

# Analysis of Customer Data using K-means Clustering

## Application Area Project

Sriram Reddy Arabelli      Vamsi Gontu      Sanyukta Nair  
Roja Kuchipudi      Annanya Jain      Shubhankar Goje

## Table of contents

Introduction . . . . .	1
Motivation . . . . .	2
Problem statement . . . . .	2
Design and Implementation . . . . .	2
Data Preparation . . . . .	2
Data Pre-Processing . . . . .	5
Data Modelling . . . . .	22
Bias . . . . .	28
Conclusion . . . . .	28

## Introduction

Data has recently become one of the most important driving forces behind running a successful business. There is an unfathomable amount of big data available from various sources, including web databases and social media. Marketing professionals could benefit greatly from this massive amount of data if it can be properly processed and analyzed. They can use this processed data as a tool in their business venture by using it to gain valuable insights into their target customers.

The practice of data science is employing cutting-edge technologies to explore and analyze a vast quantity of data. It gathers data in a more sophisticated and organized manner, and it organizes that data. Data science is used in business to pinpoint external factors that might directly or indirectly affect company's operations and income. Data science and digital marketing work hand in hand because it is such a crucial aspect in operating a firm. We may utilize data analytics and data science to create marketing plans that will work for the business

since they can anticipate market trends and, in general, make practical future predictions about how the firm will perform in the future.

## **Motivation**

Faster service and personalization appear to be valued by most e-commerce customers. In addition, marketers must, as always, compete with rivals for the attention of their target clients if they are to succeed in the commercial world. And for this reason, sales and marketing need data science. Technology has advanced significantly in the previous ten to fifteen years and particularly in the area of data science. With such a vast quantity of data at our fingertips, using it for marketing initiatives is only logical. Businesses won't require many data scientists and analysts now to produce knowledge on their target market.

A lot of automation and machine learning algorithms make it possible to examine a lot of data in a very short amount of time. The use of data analytics in marketing is no longer a distant possibility. These techniques are already being used by many large businesses to increase sales. Businesses will fall far behind those that seize this chance if they do not start doing so right away. Businesses may use strong marketing techniques and data science to better understand the wants and desires of their consumers and draw in new ones.

## **Problem statement**

Business want to understand the customer behaviors to perform the campaigns and advertising to increase the customer base and to check the loyal customers.

1. Which age group has the highest earnings and spendings?
2. Is there any correlation between income earned and spending?

## **Design and Implementation**

### **Data Preparation**

Data is fetched from a kaggle to github and read the data in R by using the github link. Data opted is related to perform customer segmentation with two hundred rows and five columns with two categorical columns(Age, Gender) and three numerical columns(CustomerID, Annual Income, Spending Score) with CustomerID as primary key.

The downloaded binary packages are in

```
/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded_packages
```

The downloaded binary packages are in  
/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

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

```
# reading the data from github
customer_data=read.csv("https://raw.githubusercontent.com/sriram8113/Data-Science-as-field
```

```
#sample Represntation of Data
head(customer_data)
```

	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

```
#structure of the data

str(customer_data)
```

```
'data.frame':  200 obs. of  5 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 ...
```

```
#Displaying names of the columns
```

```
names(customer_data)
```

```
[1] "CustomerID"      "Gender"           "Age"
[4] "Annual.Income..k.." "Spending.Score..1.100."
```

```
#selecting required columns for further analysis
```

```
numerical_customer_columns_data = customer_data%>%select('Age','Annual.Income..k..','Spending.Score..1.100.')
```

```
#sample representation of numerical data
```

```
head(numerical_customer_columns_data)
```

	Age	Annual.Income..k..	Spending.Score..1.100.
1	19	15	39
2	21	15	81
3	20	16	6
4	23	16	77
5	31	17	40
6	22	17	76

```
#summary of numerical data
```

```
summary_numerical = apply(numerical_customer_columns_data, 2, summary)
```

```
summary_numerical
```

	Age	Annual.Income..k..	Spending.Score..1.100.
Min.	18.00	15.00	1.00
1st Qu.	28.75	41.50	34.75
Median	36.00	61.50	50.00
Mean	38.85	60.56	50.20
3rd Qu.	49.00	78.00	73.00
Max.	70.00	137.00	99.00

```
#standard Deviation of numerical columns
```

```
apply(numerical_customer_columns_data, 2, sd)
```

	Age	Annual.Income..k..	Spending.Score..1.100.
	13.96901	26.26472	25.82352

```
#summary of total data
summary(customer_data)
```

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

## Data Pre-Processing

Pre-processing of data is a mandatory process after getting any data. In this report, data cleansing techniques such as finding the missing, NA values, outlier verification are performed. One outlier could be found in the numerical data and removed the outlier in the data and plotted the charts to verify the difference.

## Data Cleansing

```
#checking for Na values

which(is.na(customer_data))
```

```
integer(0)
```

we can see that there are no Na values in the data.

```
sum(is.na(customer_data))
```

```
[1] 0
```

```
#Dimensions of the data
```

```
dim(customer_data)
```

```
[1] 200  5
```

```
#Selecting unique rows
```

```
customer_data1<- customer_data%>% distinct()  
dim(customer_data1)
```

```
[1] 200  5
```

we can see that there are no duplicated values

### Outlier validation

```
dim(customer_data)
```

```
[1] 200  5
```

```
quartiles <- quantile(customer_data$Annual.Income..k.., probs=c(.25, .75), na.rm = FALSE)  
IQR <- IQR(customer_data$Annual.Income..k..)
```

```
Lower <- quartiles[1] - 1.5*IQR
```

```
Upper <- quartiles[2] + 1.5*IQR
```

```
customer_data <- subset(customer_data, customer_data$Annual.Income..k.. > Lower & customer_data$Annual.Income..k.. < Upper)
```

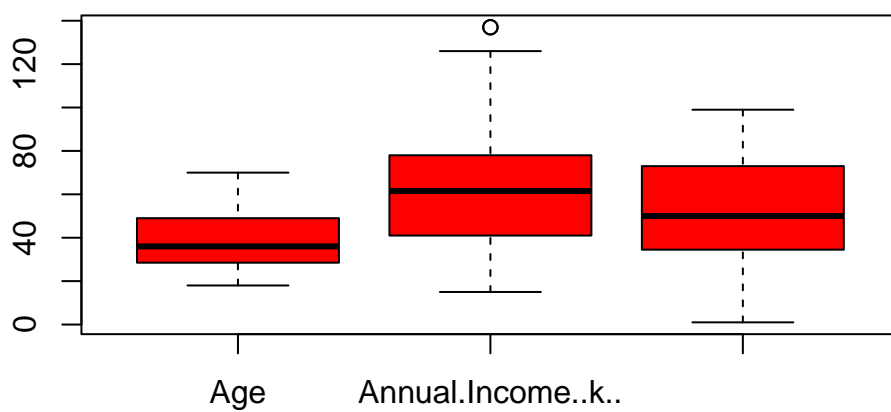
```
dim(customer_data)
```

```
[1] 198  5
```

```
#Boxplot
```

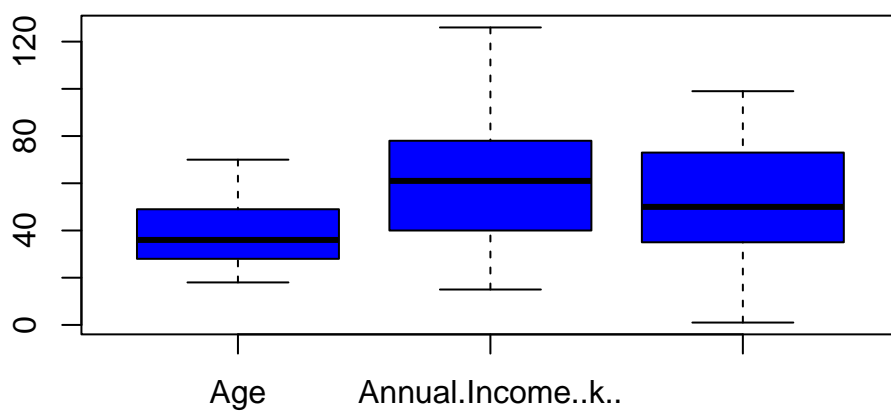
```
boxplot(numerical_customer_columns_data, col = "red", main = 'Boxplot of Numerical Data')
```

**Boxplot of Numerical Data**

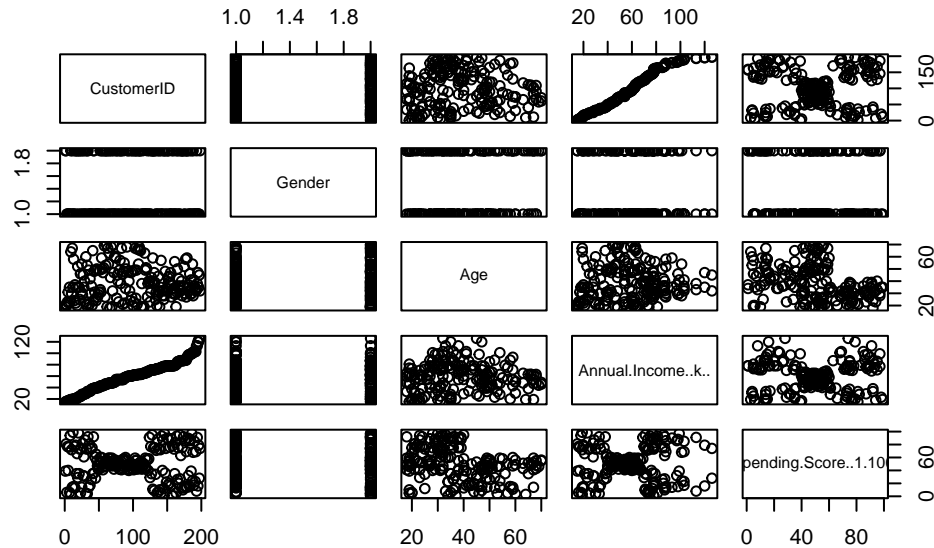


```
#Boxplot
boxplot(customer_data[,3:5], main = "Boxplot of Numerical Data", col = 'Blue')
```

**Boxplot of Numerical Data**



```
#plotting all the data together.
plot(customer_data)
```



## Data Transformation

Transformation of data plays a vital role in structuring the data by using few techniques such as normalization, attribute selection, and feature selection.

```
#mean
means = apply(numerical_customer_columns_data, 2, mean)
```

```
#standard deviation
sds = apply(numerical_customer_columns_data, 2, sd)
```

```
#sample representation of normalised data
```

```
normalised_numerical_customer_data = scale(numerical_customer_columns_data, center=means,
head(normalised_numerical_customer_data)
```

```
Age Annual.Income..k.. Spending.Score..1.100.
```

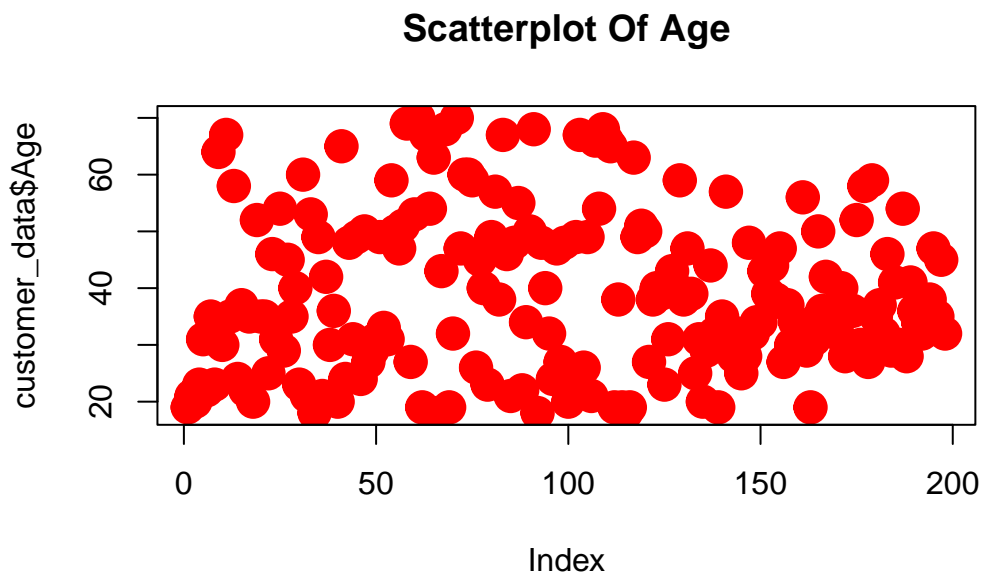


[1,]	-1.4210029	-1.734646	-0.4337131
[2,]	-1.2778288	-1.734646	1.1927111
[3,]	-1.3494159	-1.696572	-1.7116178
[4,]	-1.1346547	-1.696572	1.0378135
[5,]	-0.5619583	-1.658498	-0.3949887
[6,]	-1.2062418	-1.658498	0.9990891

## Analysis of Data using Visualizations

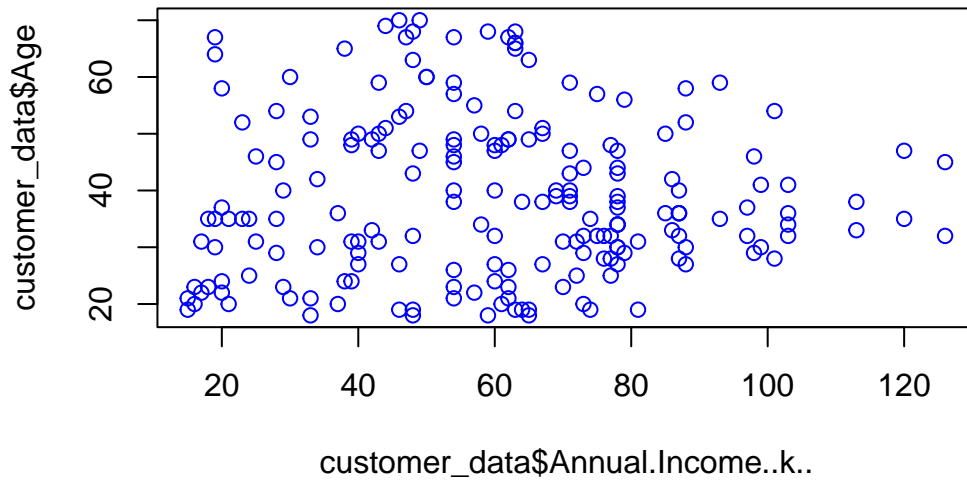
The below code represents the various scatter plots with the combination of multiple numerical columns. Exploratory data analysis such as uni-variate analysis for the Age, Gender and bivariate analysis for the annual income, spending score versus age

```
#scatter plot
plot(customer_data$Age, col = "red", lwd = 10, main = "Scatterplot Of Age")
```



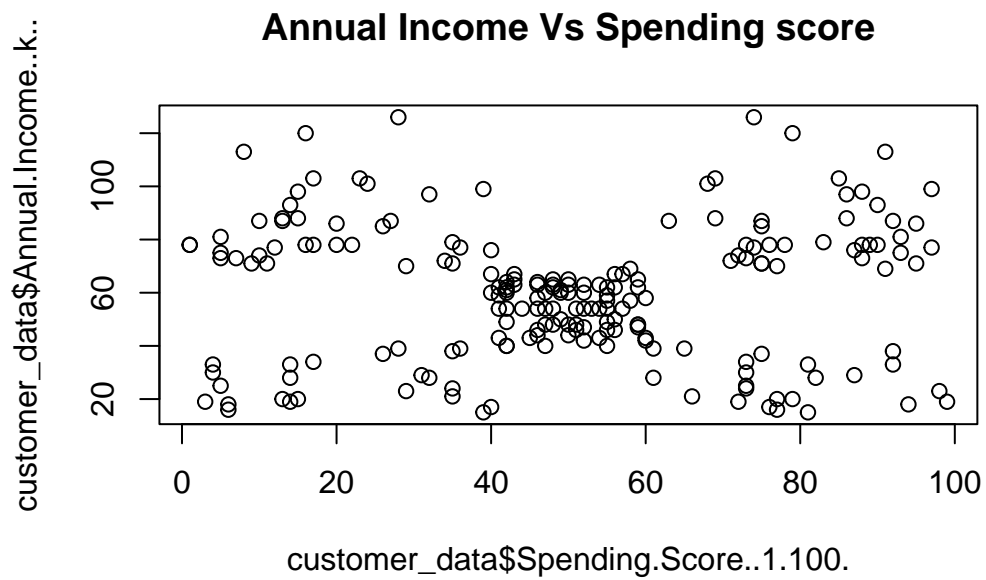
```
#Scatter plot
plot(customer_data$Age~customer_data$Annual.Income..k.., col = 'blue')
title(main ='Annual Income Vs Age')
```

## Annual Income Vs Age



```
#Scatter plot
```

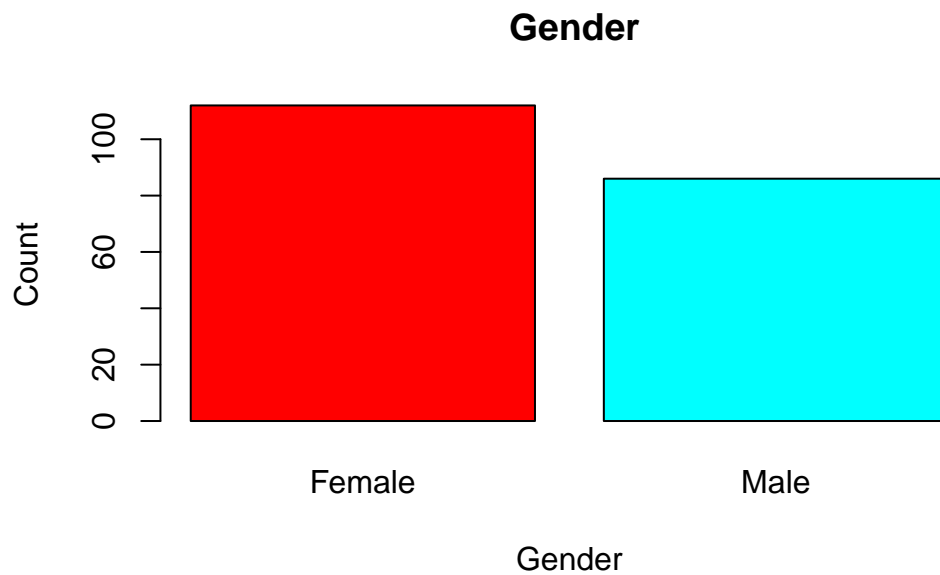
```
plot(customer_data$Annual.Income..k..~customer_data$Spending.Score..1.100.)  
title(main = 'Annual Income Vs Spending score')
```



Below code shows the charts of the individual columns spread and distribution. Bar plot shows the number of customers in each gender in the data set.

```
#Barplot representing gender

barplot(table(customer_data$Gender),main=" Gender",
        ylab="Count",
        xlab="Gender",
        col=rainbow(2),
        legend=rownames(customer_data$Gender))
```

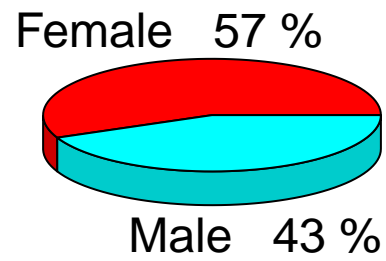


```
#Pie chart
install.packages('plotrix', repos = "http://cran.us.r-project.org")
```

The downloaded binary packages are in  
 /var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

```
percentage=round(table(customer_data$Gender)/sum(table(customer_data$Gender))*100)
labels_gender=paste(c("Female","Male")," ",percentage,"%",sep=" ")
library(plotrix)
pie3D(table(customer_data$Gender), main="Pie Chart of Female and Male", labels = labels_g
```

## Pie Chart of Female and Male



From the above two charts, it can be concluded that female customers are more in number than the male customers.

```
#Histogram of Age
```

```
ggplot(customer_data,aes(x= Age, fill=Gender))+geom_histogram(bins = 50)+ggtitle('Distribu
```



From the above histogram,

1. with increase in the age number of customers decreased.
2. Maximum number of customers are in the age between 30 and 40years.

```
ggplot(customer_data,aes(x= `Spending.Score..1.100.` , col=Gender)) + geom_freqpoly(bins=50)
```



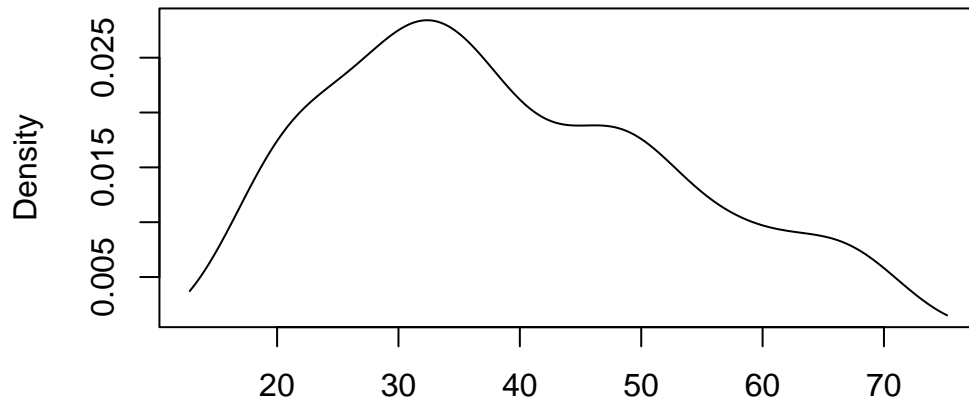
From the above chart, it can be seen spending of women are higher than the male.

```
# Kernal Density Plot
install.packages('kdensity', repos = "http://cran.us.r-project.org")
```

The downloaded binary packages are in  
 /var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

```
library(kdensity)
KDE_age <- kdensity(x=customer_data$Age, kernel = 'gaussian')
plot(KDE_age)
```

```
kdensity(x = customer_data$Age, kernel = "gaussian")
```



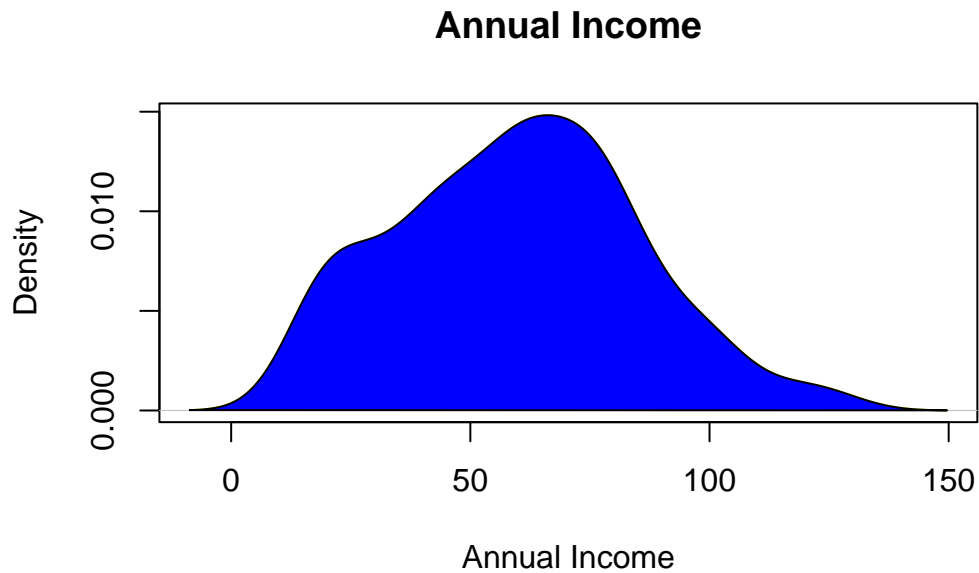
N = 198 Bandwidth = 4.381 ('nrd0')

From the above Gaussian distribution, highest density is in the range of 30 to 40 so we can say that there are more number of customers in the same age range.

```
#Density plot

plot(density(customer_data$Annual.Income..k..),
     col="yellow",
     main="Annual Income",
     xlab="Annual Income",
     )
polygon(density(customer_data$Annual.Income..k..),
        col="blue")
```





from the above density graph, it shows the maximum customers annual income ranges between \$50K - \$100K.

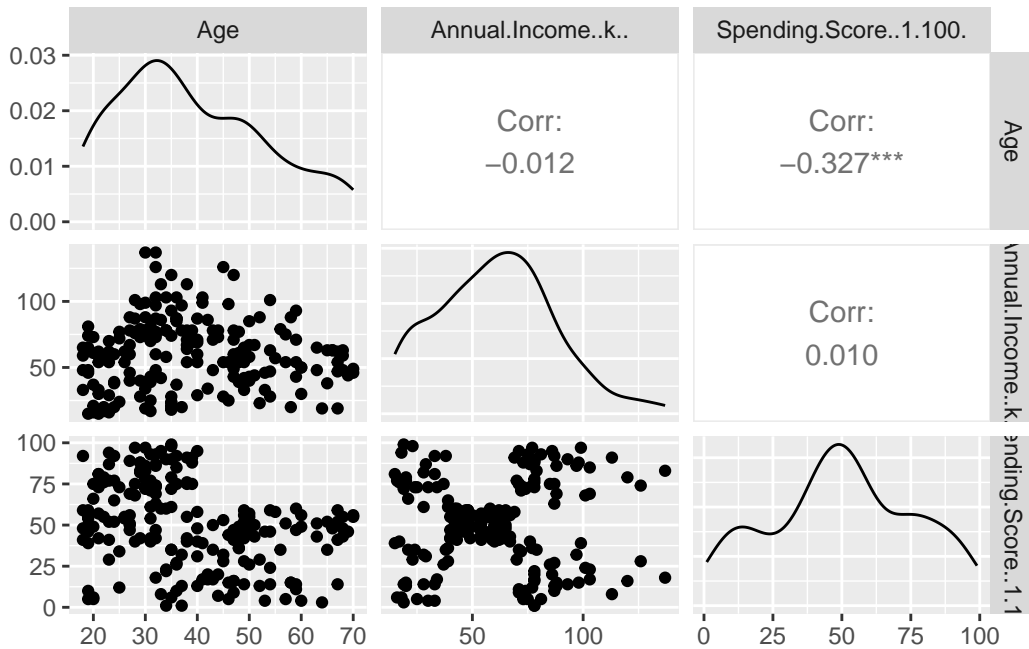
```
install.packages('GGally', repos = "http://cran.us.r-project.org")
```

The downloaded binary packages are in  
/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

```
library(GGally)
```

Registered S3 method overwritten by 'GGally':  
method from  
+.gg ggplot2

```
ggpairs(numerical_customer_columns_data)
```



from the above graph, we can conclude:

1. correlation between annual income and Age: Negative
2. correlation between spending scores and Age: Negative
3. correlation between spending scores and annual income: Positive

```
new_data <- data.frame(customer_data)
new_data$Age1 <- cut(new_data$Age, seq(10,70,10))
head(new_data)
```

	CustomerID	Gender	Age	Annual.Income..k..	Spending.Score..1.100.	Age1
1	1	Male	19	15	39	(10,20]
2	2	Male	21	15	81	(20,30]
3	3	Female	20	16	6	(10,20]
4	4	Female	23	16	77	(20,30]
5	5	Female	31	17	40	(30,40]
6	6	Female	22	17	76	(20,30]

Created a new column named Age1 for further analysis.

```
Groupby_age_data <- new_data %>% group_by(Age1)%>%summarise(total_spending = sum(Spending))
Groupby_age_data
```

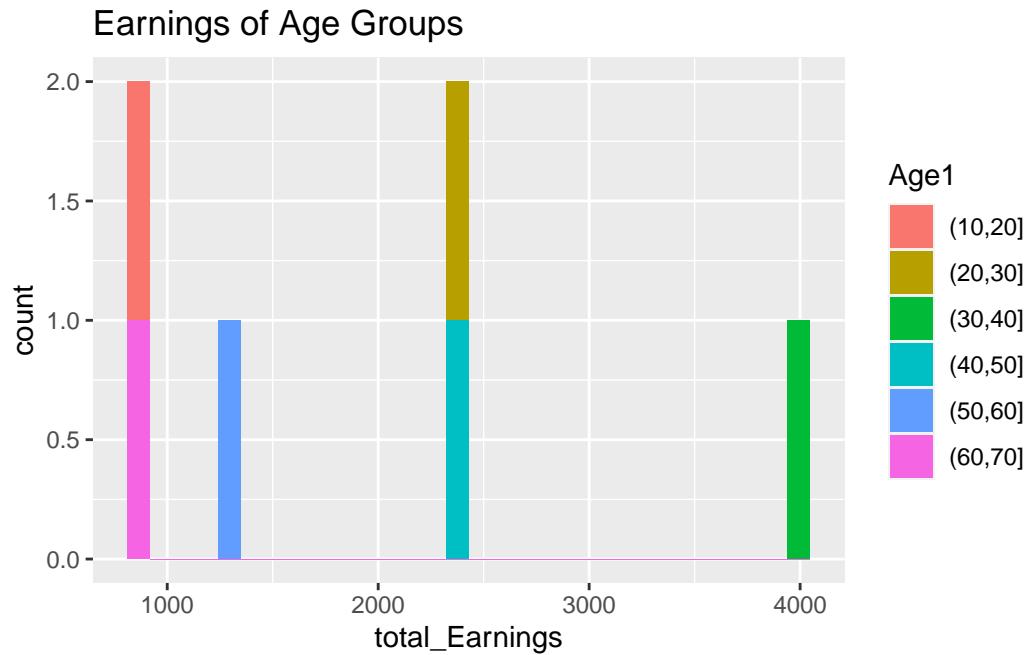
```
# A tibble: 6 x 3
  Age1    total_spending total_Earnings
  <fct>      <int>          <int>
1 (10,20]      759            869
2 (20,30]     2942           2417
3 (30,40]     3432           3981
4 (40,50]     1307           2417
5 (50,60]      748           1304
6 (60,70]      751            850
```

Above table clearly shows that:

1. Age group between 30,40 has the highest spending and earnings.
2. Age group between 60,70 has the lowest spending and earnings.

```
ggplot(Groupby_age_data,aes(x= total_Earnings, fill=
Age1))+geom_histogram()+ggtitle('Earnings of Age Groups')
```

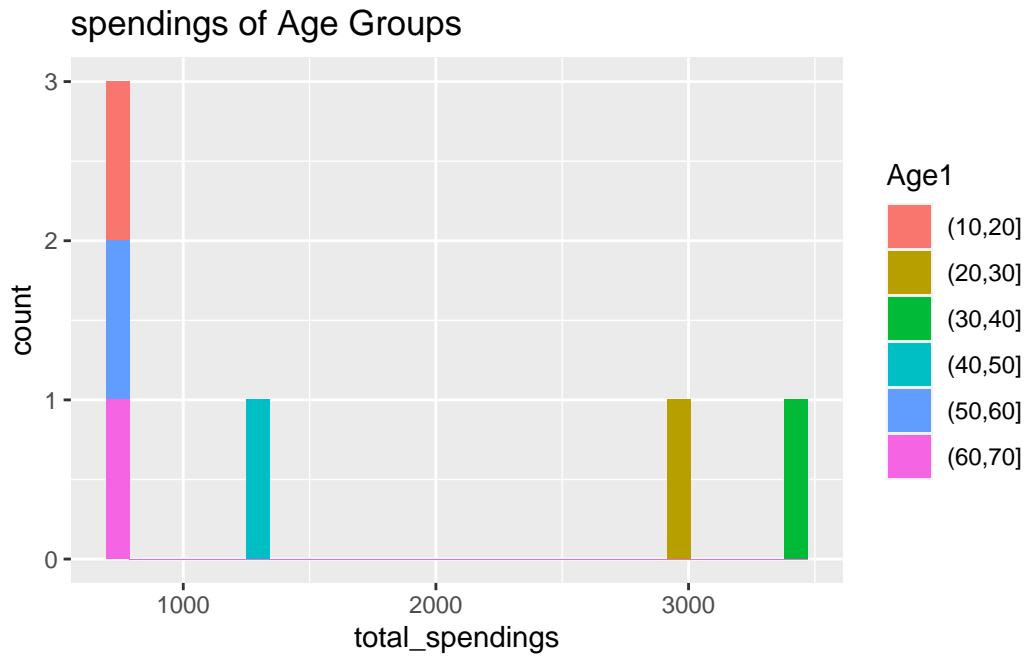
`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



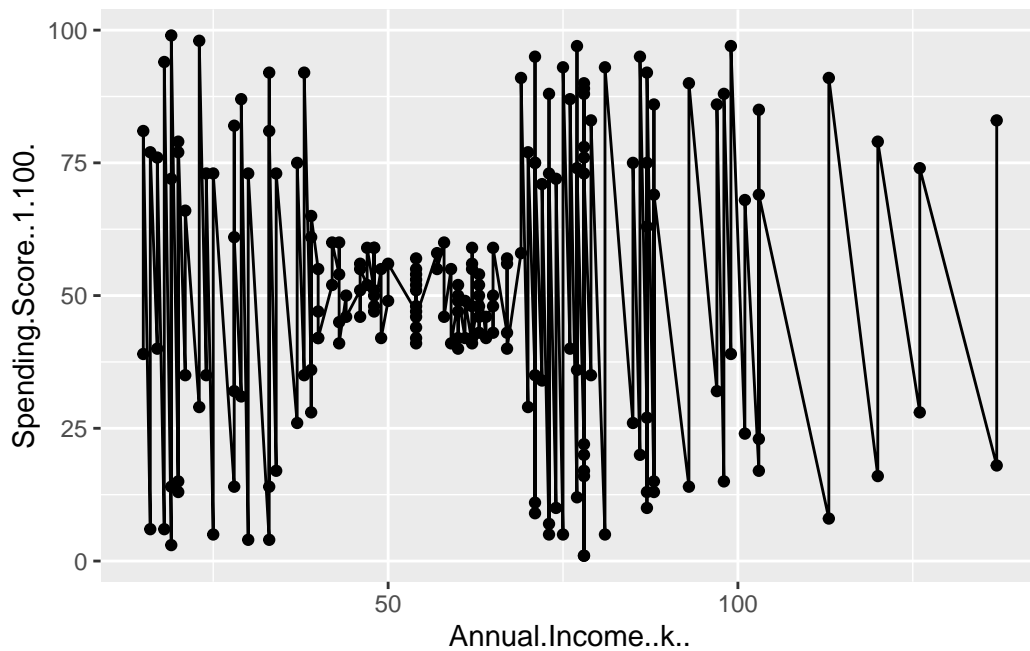
Above chart shows that with age 30 to 40 (green color bar) shows the total highest earnings.

```
ggplot(Groupby_age_data, aes(x= total_spendings, fill=
Age1))+geom_histogram()+ggtitle('spendings of Age Groups')
```

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



```
ggplot(numerical_customer_columns_data, aes(x =Annual.Income..k..)) +
  geom_point(aes(y = Spending.Score..1.100.))+geom_line(aes(y = Spending.Score..1.100.))
```



## Data Modelling

### K-means Clustering

When an unlabeled data is available its always suggestible to go ahead with the unsupervised ML model. K-means clustering is one of the models opted here in order to generate the different clusters. Silhouette method, is used here to find the number of clusters and found that  $K=6$ .

#### Finding optimal no of clusters

```
install.packages('NbClust',repos = "http://cran.us.r-project.org")
```

The downloaded binary packages are in

/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

```
install.packages('factoextra',repos = "http://cran.us.r-project.org")
```

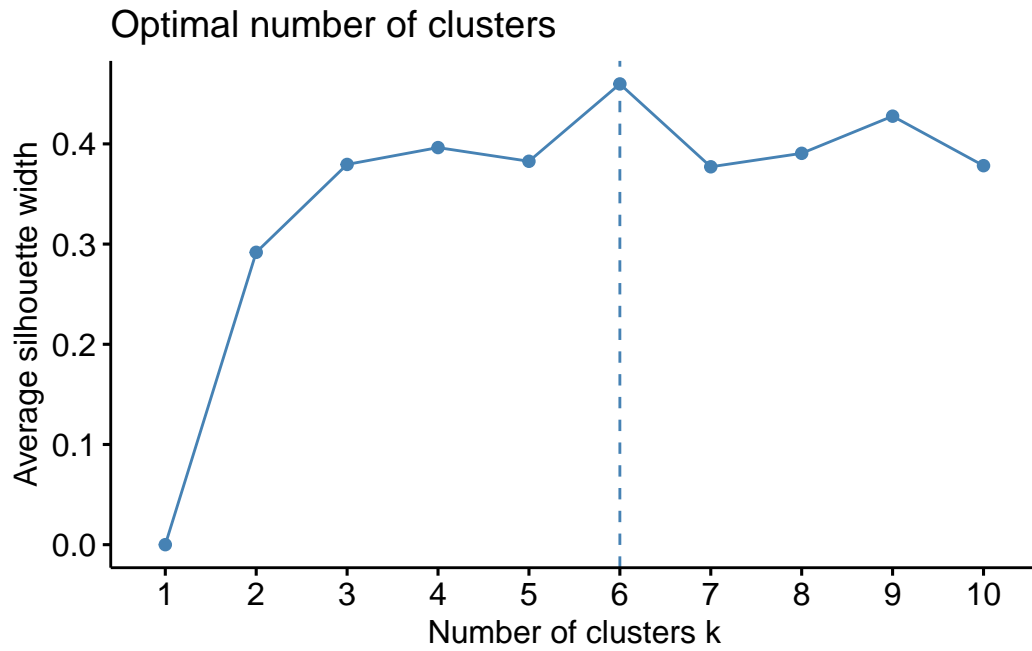
The downloaded binary packages are in

/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T//RtmptxwARl/downloaded\_packages

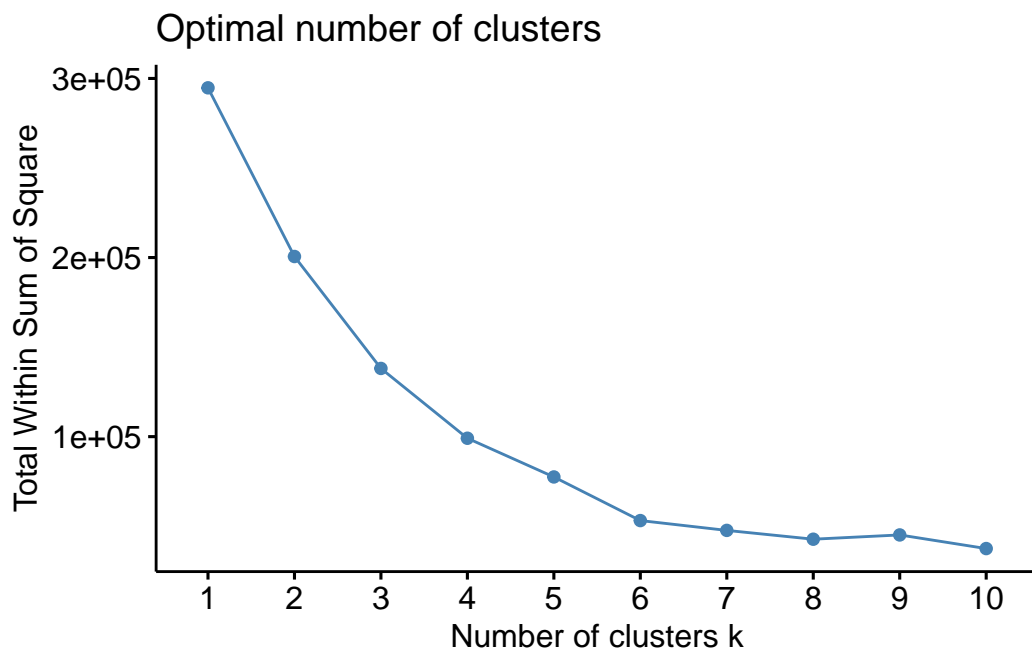
```
library(NbClust)
library(factoextra)
```

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

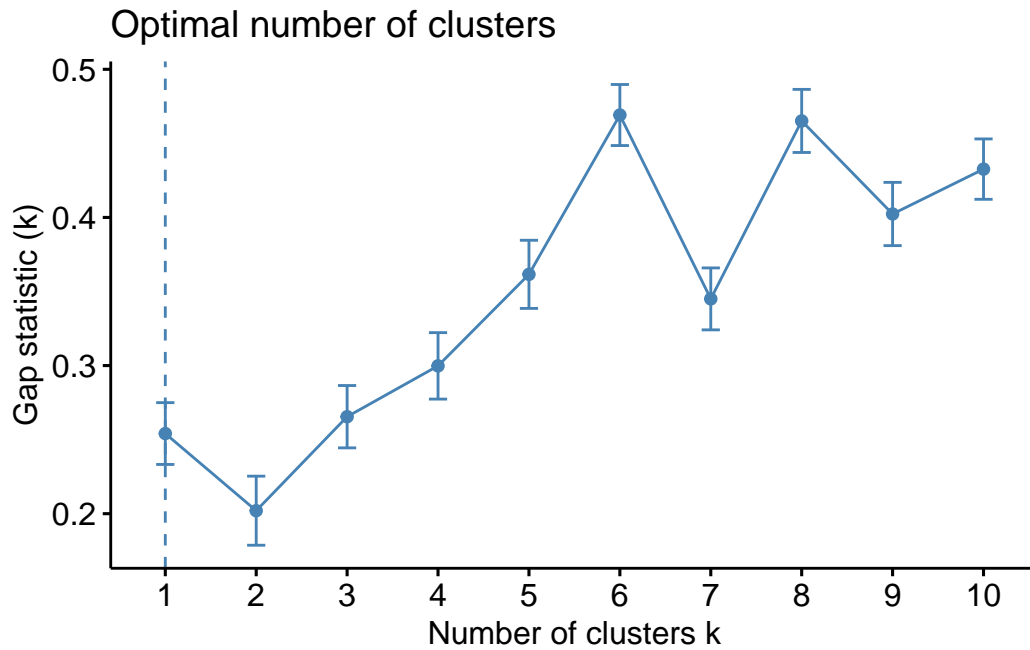
```
fviz_nbclust(customer_data[,3:5], kmeans, method = "silhouette")
```



```
fviz_nbclust(customer_data[,3:5], kmeans, method = "wss")
```



```
fviz_nbclust(customer_data[,3:5], kmeans, method = "gap_stat")
```



From these 3 methods we can see that the optimal no of clusters are 6.

```
#Applying kmeans to the data with 6 clusters

kmeans_6cluters <- kmeans(customer_data[,3:5],6)
kmeans_6cluters
```

K-means clustering with 6 clusters of sizes 11, 17, 28, 9, 38, 95

Cluster means:

	Age	Annual.Income..k..	Spending.Score..1.100.
1	27.27273	76.72727	15.36364
2	48.52941	80.29412	18.76471
3	24.82143	28.71429	74.25000
4	42.00000	106.66667	22.44444
5	32.76316	85.21053	82.10526
6	44.89474	48.70526	42.63158

Clustering vector:



```

 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
 3  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
 6  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3  6  3
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
 6  3  6  3  6  3  6  6  6  6  6  3  6  6  6  6  6  6  6  6
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
 6  3  6  6  6  3  6  6  3  6  6  6  6  6  6  6  6  6  6  6
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
 6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6  6
121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140
 6  6  6  5  1  5  2  5  2  5  2  5  1  5  1  5  2  5  1  5
141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160
 2  5  1  5  1  5  2  5  1  5  2  5  2  5  2  5  1  5  1  5
161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
 2  5  1  5  2  5  2  5  2  5  2  5  1  5  2  5  2  5  2  5
181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198
 4  5  4  5  4  5  4  5  4  5  4  5  4  5  4  5  4  5

```

Within cluster sum of squares by cluster:

```

[1] 2578.909 3168.824 9099.071 1978.222 11350.763 62300.800
(between_SS / total_SS = 69.3 %)

```

Available components:

```

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"

```

```
install.packages("ggpubr", repos = "https://cloud.r-project.org/", dependencies = TRUE)
```

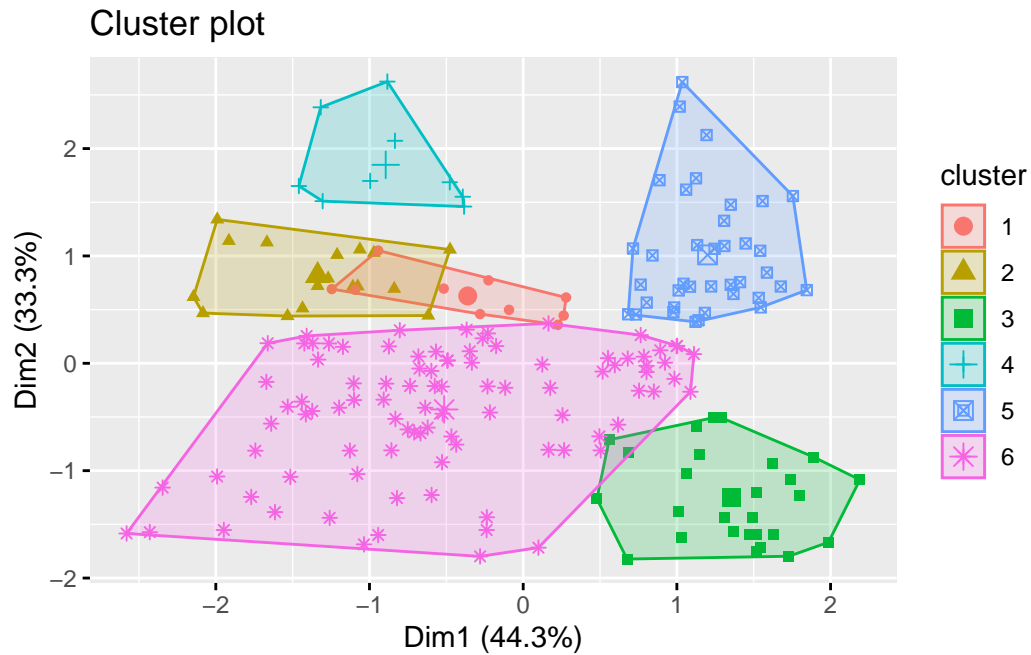
The downloaded binary packages are in

```
/var/folders/16/qnwjsjfn5011fpzs33fhppqw0000gn/T/RtmptxwARl/downloaded_packages
```

```

library(ggpubr)
fviz_cluster(kmeans_6clusters, data = customer_data[,3:5], geom = "point")

```



Here we can see the formation of 6 clusters.

```
kmeans_cluster_data <- data.frame(Cluster = kmeans_6cluters$cluster, customer_data[,3:5])
head(kmeans_cluster_data)
```

Cluster	Age	Annual.Income..k..	Spending.Score..1.100.
1	3	19	15
2	3	21	15
3	6	20	16
4	3	23	16
5	6	31	17
6	3	22	17

```
#Sizes of the clusters
kmeans_6cluters$size
```

```
[1] 11 17 28 9 38 95
```

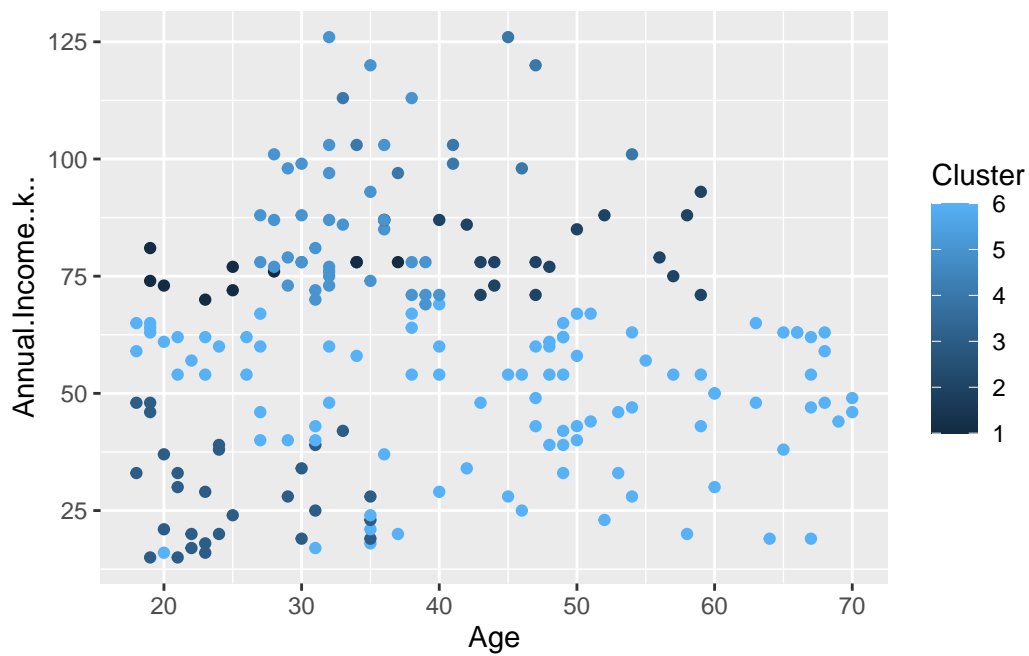
```
#Centres of the clusters
kmeans_6cluters$centers
```

	Age	Annual.Income..k..	Spending.Score..1.100.
1	27.27273	76.72727	15.36364
2	48.52941	80.29412	18.76471
3	24.82143	28.71429	74.25000
4	42.00000	106.66667	22.44444
5	32.76316	85.21053	82.10526
6	44.89474	48.70526	42.63158

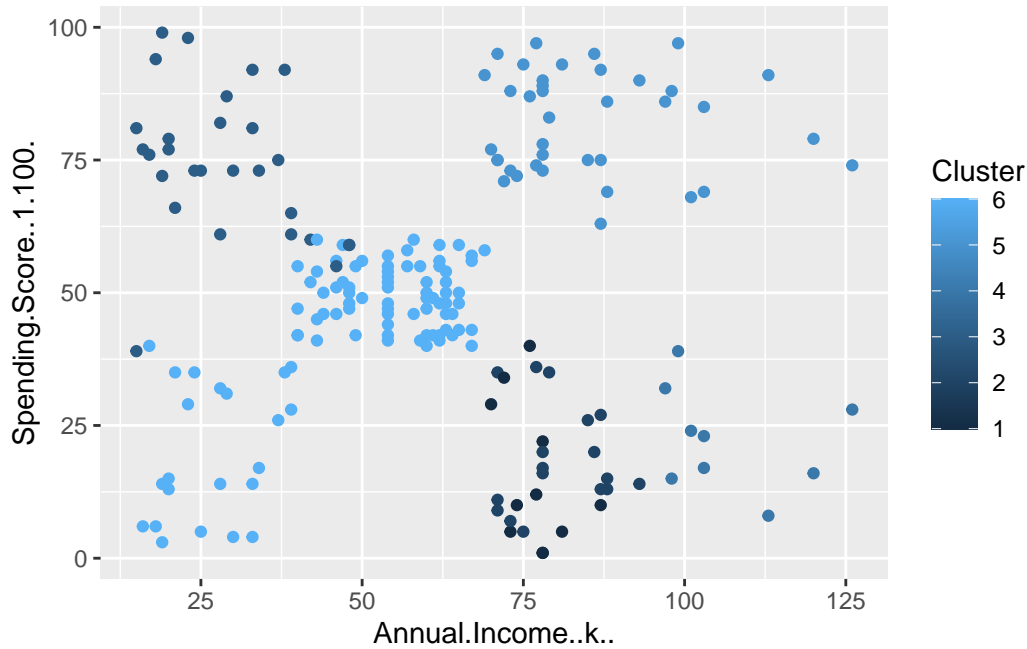
```
names(kmeans_cluster_data)
```

```
[1] "Cluster"          "Age"              "Annual.Income..k.."
[4] "Spending.Score..1.100."
```

```
ggplot() +
  geom_point(data = kmeans_cluster_data, mapping = aes(x = Age, y = Annual.Income..k.., col = Cluster))
```



```
ggplot() +
  geom_point(data = kmeans_cluster_data, mapping = aes(x = Annual.Income..k.., y = Spending.Score..1.100., col = Cluster))
```



The `echo: false` option disables the printing of code (only output is displayed).

## Bias

In general case scenarios, it is assumed that people with highest income will have the highest spending, but from the above analysis it shows that there is no such correlation between income earnings and spending.

## Conclusion

It is clearly proved that customer segmentation in marketing domain can be easily performed by the unsupervised learning to personalize marketing campaigns and advertisements more effectively. This report gives the step-by-step process of cleaning, transforming and analyzing the data in targeting and prioritizing the best customer segments. It also depends on what kind of data is being used.

By performing the best customer segmentation process, the impact on various domains of the organisation such as marketing, advertising and customer services is immense. By executing the customer segmentation, the organizations focuses on segments rather every customer individuals.