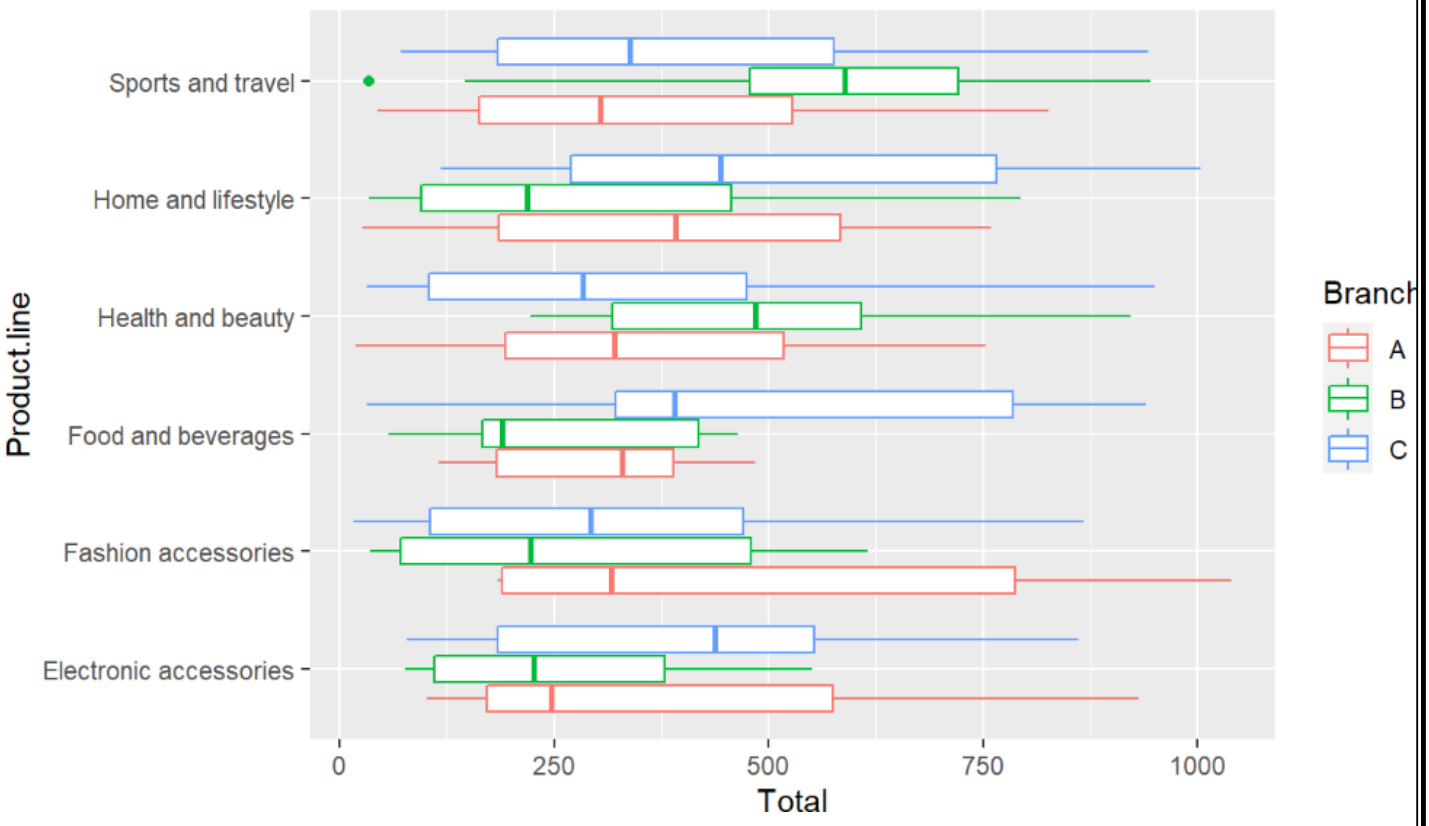
The first graph shows the distribution of the ratings per branch The second is a boxplot representing the statistical summary of the branch The third is a boxplot representing the ratings per product line by branch.

The first graph is perhaps the most imporant here. While the second graph shows branches A and C have similar ratings based on the average, we can see that branch C has a sort of double hump distribution. This represents a large amount of moderately dissatisfied customers (rating of about 6) and reasonably satisfied customers (rating about 8). Meanwhile, the bulk of branch A's ratings are centered around 7. On the other hand, branch B has obviously lower ratings than the other 2. This indicates that each branch may need a separate strategy in order to bring customer satisfaction up.

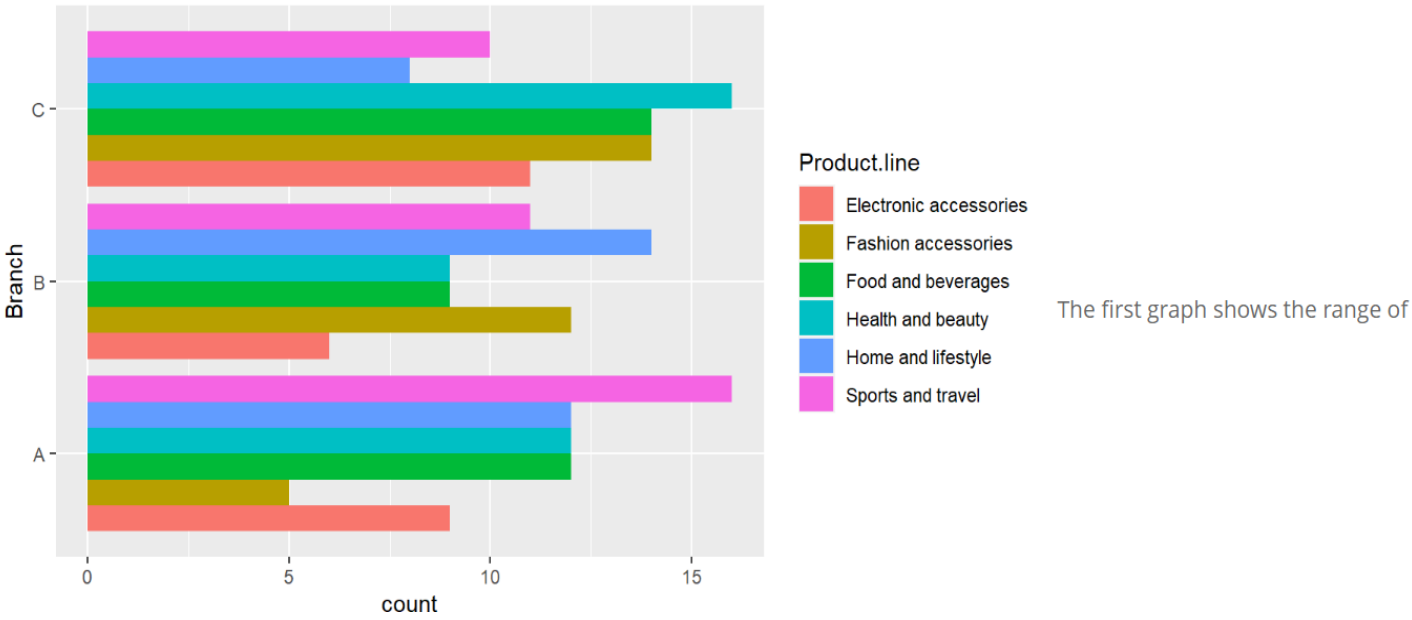
The third graph could be used to fine tune the strategy to increase customer satisfaction. We can see that the rating distrubution for product lines in each branch vary significantly. Perhaps the most interesting observation is in branch B. The ratings for nearly every category are either similar or less than the ratings of the other stores. However, it appears that the food and beverage category has a significantly better

average rating than the other categories as well as the other stores. This would be a key insight for further analysis in determining a course of action for customer satisfaction improvement.

```
ggplot(data=retail, mapping=aes(x=Product.line, y=Total, color=Branch))+
  geom_boxplot()+
  coord_flip()
```



```
ggplot (data=retail)+
  geom_bar(mapping=aes(x=Branch, fill=Product.line), position="dodge") +
    coord_flip()
```



The first graph shows the range of sales totals based on the category, the second graph is like the second graph above except flipped for easier comparison

A key observation from these graphs is the relatively low number of health and beauty purchases for branch A, as well as a relatively small range of purchase totals for health and beauty, as well as food and beverages.

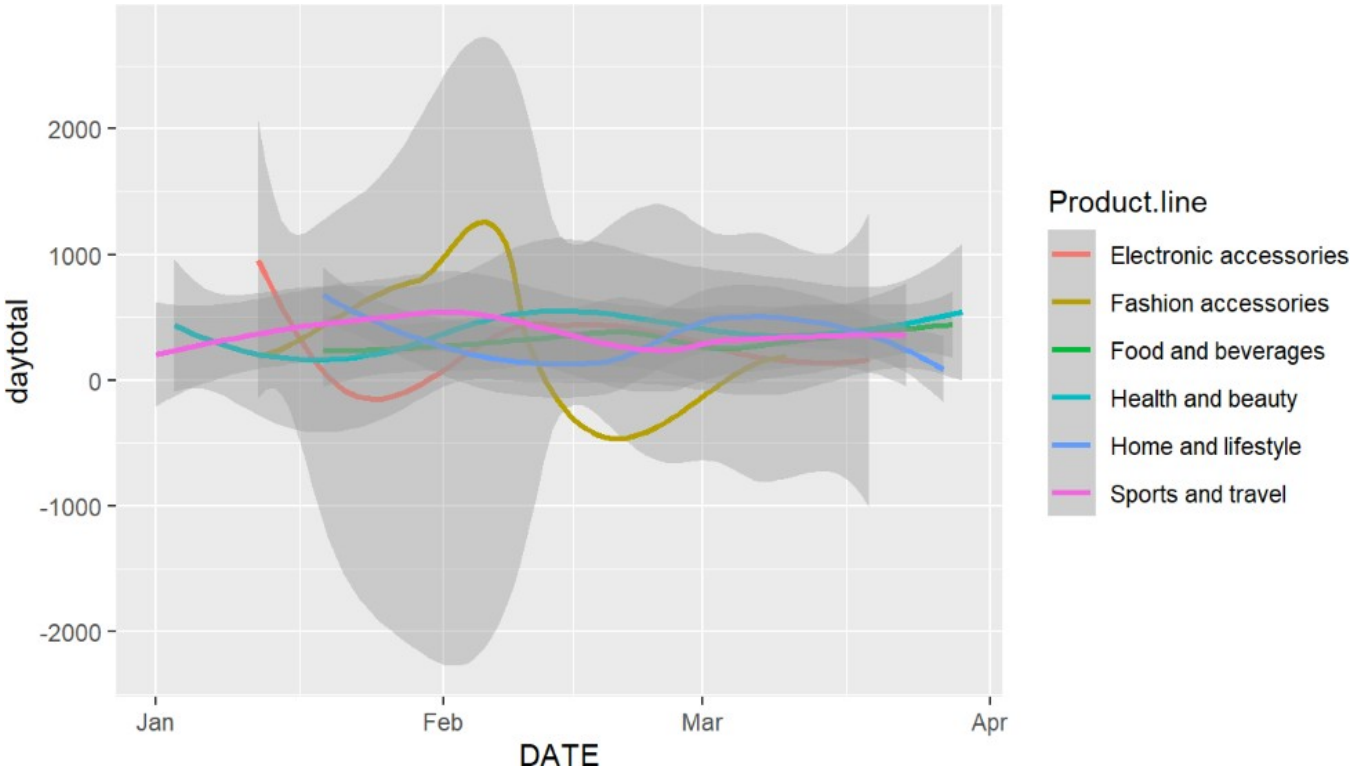
This could be explained by the demographic, which could be urban/city in for this location. These locations, which are more populous, also tend to have smaller household sizes. This could explain the larger number of total transactions for this branch, but the smaller range and average of individual purchase totals for household consumables.

More analysis is needed but perhaps this store would benefit by removing slow moving products, such as family oriented value size packs, and replacing them with products from a better performing category

Another key observation from these graphs is a relatively low number of sales for Sports and Travel as well as Home and Lifestyle for branch C. While the sports and lifestyle total range is similar to the other stores, the home and lifestyle range is smaller. A possible explanation for this is that this location is near a freeway, and attracts more “transient” customers who do not want to purchase large items such as these as they are far from home. This store would likely benefit from a reduction of inventory size from these categories.

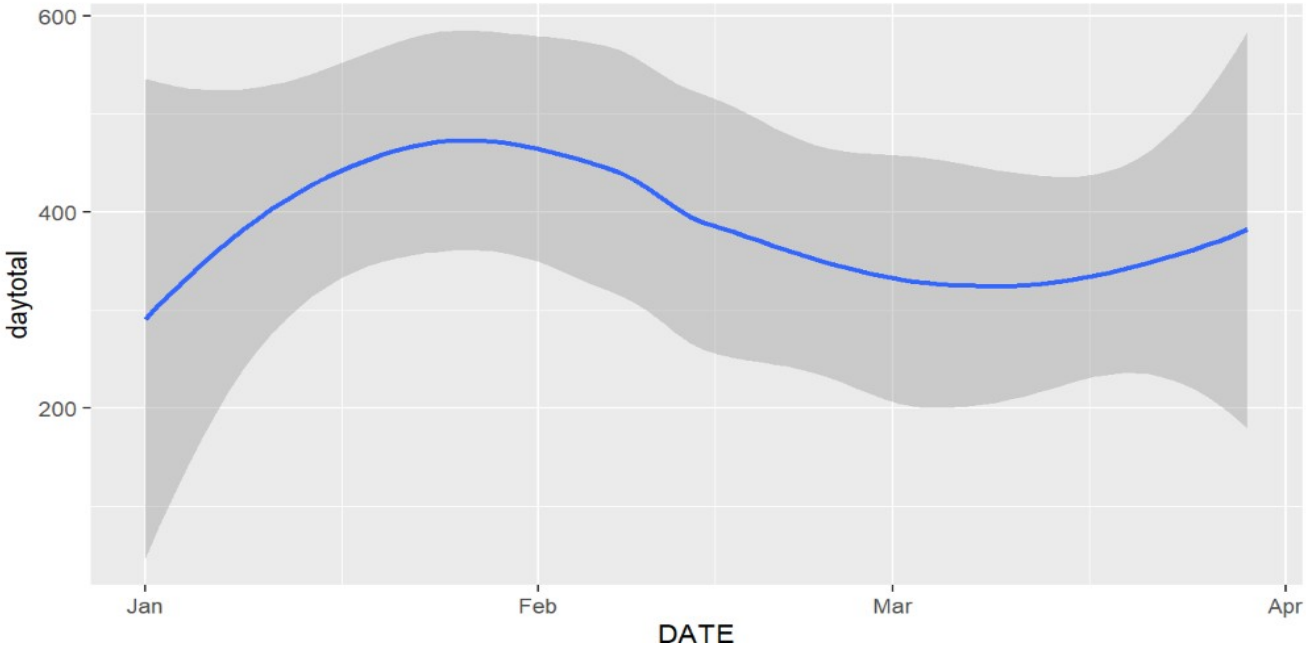
The next analyses we will do will be based on each store over time. However, the data is not in a useable format yet. We need to divide the dataset up by store as well as transform the date data into a universal format. We will also aggregate the data based on total sales per day per product line

```
ggplot(data=retailAsum)+
    geom_smooth(mapping=aes(x=DATE, y=daytotal ,color=Product.line))
```

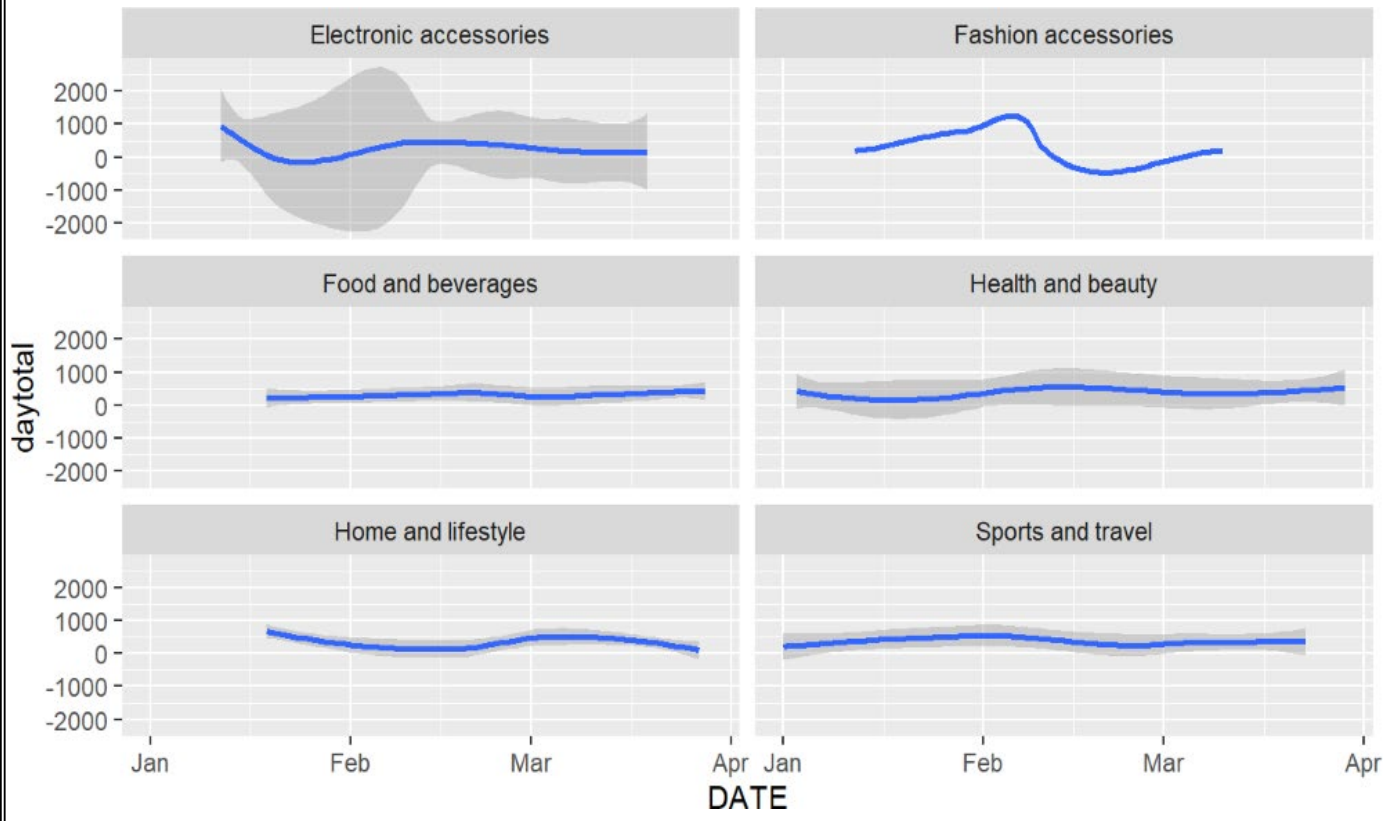


```
ggplot(data=retailAsum)+
    geom_smooth(mapping=aes(x=DATE, y=daytotal))
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
ggplot(data=retailAsum) +
    geom_smooth(mapping=aes(x=DATE, y=daytotal)) +
    facet_wrap(~Product.line, nrow=3)
```



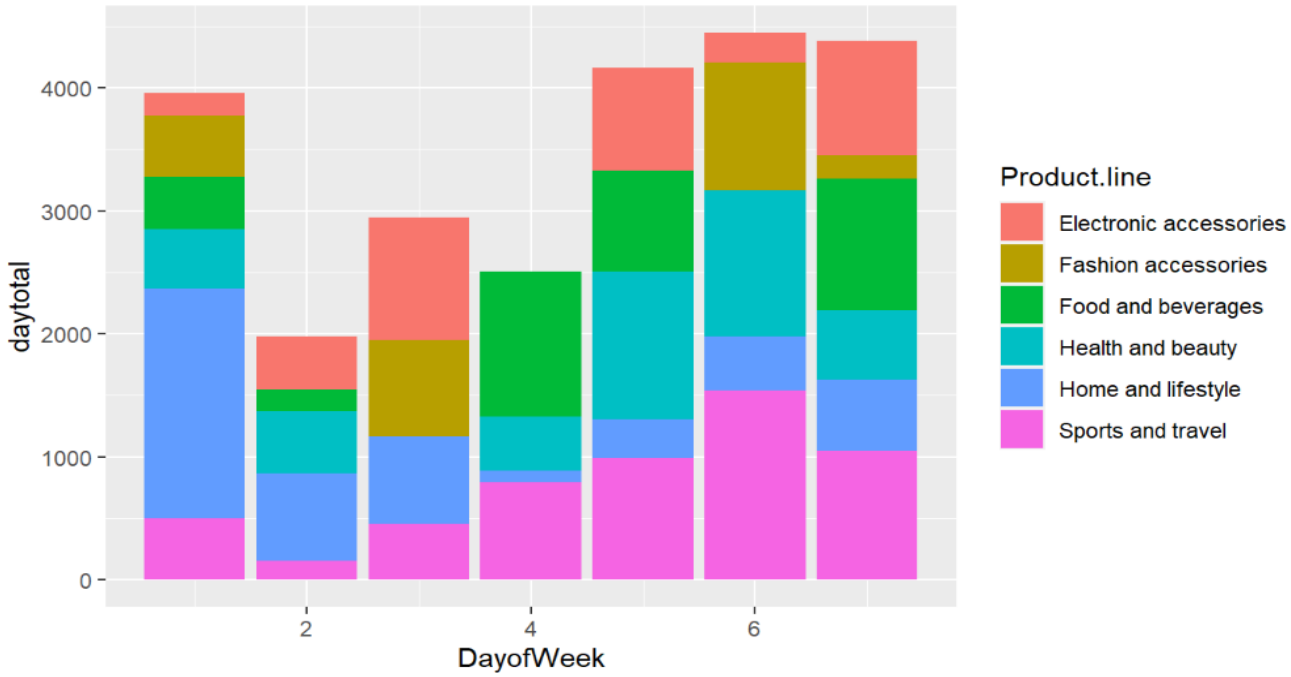
While this graph is a little difficult to see, it shows the monetary value of the total sales per day per product line over the time period. This type of graph could be extremely useful in seasonal displays and advertising. We can clearly see that sports and travel sales spike towards the spring, while electronic accessories falls severly after january. These are fairly obvious seasonal changes, but the other trends shown by this graph indicate they are not the only ones. Food and beverage has a strange wavy pattern with a strong climb in late march. Health and beauty/home and lifestyle follow similar trends that converge by the end of March.

The second graph shows the trendline of the total per day for location A. It is similar to the above graph but without the categories. From it we see there is a decline in sales during Feburary, with an uptick in March.

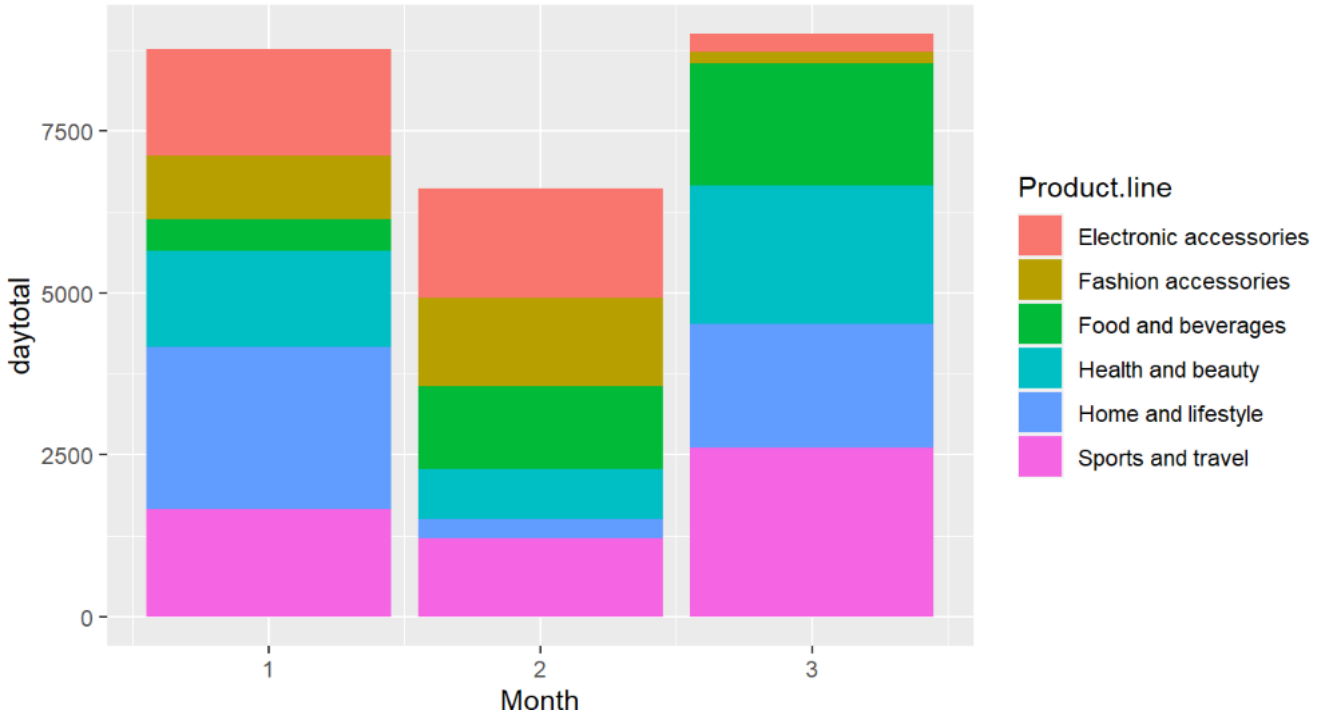
The observations from these charts would be a great start towards creating a machine learning model to analyze sales trends vs month of year in order to create bundles, optimize high traffic displays, and in advertising.

The third chart included is just the first chart broken up. It is easier to see individual trends but more difficult to see comparative trends. Both graphs serve a purpose here.

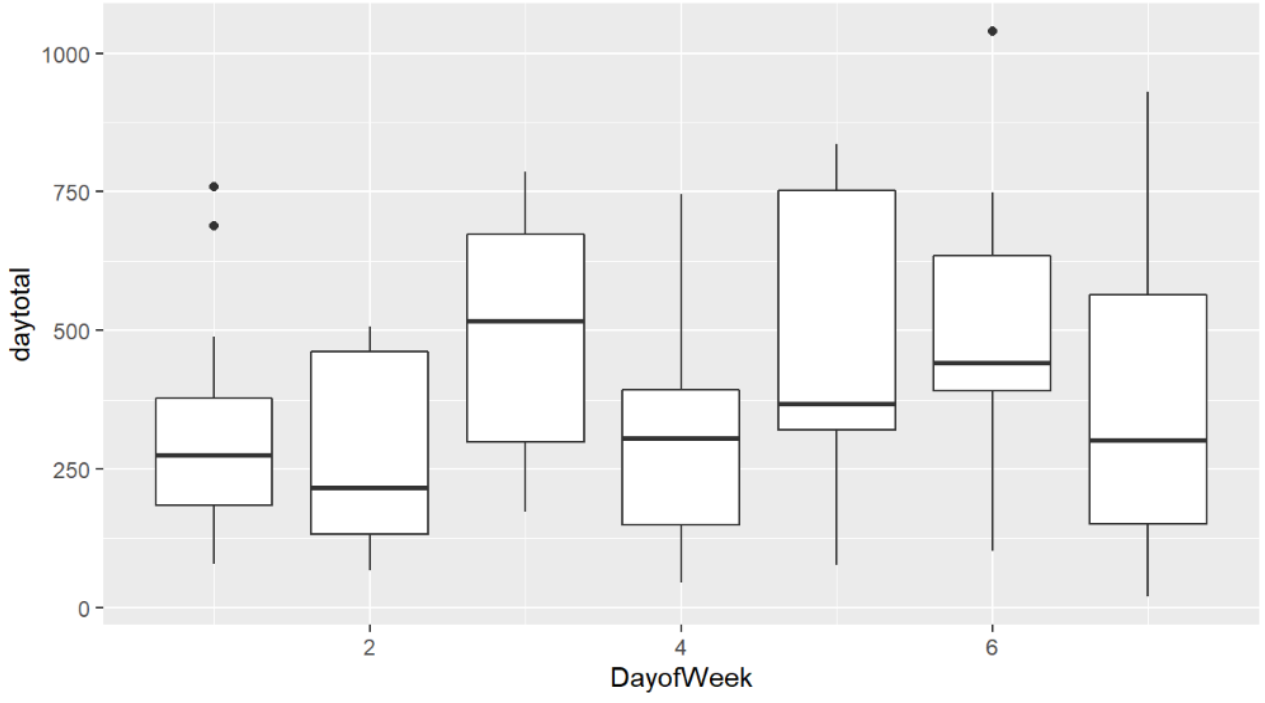
```
retailAsum <- retailAsum %>%
  mutate(
    Month=month(DATE)
  )

retailAsum <- retailAsum %>%
  mutate(
    DayofWeek=wday(DATE)
  )

retailAsum <- retailAsum %>%
  mutate(
    DayofYear=yday(DATE)
  )


ggplot(data=retailAsum)+
    geom_bar(mapping=aes(x=DayofWeek, y=daytotal, fill=Product.line),stat='identity')
```



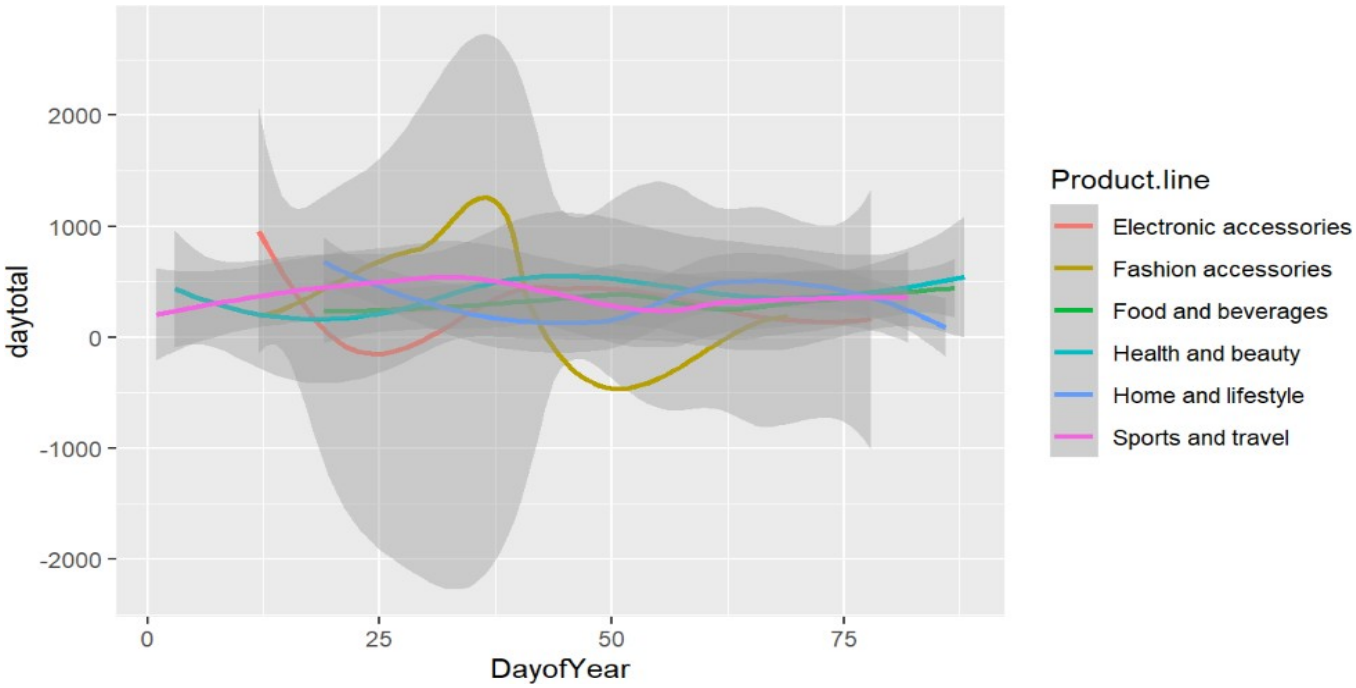
```
ggplot(data=retailAsum)+
    geom_bar(mapping=aes(x=Month, y=daytotal, fill=Product.line),stat='identity')
```




```
ggplot(data=retailAsum)+
    geom_boxplot(mapping=aes(x=DayofWeek, y=daytotal, group=DayofWeek))
```



```
ggplot(data=retailAsum)+
    geom_smooth(mapping=aes(x=DayofYear, y=daytotal, color=Product.line))
```



K-means

While using the k-means clustering algorithm, the first step is to indicate the number of clusters (k) that we wish to produce in the final output. The algorithm starts by selecting k objects from dataset randomly that will serve as the initial centers for our clusters. These selected objects are the cluster means, also known as centroids. Then, the remaining objects have an assignment of the closest centroid.

This centroid is defined by the Euclidean Distance present between the object and the cluster mean. We refer to this step as “cluster assignment”. When the assignment is complete, the algorithm proceeds to calculate new mean value of each cluster present in the data. After the recalculation of the centers, the observations are checked if they are closer to a different cluster. Using the updated cluster mean, the objects undergo reassignment. This goes on repeatedly through several iterations until the cluster assignments stop altering. The clusters that are present in the current iteration are the same as the ones obtained in the previous iteration.

If you want to work one of the major challenges then knowledge Big Data is crucial. Therefore, I recommend to check out Hadoop for Data Science.

Summing up the K-means clustering –

We specify the number of clusters that we need to create. The algorithm selects k objects at random from the dataset. This object is the initial cluster or mean. The closest centroid obtains the assignment of a new observation. We base this assignment on the Euclidean Distance between object and the centroid. k clusters in the data points update the centroid through calculation of the new mean values present in all the data points of the cluster. The kth cluster's centroid has a length of p that contains means of all variables for observations in the k-th cluster.

We denote the number of variables with p. Iterative minimization of the total within the sum of squares. Then through the iterative minimization of the total sum of the square, the assignment stop wavering when we achieve maximum iteration. The default value is 10 that the R software uses for the maximum iterations. Determining Optimal Clusters While working with clusters, you need to specify the number of clusters to use. You would like to utilize the optimal number of clusters. To help you in determining the optimal clusters, there are three popular methods –

Elbow method Silhouette method Gap statistic

Elbow Method

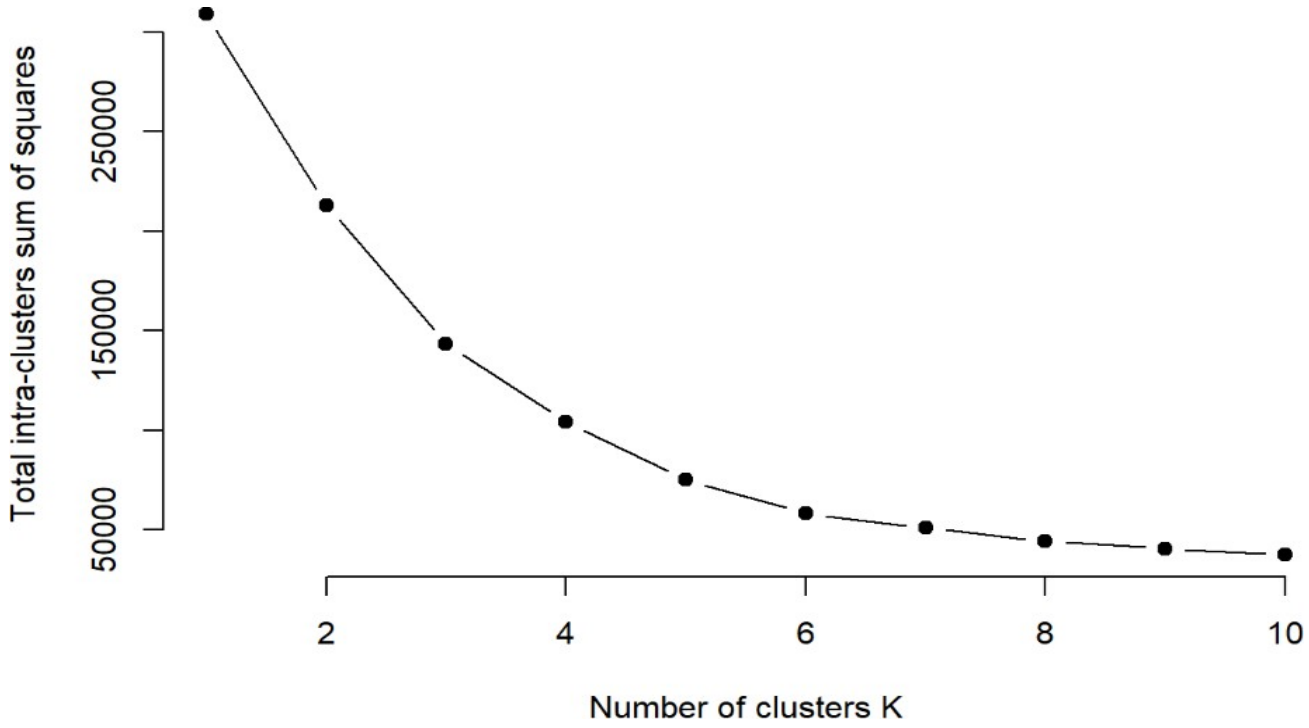
The main goal behind cluster partitioning methods like k-means is to define the clusters such that the intra-cluster variation stays minimum.

minimize(sum W(Ck)), k=1…k

Where Ck represents the kth cluster and W(Ck) denotes the intra-cluster variation. With the measurement of the total intra-cluster variation, one can evaluate the compactness of the clustering boundary. We can then proceed to define the optimal clusters as follows –

First, we calculate the clustering algorithm for several values of k. This can be done by creating a variation within k from 1 to 10 clusters. We then calculate the total intra-cluster sum of square (iss). Then, we proceed to plot iss based on the number of k clusters. This plot denotes the appropriate number of clusters required in our model. In the plot, the location of a bend or a knee is the indication of the optimum number of clusters.

```r
library(purrr)
set.seed(123)
# function to calculate total intra-cluster sum of square
iss <- function(k) {
  kmeans(Mall_Customers[,3:5],k,iter.max=100,nstart=100,algorithm="Lloyd" )$tot.withinss
}
k.values <- 1:10
iss_values <- map_dbl(k.values, iss)
Plot_No_of_clusters_K_VS_total_intra_clusters <- function(){
plot(k.values, iss_values,
    type="b", pch = 19, frame = FALSE,
    xlab="Number of clusters K",
    ylab="Total intra-clusters sum of squares")
}
Plot_No_of_clusters_K_VS_total_intra_clusters()
```



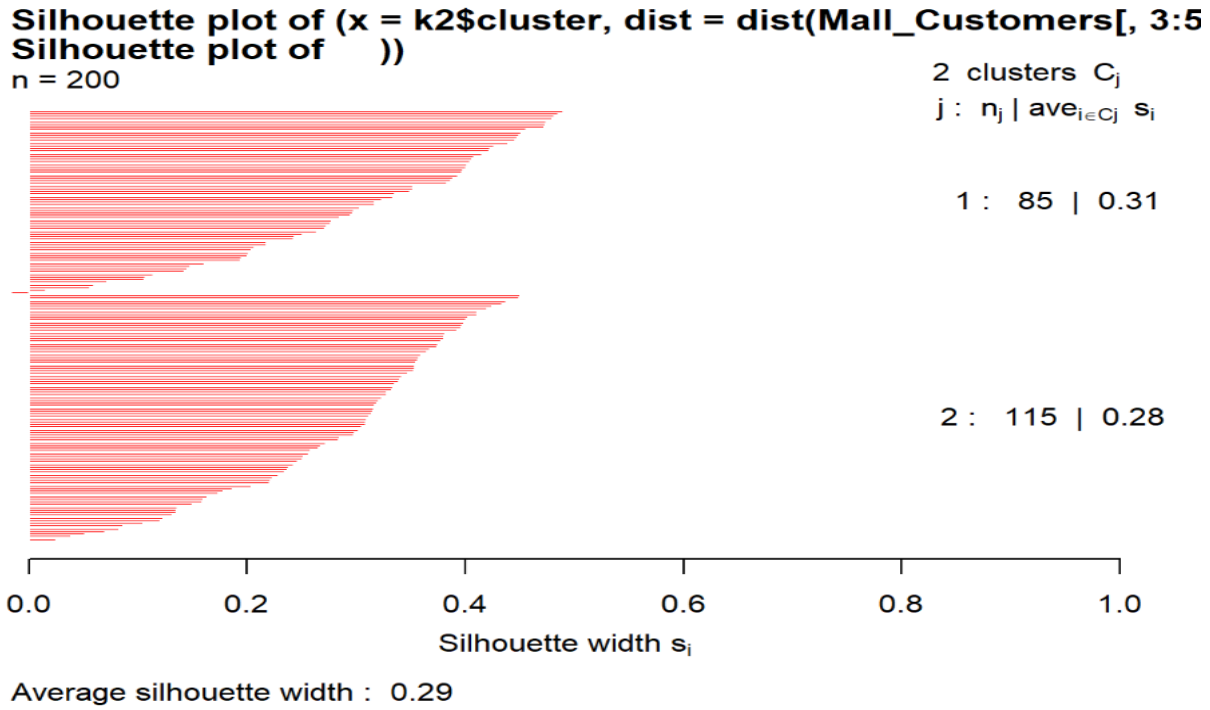
Average Silhouette Method

With the help of the average silhouette method, we can measure the quality of our clustering operation. With this, we can determine how well within the cluster is the data object. If we obtain a high average silhouette width, it means that we have good clustering. The average silhouette method calculates the mean of silhouette observations for different k values. With the optimal number of k clusters, one can maximize the average silhouette over significant values for k clusters.

Using the silhouette function in the cluster package, we can compute the average silhouette width using the kmean function. Here, the optimal cluster will possess highest average.

```
library(cluster)
library(gridExtra)
```

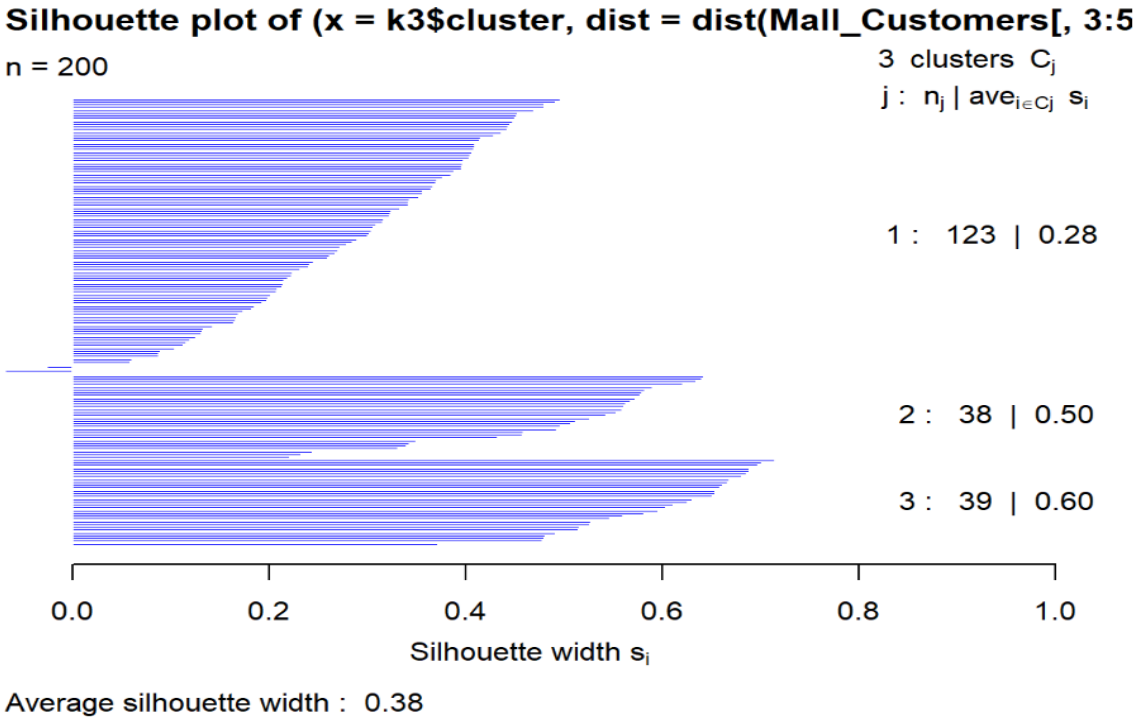
```
library(grid)
library(dplyr)
silhouette_s2 <- function(){
k2<-kmeans(Mall_Customers[,3:5],2,iter.max=100,nstart=50,algorithm="Lloyd")
s2<-plot(silhouette(k2$cluster,dist(Mall_Customers[,3:5],"euclidean",)),col = "red")
}
silhouette_s2()
```




```
## K-means clustering with 2 clusters of sizes 85, 115
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 28.95294           62.17647                73.62353
## 2 46.16522           59.36522                32.88696
##
## Clustering vector:
##   [1] 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2
##  [38] 1 2 1 2 1 2 1 2 1 2 2 2 2 2 1 1 2 2 2 2 2 1 2 2 1 2 2 2 1 2 2 1 2 2 2 2 2
##  [75] 2 1 2 2 1 2 2 1 2 2 1 2 2 1 1 2 2 2 2 2 2 1 2 1 2 1 2 2 2 1 2 2 2 2 2 2 2
## [112] 1 2 1 1 1 2 1 2 2 1 2 1 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1
## [149] 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2
## [186] 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1
##
## Within cluster sum of squares by cluster:
## [1]  88300.12 124540.05
##  (between_SS / total_SS =  31.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Performance Metrics for (K =2) = 31.1%

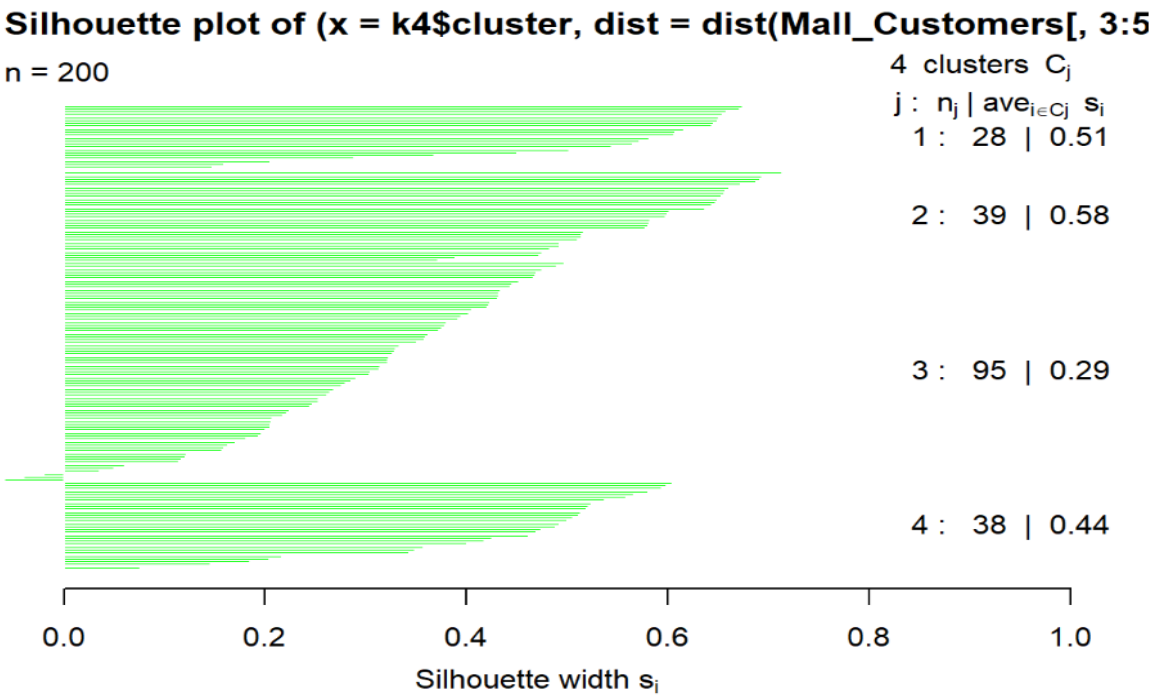
```
silhouette_s3 <- function(){
k3<-kmeans(Mall_Customers[,3:5],3,iter.max=100,nstart=50,algorithm="Lloyd")
s3<-plot(silhouette(k3$cluster,dist(Mall_Customers[,3:5],"euclidean")),col = "blue")
}
silhouette_s3()
```



```
## K-means clustering with 3 clusters of sizes 38, 123, 39
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 40.39474           87.00000                18.63158
## 2 40.32520           44.15447                49.82927
## 3 32.69231           86.53846                82.12821
##
## Clustering vector:
##   [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
## [149] 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1
## [186] 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
##
## Within cluster sum of squares by cluster:
## [1]  18993.92 110376.47  13972.36
##  (between_SS / total_SS =  53.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Performance Metrics for (k=3) = 53.6%

```
silhouette_s4 <- function(){
k4<-kmeans(Mall_Customers[,3:5],4,iter.max=100,nstart=50,algorithm="Lloyd")
s4<-plot(silhouette(k4$cluster,dist(Mall_Customers[,3:5],"euclidean")),col="green")
}
silhouette_s4()
```



```
## K-means clustering with 4 clusters of sizes 39, 38, 28, 95
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 32.69231           86.53846                82.12821
## 2 40.39474           87.00000                18.63158
## 3 24.82143           28.71429                74.25000
## 4 44.89474           48.70526                42.63158
##
## Clustering vector:
##   [1] 3 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4
##  [38] 3 4 3 4 3 4 3 4 3 4 4 4 4 4 3 4 4 4 4 4 4 4 4 3 4 4 4 3 4 4 3 4 4 4 4 4 4
##  [75] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
## [112] 4 4 4 4 4 4 4 4 4 4 4 4 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1
## [149] 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2
## [186] 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1
##
## Within cluster sum of squares by cluster:
## [1] 13972.359 18993.921  9099.071 62300.800
##  (between_SS / total_SS =  66.2 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Performance Metrics for (k=4) = 66.2%

```
silhouette_s5 <- function(){
k5<-kmeans(Mall_Customers[,3:5],5,iter.max=100,nstart=50,algorithm="Lloyd")
s5<-plot(silhouette(k5$cluster,dist(Mall_Customers[,3:5],"euclidean")),col="orange")
}
silhouette_s5()
```

```
## K-means clustering with 5 clusters of sizes 80, 22, 23, 39, 36
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 42.93750           55.08750               49.71250
## 2 25.27273           25.72727               79.36364
## 3 45.21739           26.30435               20.91304
## 4 32.69231           86.53846               82.12821
## 5 40.66667           87.75000               17.58333
##
## Clustering vector:
##   [1] 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3
##  [38] 2 3 2 3 2 3 1 3 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [112] 1 1 1 1 1 1 1 1 1 1 1 4 5 4 1 4 5 4 5 4 5 4 5 4 5 4 5 4 1 4 5 4 5 4
## [149] 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5
## [186] 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4
##
## Within cluster sum of squares by cluster:
## [1] 30673.462  4099.818  8948.609 13972.359 17669.500
##  (between_SS / total_SS =  75.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
s5<-plot(silhouette(k5$cluster,dist(Mall_Customers[,3:5],"euclidean")),col="orange")
```

Performance Metrics for(k=5) = 75.6%

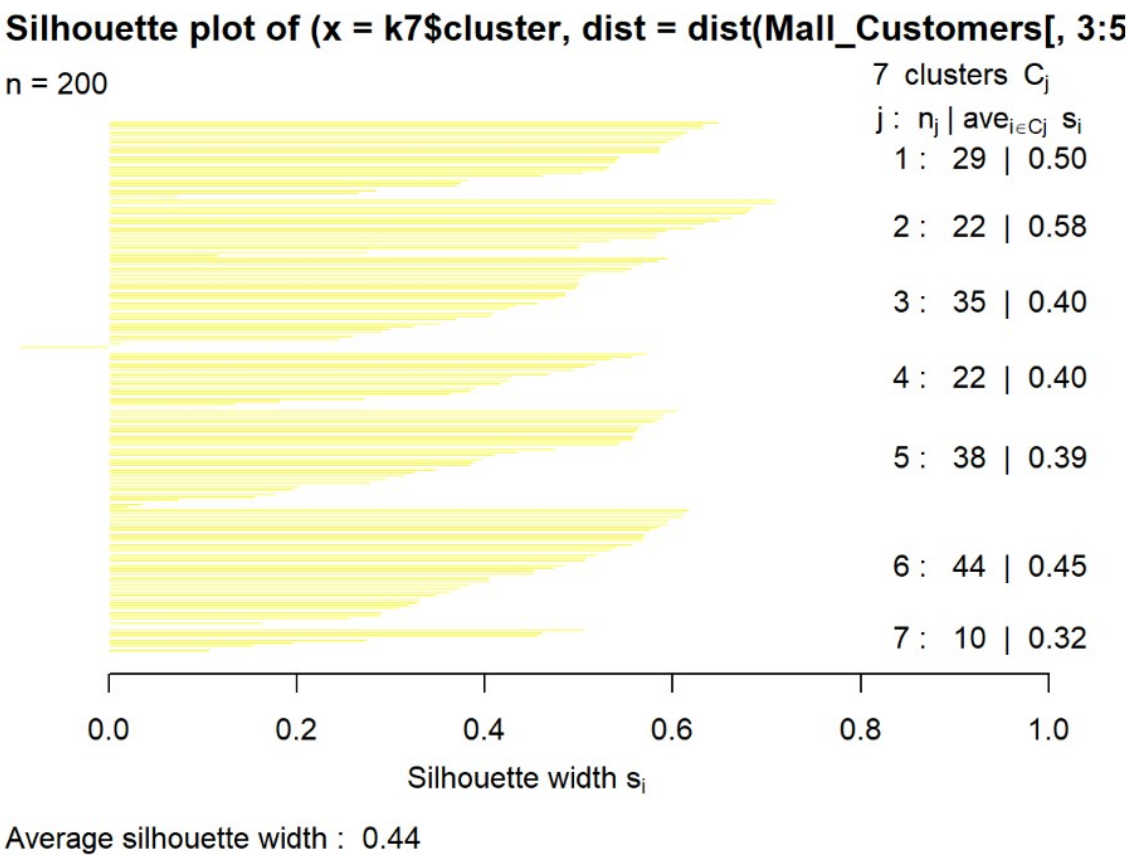
```
silhouette_s6 <- function(){
k6<-kmeans(Mall_Customers[,3:5],6,iter.max=100,nstart=50,algorithm="Lloyd")
s6<-plot(silhouette(k6$cluster,dist(Mall_Customers[,3:5],"euclidean")),col = "purple")
}
silhouette_s6()
```

```
k6<-kmeans(Mall_Customers[,3:5],6,iter.max=100,nstart=50,algorithm="Lloyd")
k6
```

```
## K-means clustering with 6 clusters of sizes 35, 38, 45, 21, 39, 22
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 41.68571           88.22857                17.28571
## 2 27.00000           56.65789                49.13158
## 3 56.15556           53.37778                49.08889
## 4 44.14286           25.14286                19.52381
## 5 32.69231           86.53846                82.12821
## 6 25.27273           25.72727                79.36364
##
## Clustering vector:
##   [1] 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4 6 4
##  [38] 6 4 6 3 6 3 2 4 6 3 2 2 2 3 2 2 3 3 3 3 3 2 3 3 2 3 3 3 2 3 3 2 2 3 3 3 3
##  [75] 3 2 3 2 2 3 3 2 3 3 2 3 3 2 2 3 3 2 3 2 2 2 3 2 3 2 2 3 3 2 3 2 3 3 3 3 3
## [112] 2 2 2 2 2 3 3 3 3 2 2 2 5 2 5 1 5 1 5 1 5 2 5 1 5 1 5 1 5 1 5 2 5 1 5 1 5
## [149] 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5 1
## [186] 5 1 5 1 5 1 5 1 5 1 5 1 5 1 5
##
## Within cluster sum of squares by cluster:
## [1] 16690.857  7742.895  8062.133  7732.381 13972.359  4099.818
##  (between_SS / total_SS =  81.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Performance Metrics for (k=6) = 81.1%

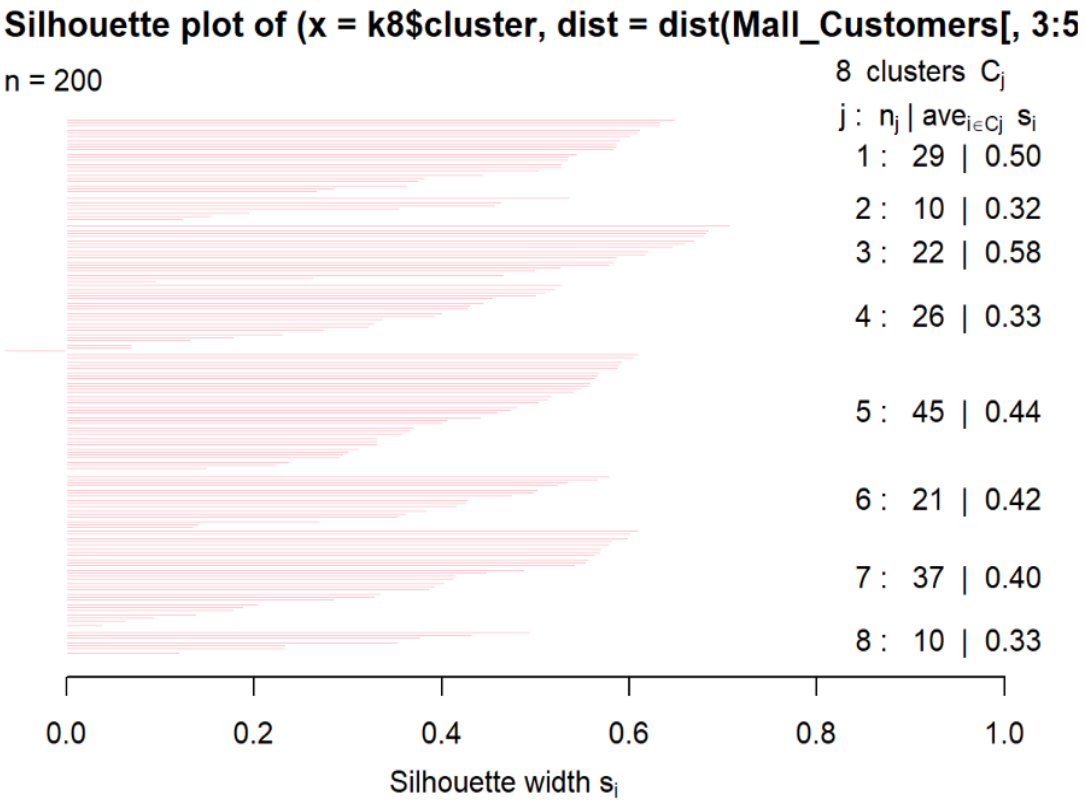
```
silhouette_s7 <- function(){
k7<-kmeans(Mall_Customers[,3:5],7,iter.max=100,nstart=50,algorithm="Lloyd")
s7<-plot(silhouette(k7$cluster,dist(Mall_Customers[,3:5],"euclidean")),col = "yellow")
}
silhouette_s7()
```




```
K-means clustering with 7 clusters of sizes 45, 22, 35, 28, 21, 38, 11

Cluster means:
       Age Annual.Income..k.. Spending.Score..1.100.
1 56.15556           53.37778                49.08889
2 25.27273           25.72727                79.36364
3 41.68571           88.22857                17.28571
4 32.78571           78.03571                81.89286
5 44.14286           25.14286                19.52381
6 27.00000           56.65789                49.13158
7 32.45455          108.18182                82.72727

Clustering vector:
  [1] 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 1 2 1 6 5 2
 [47] 1 6 6 6 1 6 6 1 1 1 1 1 6 1 1 6 1 1 1 6 1 1 6 6 1 1 1 1 1 6 1 6 6 1 1 6 1 1 6 1 1 6 6 1 1 6
 [93] 1 6 6 6 1 6 1 6 6 1 1 6 1 6 1 1 1 1 1 6 6 6 6 6 1 1 1 1 6 6 6 4 6 4 3 4 3 4 3 4 6 4 3 4 3 4
[139] 3 4 3 4 6 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 7 3 7 3 7
[185] 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7

Within cluster sum of squares by cluster:
[1]  8062.133  4099.818 16690.857  3864.357  7732.381  7742.895  2924.545
 (between_SS / total_SS =  83.4 %)
```

Performance Metrics for (k=7) = 83.4%

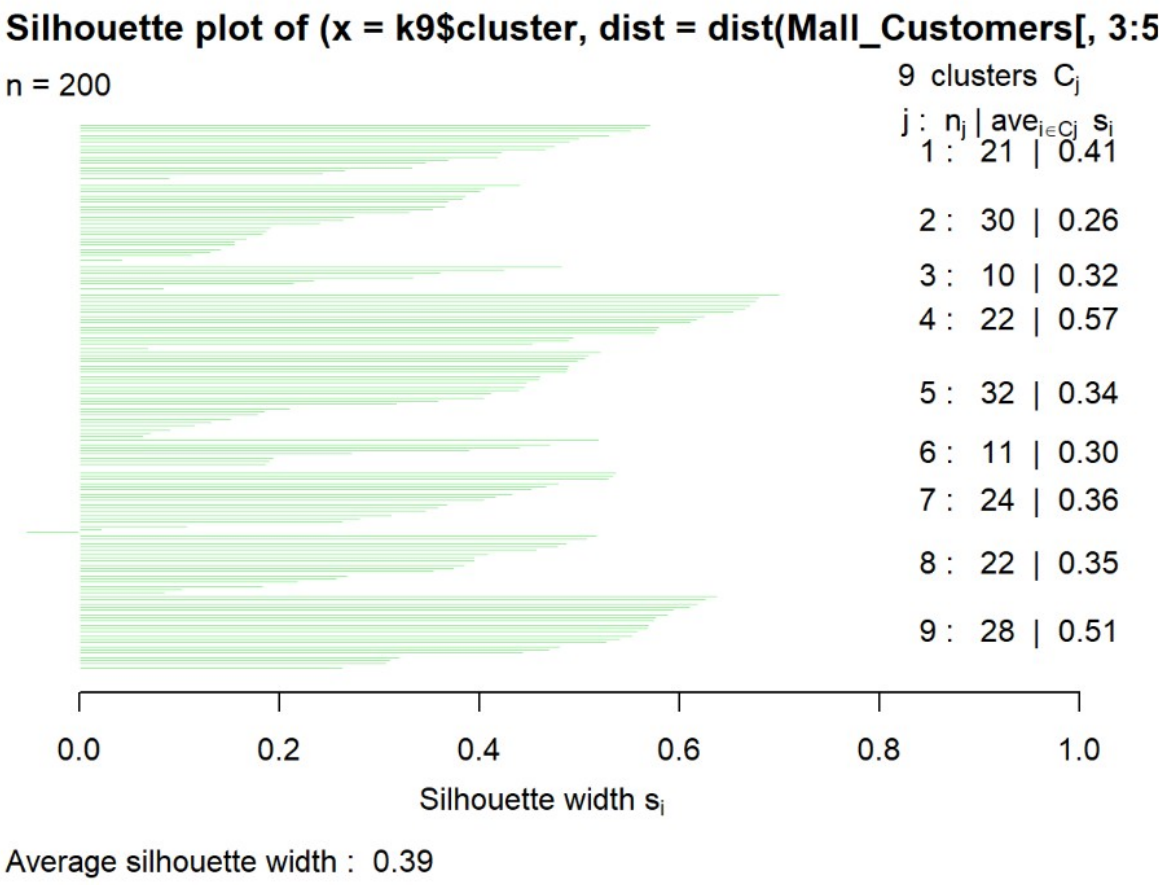
```
silhouette_s8 <- function(){
k8<-kmeans(Mall_Customers[,3:5],8,iter.max=100,nstart=50,algorithm="Lloyd")
s8<-plot(silhouette(k8$cluster,dist(Mall_Customers[,3:5],"euclidean")),col="pink")
}
silhouette_s8()
```



```
K-means clustering with 8 clusters of sizes 25, 29, 22, 10, 10, 38, 21, 45

Cluster means:
       Age Annual.Income..k.. Spending.Score..1.100.
1 41.96000           79.64000                15.40000
2 32.86207           78.55172                82.17241
3 25.27273           25.72727                79.36364
4 41.00000          109.70000                22.00000
5 32.20000          109.70000                82.00000
6 27.00000           56.65789                49.13158
7 44.14286           25.14286                19.52381
8 56.15556           53.37778                49.08889

Clustering vector:
  [1] 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 7 3 8 3 8 6 7 3
 [47] 8 6 6 6 8 6 6 8 8 8 8 8 6 8 8 6 8 8 8 6 8 8 6 6 8 8 8 8 6 8 6 6 8 8 6 8 8 6 8 8 6
 [93] 8 6 6 6 8 6 8 6 6 8 8 6 8 6 8 8 8 8 8 6 6 6 6 6 8 8 8 8 6 6 6 2 6 2 1 2 1 2 1 2 6 2 1 2 1 2
[139] 1 2 1 2 6 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 4 5 4 5
[185] 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5

Within cluster sum of squares by cluster:
[1] 7004.720 4148.759 4099.818 2914.100 2605.700 7742.895 7732.381 8062.133
 (between_SS / total_SS =  85.7 %)
```

Performance Metrics for (k=8) = 85.7%

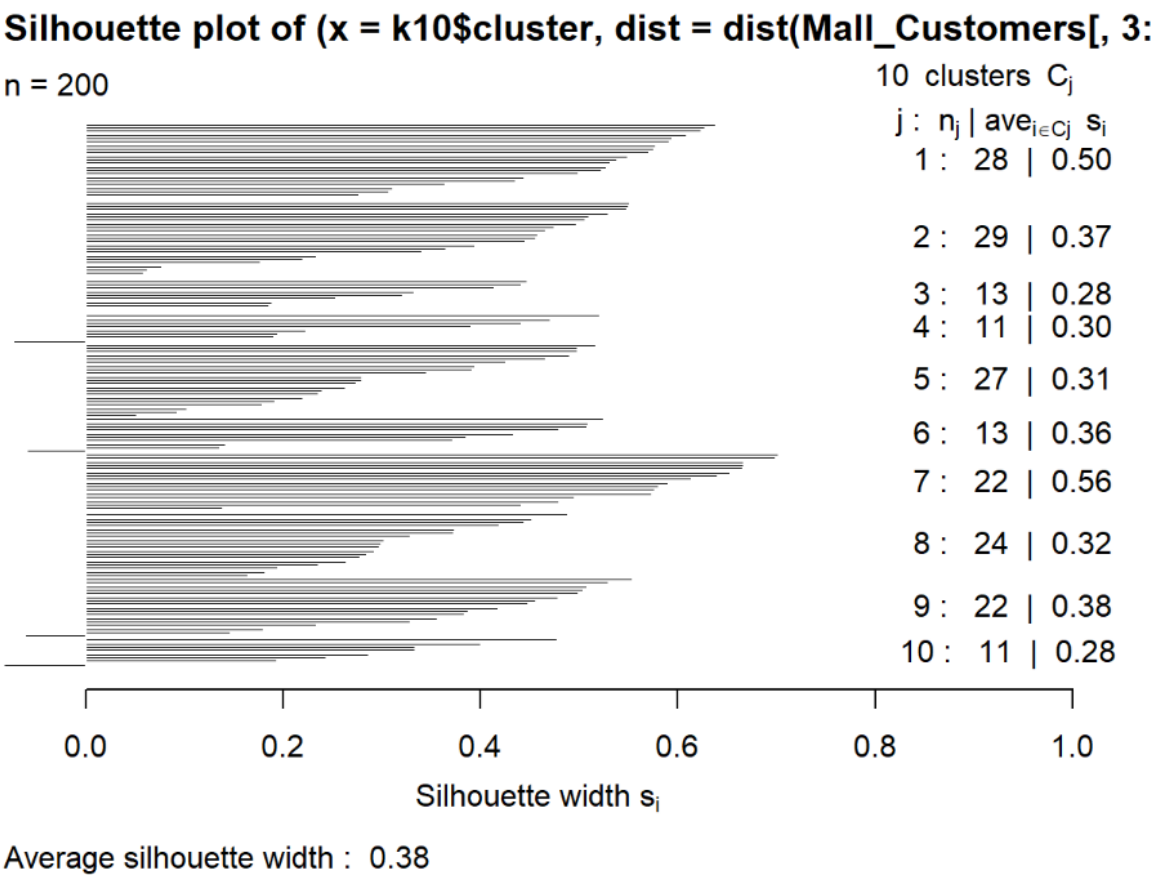
```
silhouette_s9 <- function(){
k9<-kmeans(Mall_Customers[,3:5],9,iter.max=100,nstart=50,algorithm="Lloyd")
s9<-plot(silhouette(k9$cluster,dist(Mall_Customers[,3:5],"euclidean")),col="light green")
}
silhouette_s9()
```



```
K-means clustering with 9 clusters of sizes 10, 11, 28, 12, 36, 22, 44, 12, 25

Cluster means:
       Age Annual.Income..k.. Spending.Score..1.100.
1 41.00000          109.70000              22.000000
2 32.45455          108.18182              82.727273
3 32.78571           78.03571              81.892857
4 48.75000           24.58333               9.583333
5 26.83333           57.58333              49.527778
6 25.27273           25.72727              79.363636
7 56.34091           53.70455              49.386364
8 37.50000           29.33333              34.583333
9 41.96000           79.64000              15.400000

Clustering vector:
  [1] 8 6 4 6 8 6 4 6 4 6 4 6 4 6 4 6 8 6 8 6 8 6 4 6 4 6 8 6 8 6 4 6 4 6 4 6 4 6 8 6 7 6 8 5 8 6
 [47] 7 5 8 8 7 5 5 7 7 7 7 7 5 7 7 5 7 7 7 5 7 7 5 5 7 7 7 7 7 5 7 5 5 7 7 5 7 7 5 7 7 5 5 7 7 5
 [93] 7 5 5 5 7 5 7 5 5 7 7 5 7 5 7 7 7 7 7 5 5 5 5 5 7 7 7 7 5 5 5 3 5 3 9 3 9 3 9 3 5 3 9 3 9 3
[139] 9 3 9 3 5 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 3 9 2 1 2 1 2
[185] 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2

Within cluster sum of squares by cluster:
[1] 2914.100 2924.545 3864.357 2830.083 7028.722 4099.818 7607.477 2328.583 7004.720
 (between_SS / total_SS =  86.9 %)
```

Performance Metrics for(k=9) = 86.9%

```
silhouette_s10 <- function(){
k10<-kmeans(Mall_Customers[,3:5],10,iter.max=100,nstart=50,algorithm="Lloyd")
s10<-plot(silhouette(k10$cluster,dist(Mall_Customers[,3:5],"euclidean")),col = "black")
}
silhouette_s10()
```



```
## K-means clustering with 10 clusters of sizes 22, 13, 12, 10, 29, 27, 29, 11, 25, 22
##
## Cluster means:
##         Age Annual.Income..k.. Spending.Score..1.100.
## 1  25.27273           25.72727              79.363636
## 2  38.23077           30.38462              35.076923
## 3  48.75000           24.58333               9.583333
## 4  32.20000          109.70000              82.000000
## 5  32.86207           78.55172              82.172414
## 6  61.44444           51.18519              50.444444
## 7  24.44828           56.37931              50.724138
## 8  42.63636          108.18182              21.272727
## 9  46.16000           61.32000              46.360000
## 10 39.36364           79.13636              13.363636
##
## Clustering vector:
##   [1]  2  1  3  1  2  1  3  1  3  1  3  1  3  1  3  1  2  1  2  1  2  1  3  1  3
##  [26]  1  2  1  2  1  3  1  3  1  3  1  3  1  2  1  6  1  2  7  2  1  6  7  2  2
##  [51]  6  7  7  6  6  2  6  6  7  6  6  7  6  6  6  7  9  6  7  7  6  9  6  6  6
##  [76]  7  9  9  7  9  6  9  6  9  7  9  6  7  7  9  6  7  9  9  7  7  9  7  9  7
## [101]  7  9  6  7  9  7  6  9  6  6  6  7  9  7  7  7  6  9  9  9  7  9  9  5 10
## [126]  5  9  5 10  5 10  5  7  5 10  5 10  5 10  5 10  5  7  5 10  5  9  5 10  5
## [151] 10  5 10  5 10  5 10  5 10  5  9  5 10  5 10  5 10  5 10  5 10  5 10  5 10
## [176]  5 10  5  8  5  8  4  8  4  8  4  8  4  8  4  8  4  8  4  8  4  8  4  8  4
##
## Within cluster sum of squares by cluster:
##  [1] 4099.818 2622.308 2830.083 2605.700 4148.759 3909.407 4661.793 3520.364
##  [9] 3278.560 5416.773
##  (between_SS / total_SS =  88.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

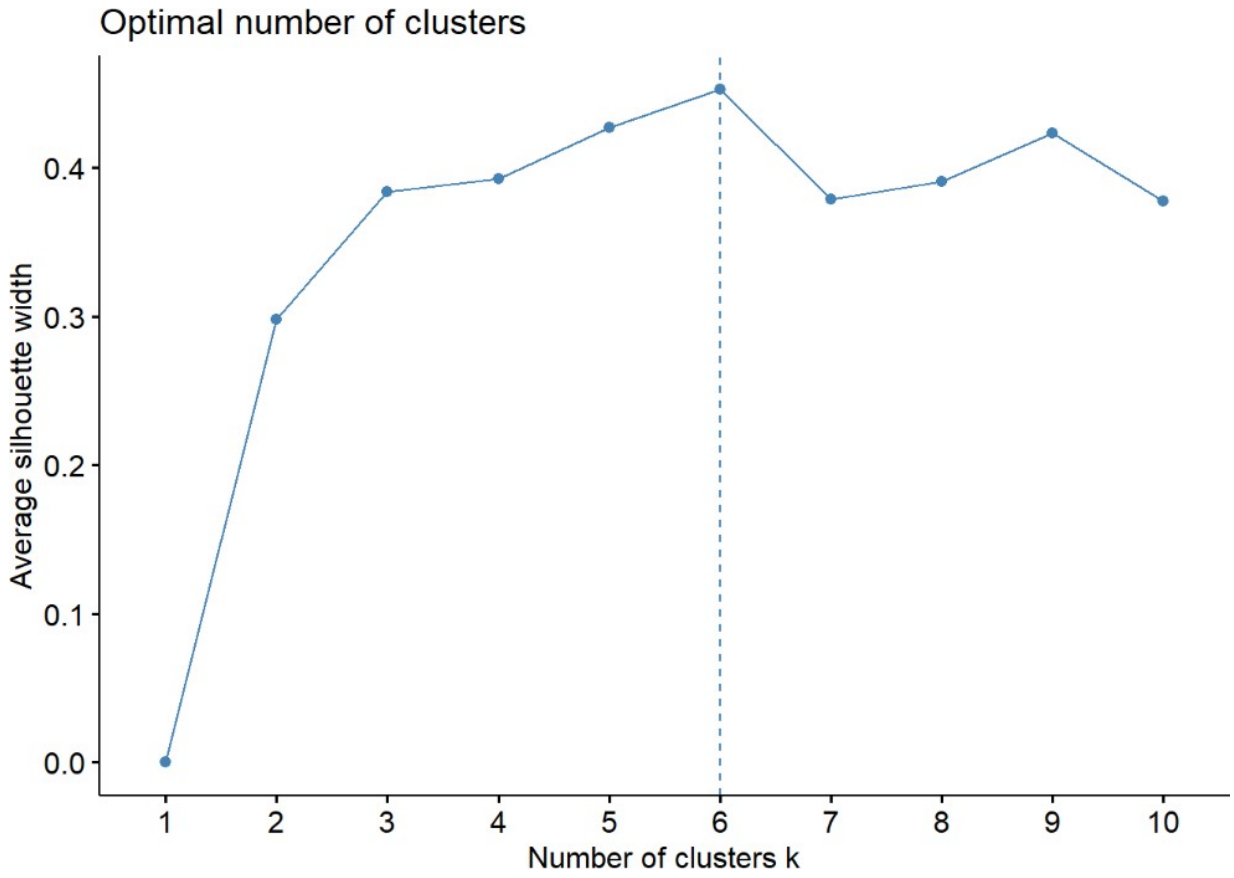
Performance Metrics for(k=10) = 88%

```
library(NbClust)
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 4.2.2
```

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

```
silhouette_No_of_clusters_k_VS_Avg_Silhouette_width<- function(){
fviz_nbclust(Mall_Customers[,3:5], kmeans, method = "silhouette")
}
silhouette_No_of_clusters_k_VS_Avg_Silhouette_width()
```

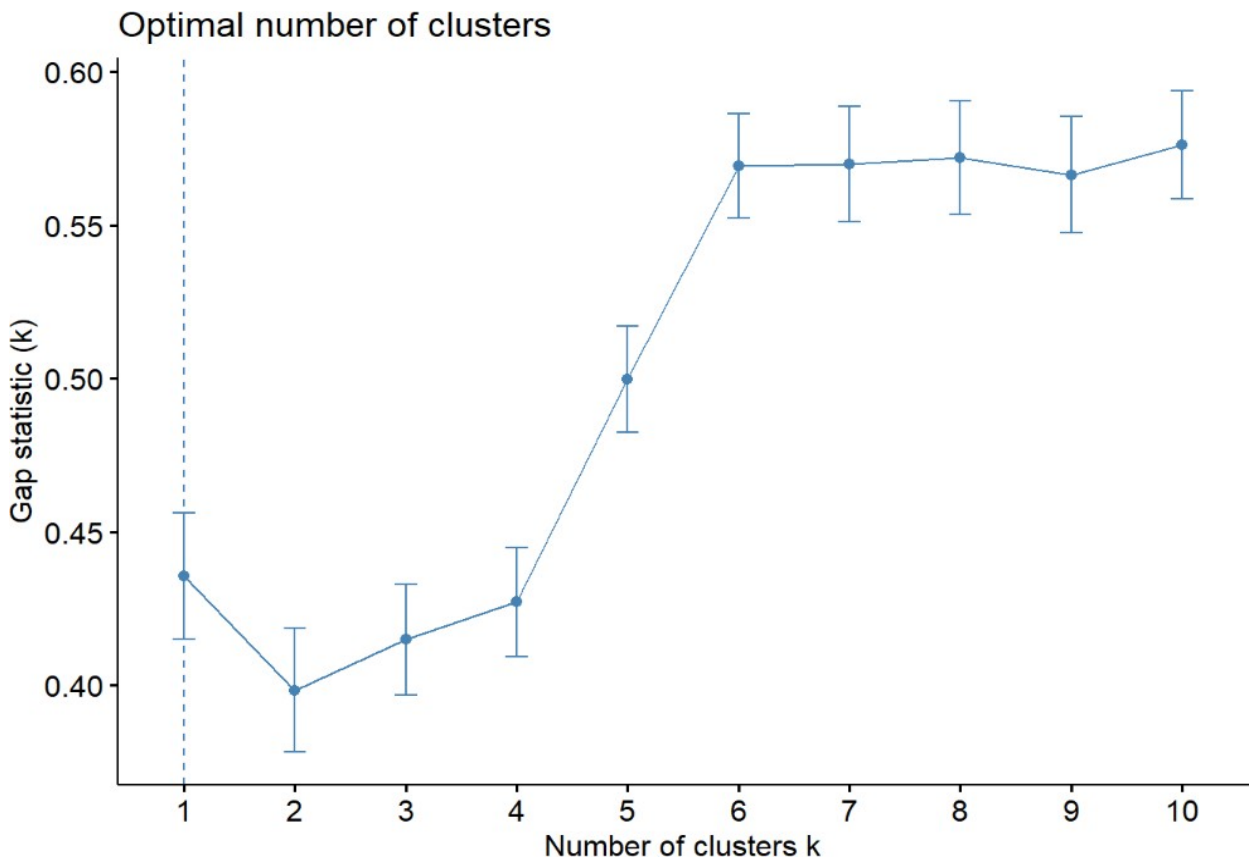


Gap Statistic Method

We can use this method to any of the clustering method like K-means, hierarchical clustering etc. Using the gap statistic, one can compare the total intracluster variation for different values of k along with their expected values under the null reference distribution of data. With the help of Monte Carlo simulations, one can produce the sample dataset. For each variable in the dataset, we can calculate the range between min(xi) and max (xj) through which we can produce values uniformly from interval lower bound to upper bound.

For computing the gap statistics method we can utilize the clusGap function for providing gap statistic as well as standard error for a given output.

```
set.seed(125)
stat_gap <- clusGap(Mall_Customers[,3:5], FUN = kmeans, nstart = 25,
            K.max = 10, B = 50)
fviz_gap_stat(stat_gap)
```



Now, let us take k = 6 as our optimal cluster –

```
k6<-kmeans(Mall_Customers[,3:5],6,iter.max=100,nstart=50,algorithm="Lloyd")
```

In the output of our kmeans operation, we observe a list with several key information. From this, we conclude the useful information being –

cluster – This is a vector of several integers that denote the cluster which has an allocation of each point. totss – This represents the total sum of squares. centers – Matrix comprising of several cluster centers withinss – This is a vector representing the intra-cluster sum of squares having one component per cluster. tot.withinss – This denotes the total intra-cluster sum of squares. betweenss – This is the sum of between-cluster squares. size – The total number of points that each cluster holds.

Visualizing the Clustering Results using the First Two Principle Components

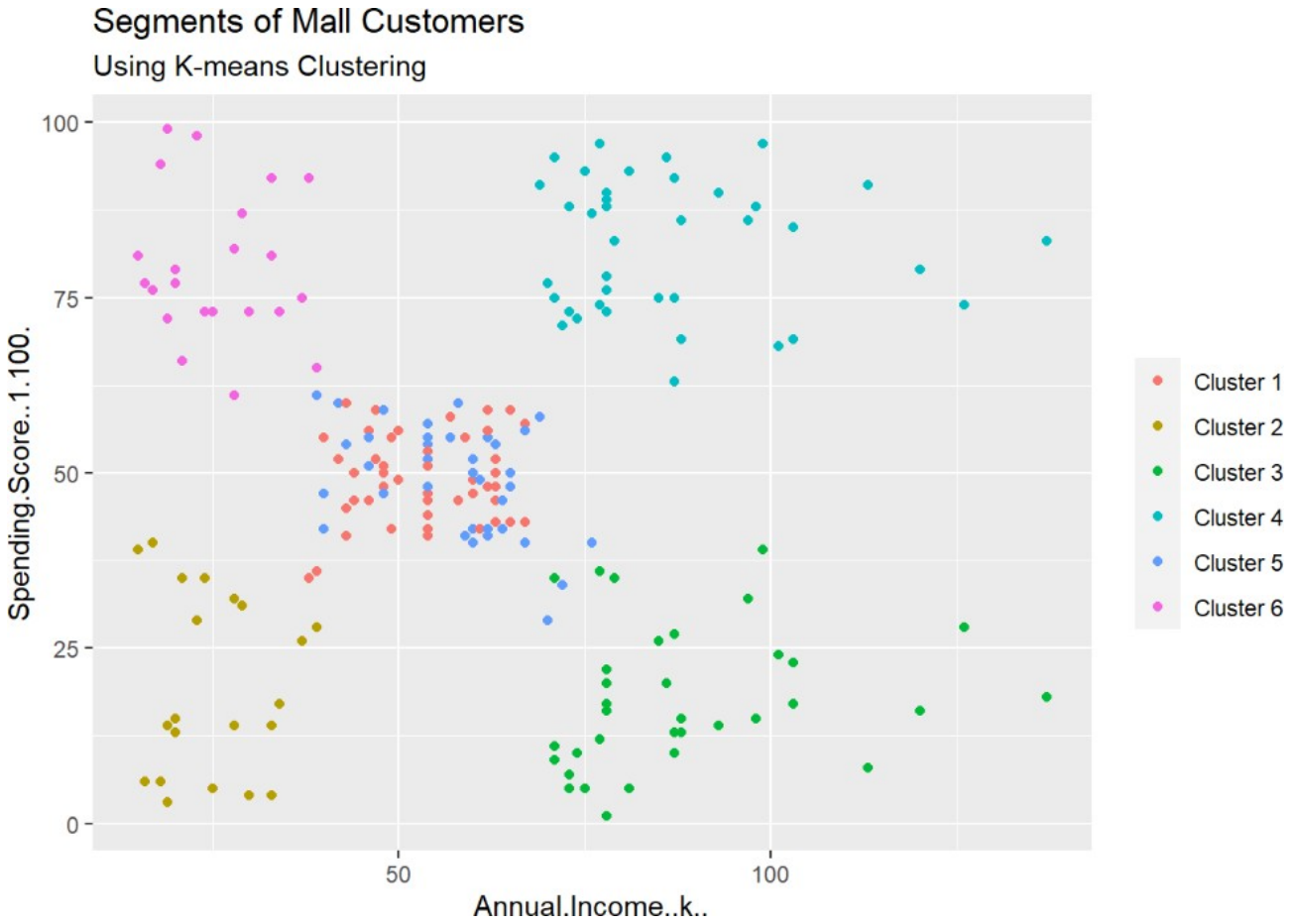
```
pcclust=prcomp(Mall_Customers[,3:5],scale=FALSE) #principal component analysis
summary(pcclust)
```

```
## Importance of components:
##                           PC1     PC2     PC3
## Standard deviation     26.4625 26.1597 12.9317
## Proportion of Variance  0.4512  0.4410  0.1078
## Cumulative Proportion   0.4512  0.8922  1.0000
```

```
pcclust$rotation[,1:2]
```

```
##                                PC1        PC2
## Age                     0.1889742 -0.1309652
## Annual.Income..k..     -0.5886410 -0.8083757
## Spending.Score..1.100. -0.7859965  0.5739136
```

```
set.seed(1)
K_means_Plot_Annual_Income_VS_Spending_Score <- function(){
ggplot(Mall_Customers, aes(x =Annual.Income..k.., y = Spending.Score..1.100.)) +
  geom_point(stat = "identity", aes(color = as.factor(k6$cluster))) +scale_color_discrete(name=" ",breaks=c("1", "2", "3",
"4", "5","6"),labels=c("Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4", "Cluster 5","Cluster 6")) +ggtitle("Segments of M
all Customers", subtitle = "Using K-means Clustering")
}
K_means_Plot_Annual_Income_VS_Spending_Score()
```



From the above visualization, we observe that there is a distribution of 6 clusters as follows –

Cluster 6 and 4 – These clusters represent the customer_data with the medium income salary as well as the medium annual spend of salary.

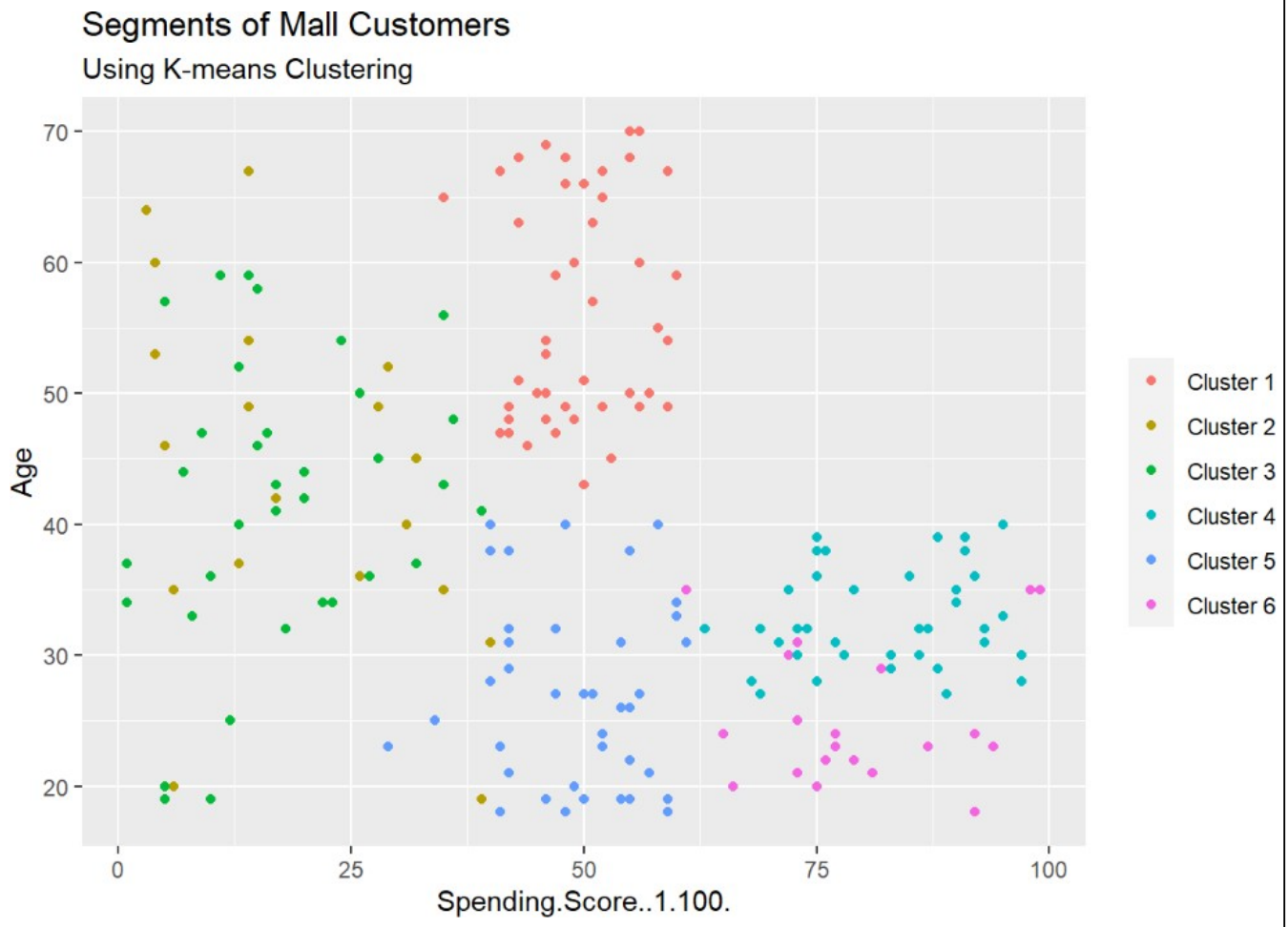
Cluster 1 – This cluster represents the customer_data having a high annual income as well as a high annual spend.

Cluster 3 – This cluster denotes the customer_data with low annual income as well as low yearly spend of income.

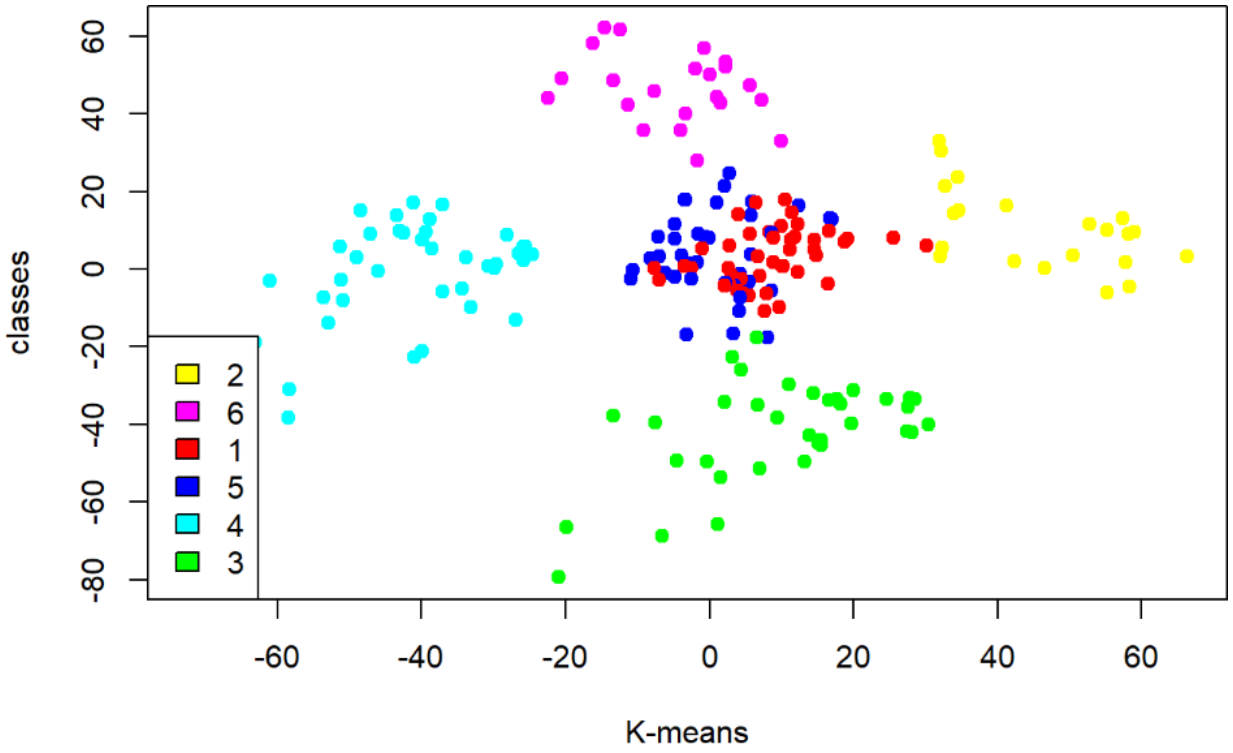
Cluster 2 – This cluster denotes a high annual income and low yearly spend.

Cluster 5 – This cluster represents a low annual income but its high yearly expenditure.

```
K_means_Plot_Spending_Score_VS_Age <- function(){
ggplot(Mall_Customers, aes(x =Spending.Score..1.100., y =Age)) +
  geom_point(stat = "identity", aes(color = as.factor(k6$cluster))) +
  scale_color_discrete(name=" ",
                       breaks=c("1", "2", "3", "4", "5","6"),
                       labels=c("Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4", "Cluster 5","Cluster 6")) +
  ggtitle("Segments of Mall Customers", subtitle = "Using K-means Clustering")
}
K_means_Plot_Spending_Score_VS_Age()
```




```
kCols=function(vec){cols=rainbow (length (unique (vec)))
return (cols[as.numeric(as.factor(vec))])}

digCluster<-k6$cluster; dignm<-as.character(digCluster); # K-means clusters
Plot_K_means_VS_Classes <- function(){
plot(pcclust$x[,1:2], col =kCols(digCluster),pch =19,xlab ="K-means",ylab="classes")
legend("bottomleft",unique(dignm),fill=unique(kCols(digCluster)))
}
Plot_K_means_VS_Classes()
```



Cluster 4 and 1 – These two clusters consist of customers with medium PCA1 and medium PCA2 score.

Cluster 6 – This cluster represents customers having a high PCA2 and a low PCA1.

Cluster 5 – In this cluster, there are customers with a medium PCA1 and a low PCA2 score.

Cluster 3 – This cluster comprises of customers with a high PCA1 income and a high PCA2.

Cluster 2 – This comprises of customers with a high PCA2 and a medium annual spend of income.

With the help of clustering, we can understand the variables much better, prompting us to take careful decisions. With the identification of customers, companies can release products and services that target customers based on several parameters like income, age, spending patterns, etc. Furthermore, more complex patterns like product reviews are taken into consideration for better segment

References

1. https://www.kaggle.com/datasets/vjchoudhary7/customer-segmentation-tutorial-in-python
2. https://hbr.org/topic/subject/market-segmentation
3. https://hbr.org/2013/06/a-new-framework-for-customer-s
4. https://www.hbs.edu/faculty/research/publications/Pages/default.aspx?q=Market%20Segmentation%20And%20Target%20Market%20Selection
5. https://ironlinx.com/market-segmentation-for-ecommerce-brands/
6. https://supermetrics.com/blog/ecommerce-customer-segmentation

Publicity

We have uploaded our project which we did in the RMD and html file in github and pasted the link below. The file consists of all the modules which we covered in our project in order.

https://github.com/AayushShukla03/Product-and-Customer-Segmentation

Thank You