Clustering & PCA

```
#Load data
customers = read.csv('~/Downloads/Mall_Customers.csv')
summary(customers)
```

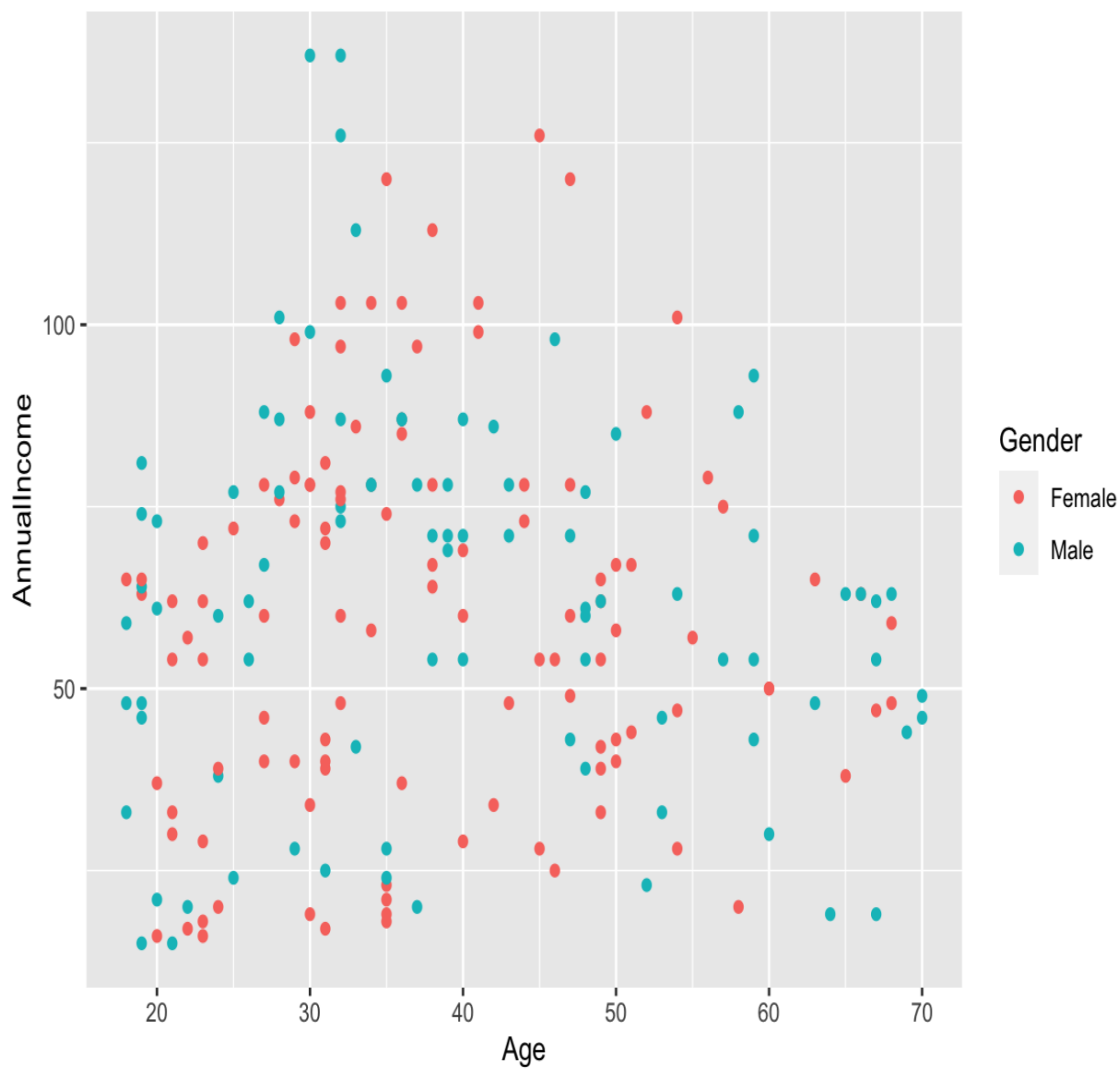
```
##    CustomerID        Gender               Age        Annual.Income..k..
##  Min.   :  1.00   Length:200         Min.   :18.00   Min.   : 15.00
##  1st Qu.: 50.75   Class :character   1st Qu.:28.75   1st Qu.: 41.50
##  Median :100.50   Mode  :character   Median :36.00   Median : 61.50
##  Mean   :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.
##  Min.   : 1.00
##  1st Qu.:34.75
##  Median :50.00
##  Mean   :50.20
##  3rd Qu.:73.00
##  Max.   :99.00
```

```
# Change name of 2 variables
names(customers)[4] <- paste('AnnualIncome')
names(customers)[5] <- paste('SpendingScore')
customers$Gender = as.factor(customers$Gender)
# Ignore customer ID since it does not have any relationship with other variable
s
customers <- customers[,2:5]
summary(customers)
```

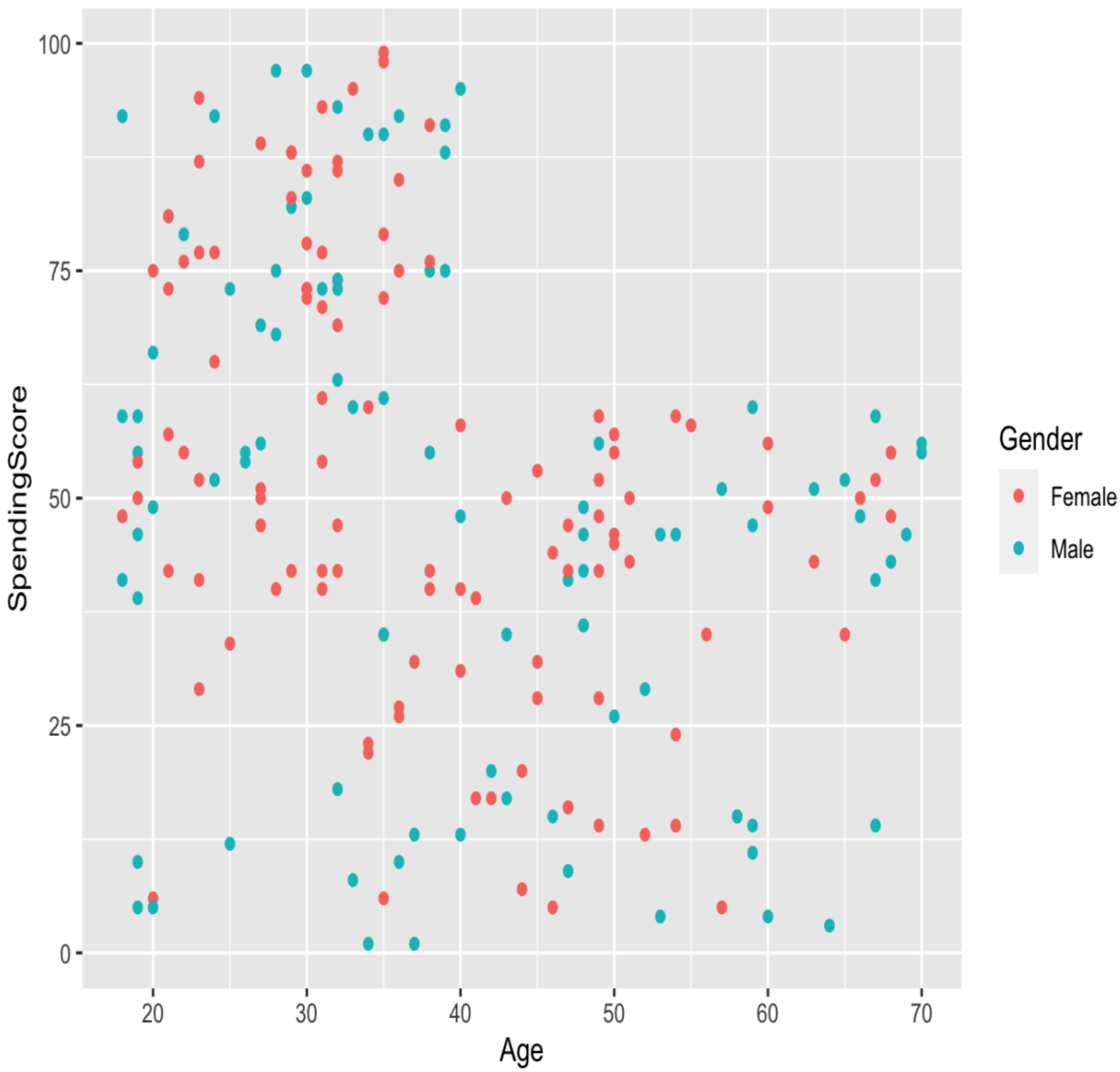
```
##     Gender          Age         AnnualIncome     SpendingScore
##  Female:112   Min.   :18.00   Min.   : 15.00   Min.   : 1.00
##  Male  : 88   1st Qu.:28.75   1st Qu.: 41.50   1st Qu.:34.75
##               Median :36.00   Median : 61.50   Median :50.00
##               Mean   :38.85   Mean   : 60.56   Mean   :50.20
##               3rd Qu.:49.00   3rd Qu.: 78.00   3rd Qu.:73.00
```

```
##                  Max.   :70.00   Max.   :137.00   Max.   :99.00
```

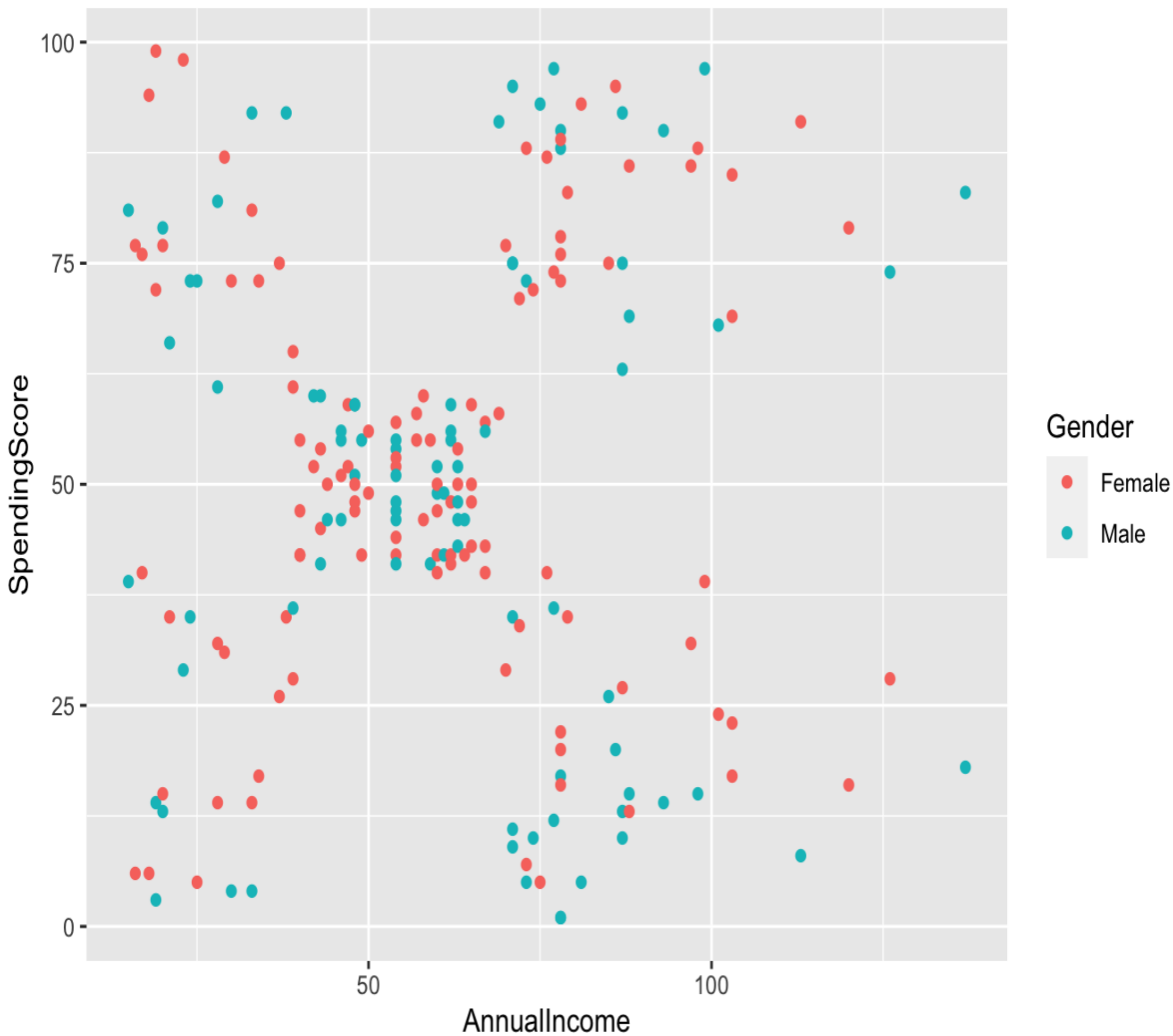
```
# Plot to see relationship among variables
library(ggplot2)
ggplot(customers) +
  geom_point(aes(x = Age, y = AnnualIncome, col = Gender))
```




```
ggplot(customers) +
  geom_point(aes(x = Age, y = SpendingScore, col = Gender))
```




```
ggplot(customers) +
  geom_point(aes(x = AnnualIncome, y = SpendingScore, col = Gender))
```



The first plot shows that the highest income are obtained by people who age from 30 to 50. The second plot demonstrates all the huge spenders are less than 40 years old. Customers above that age have the highest values of Spending Score are around 60 points. The last plot shows that observations tend to classify themselves in a couple of areas on the graph. There is a numerous group right in the middle and a few groups in the corners of the plot. Gender seems to have little effect when income and spending of customers is analysed.

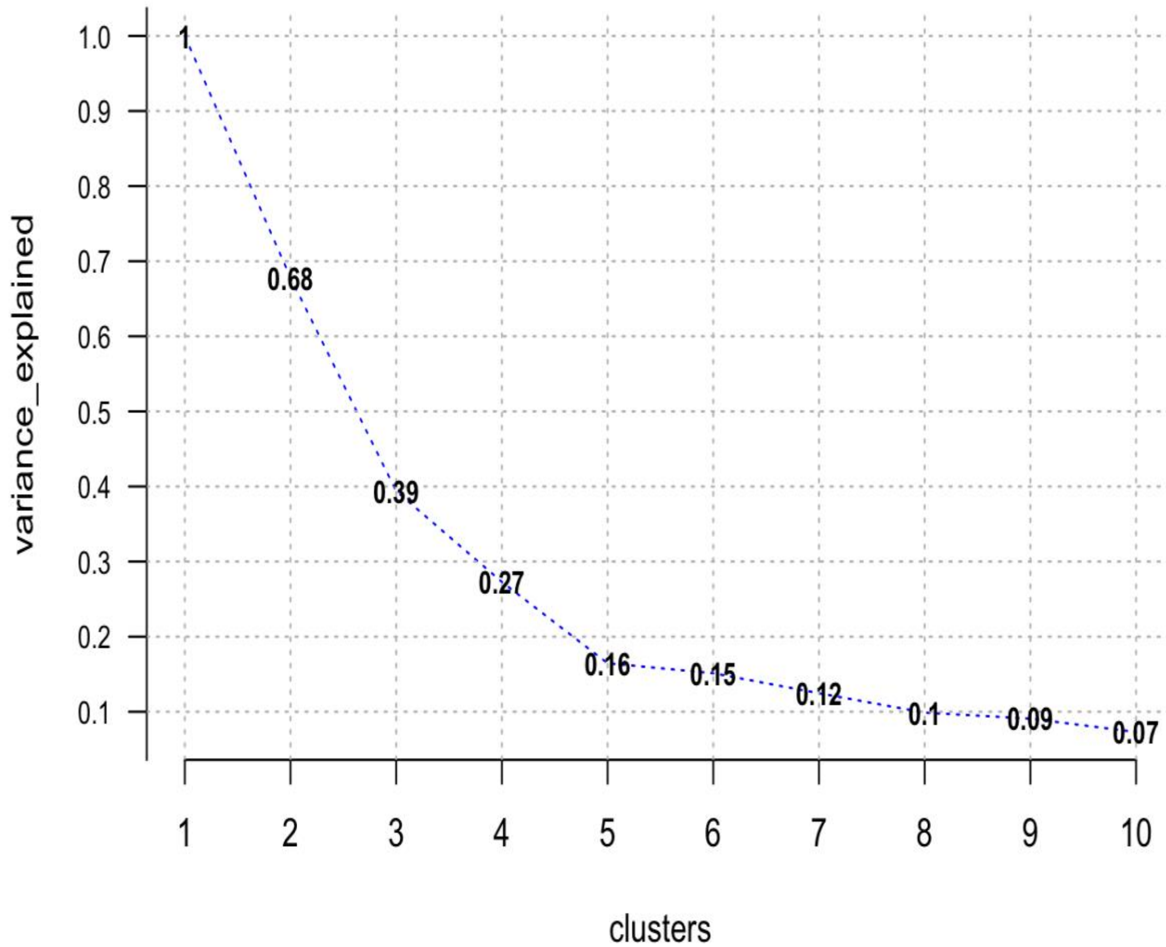
7.1 Perform k-means clustering on the dataset

Based on the above graphs, we found out that two variables: AnnualIncome and SpendingScore are the ones that influence consumer behaviour the most. Therefore the clusters will be generated only on the basis of these two variables.

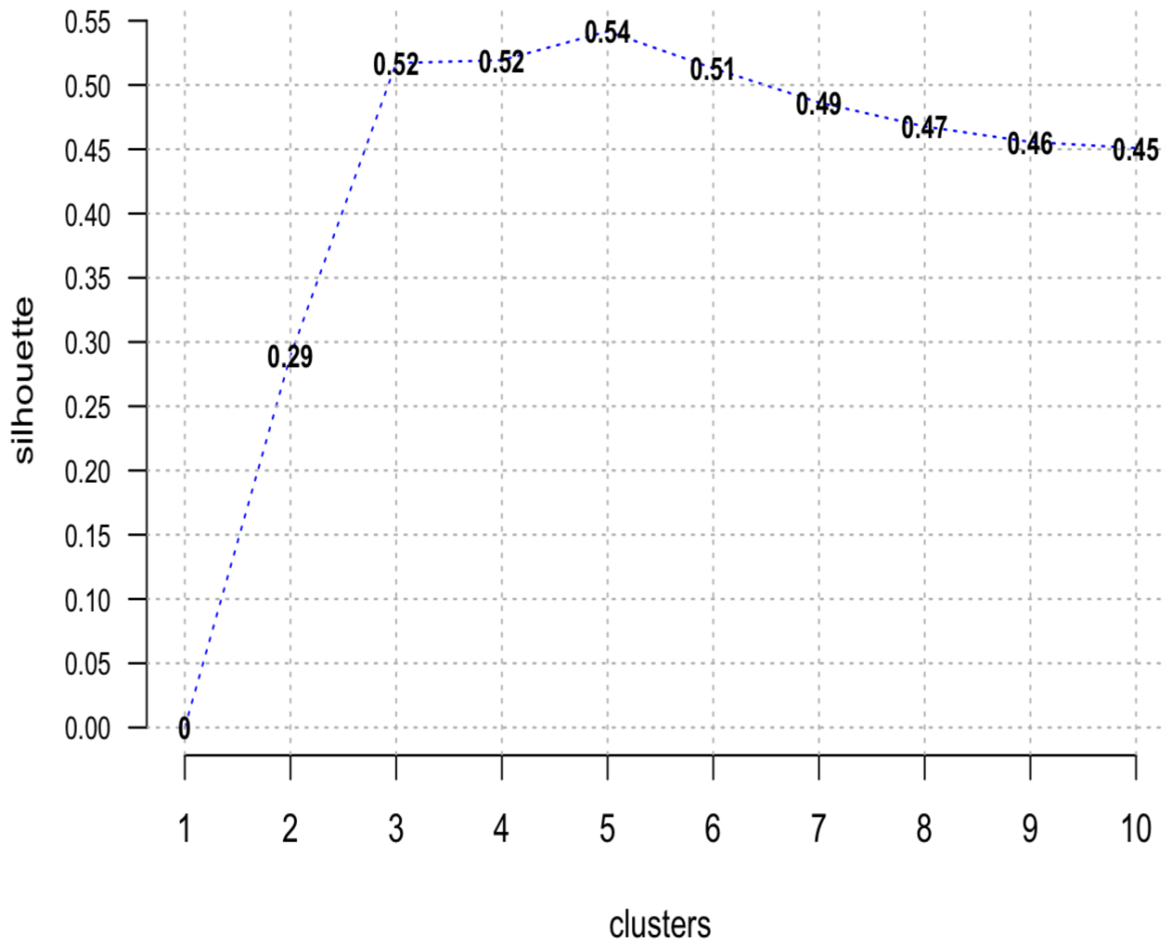
```
# Define the most optimal numbers of clusters
library(ClusterR)
```

```
## Loading required package: gtools

##

## Attaching package: 'gtools'

## The following object is masked from 'package:e1071':

##

##     permutations

## The following object is masked from 'package:car':

##

##     logit
```

```
opt <- Optimal_Clusters_KMeans(customers[, 3:4], max_clusters = 10, plot_cluster
s = T)
```




```
# Use another method to define optimal number of clusters

opt <- Optimal_Clusters_KMeans(customers[, 3:4], max_clusters = 10, plot_cluster
s = T, criterion = 'silhouette')
```



T
he highest average sillhoute value (equal to 0.54) is present for k = 5. Therefore we should opt for 5 clusters in our further analysis with k-means algorithm.

```
set.seed(22)
# Perform k-means clustering on the dataset
km <- kmeans(customers[,3:4], 5)
customers$ClusterNumber <- km$cluster
km
```

```
## K-means clustering with 5 clusters of sizes 35, 81, 23, 22, 39
##
## Cluster means:
```

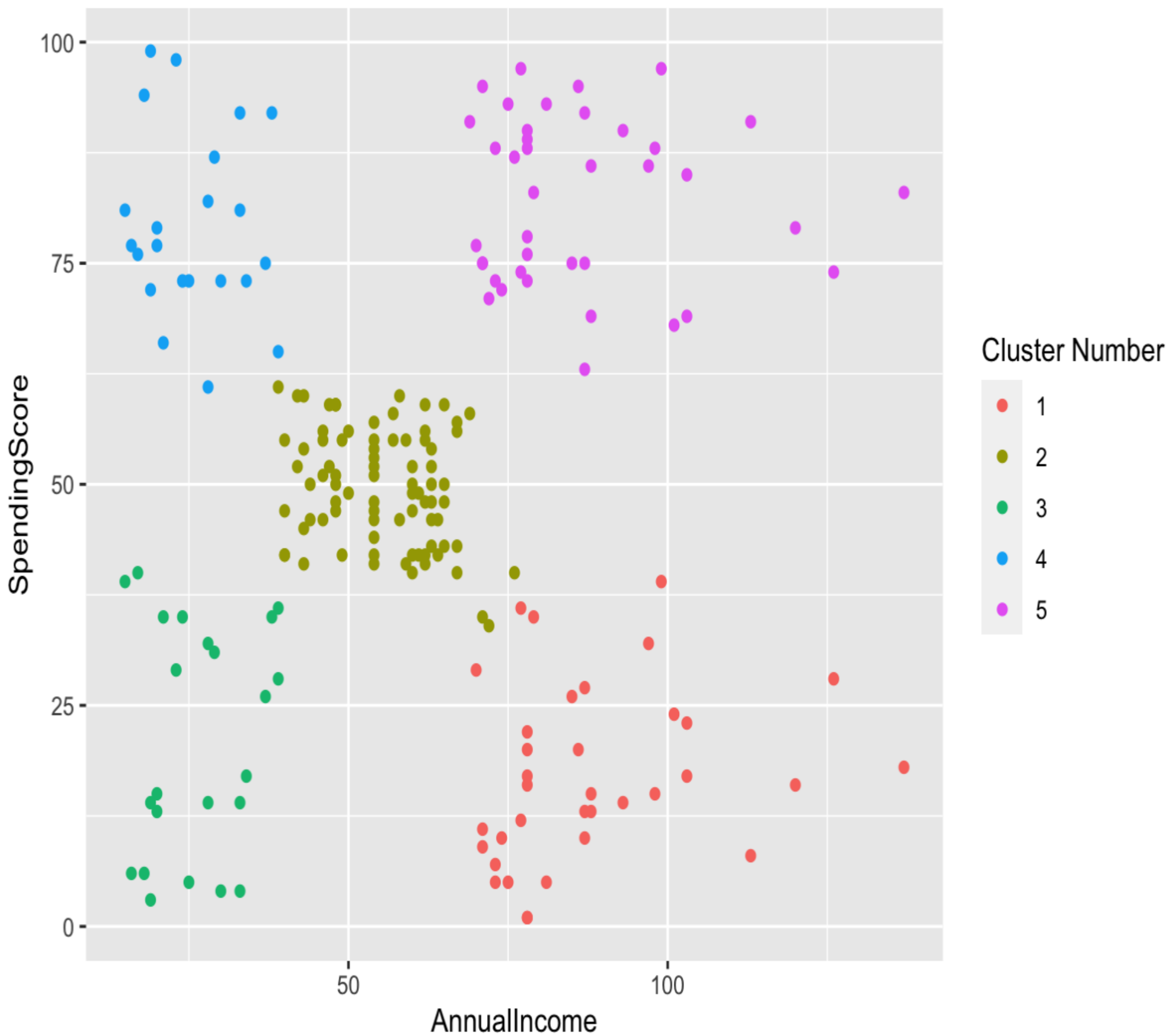
```
##   AnnualIncome SpendingScore
## 1     88.20000      17.11429
## 2     55.29630      49.51852
## 3     26.30435      20.91304
## 4     25.72727      79.36364
## 5     86.53846      82.12821
##
## Clustering vector:
##   [1] 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4
3
##  [38] 4 3 4 3 4 3 2 3 4 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
2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 5 1 5 2 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] 12511.143  9875.111  5098.696  3519.455 13444.051
##  (between_SS / total_SS =  83.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss
"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
summary(km)
```

```
##              Length Class  Mode
## cluster      200    -none- numeric
## centers       10    -none- numeric
## totss          1    -none- numeric
## withinss       5    -none- numeric
## tot.withinss   1    -none- numeric
## betweenss      1    -none- numeric
```

```
## size             5     -none- numeric
## iter             1     -none- numeric
## ifault           1     -none- numeric
```

```r
# Plot your results
ggplot(customers[,3:5])  +
  geom_point(aes(x = AnnualIncome, y = SpendingScore, col = as.factor(ClusterNum
ber))) +
  scale_color_discrete(name="Cluster Number")
```



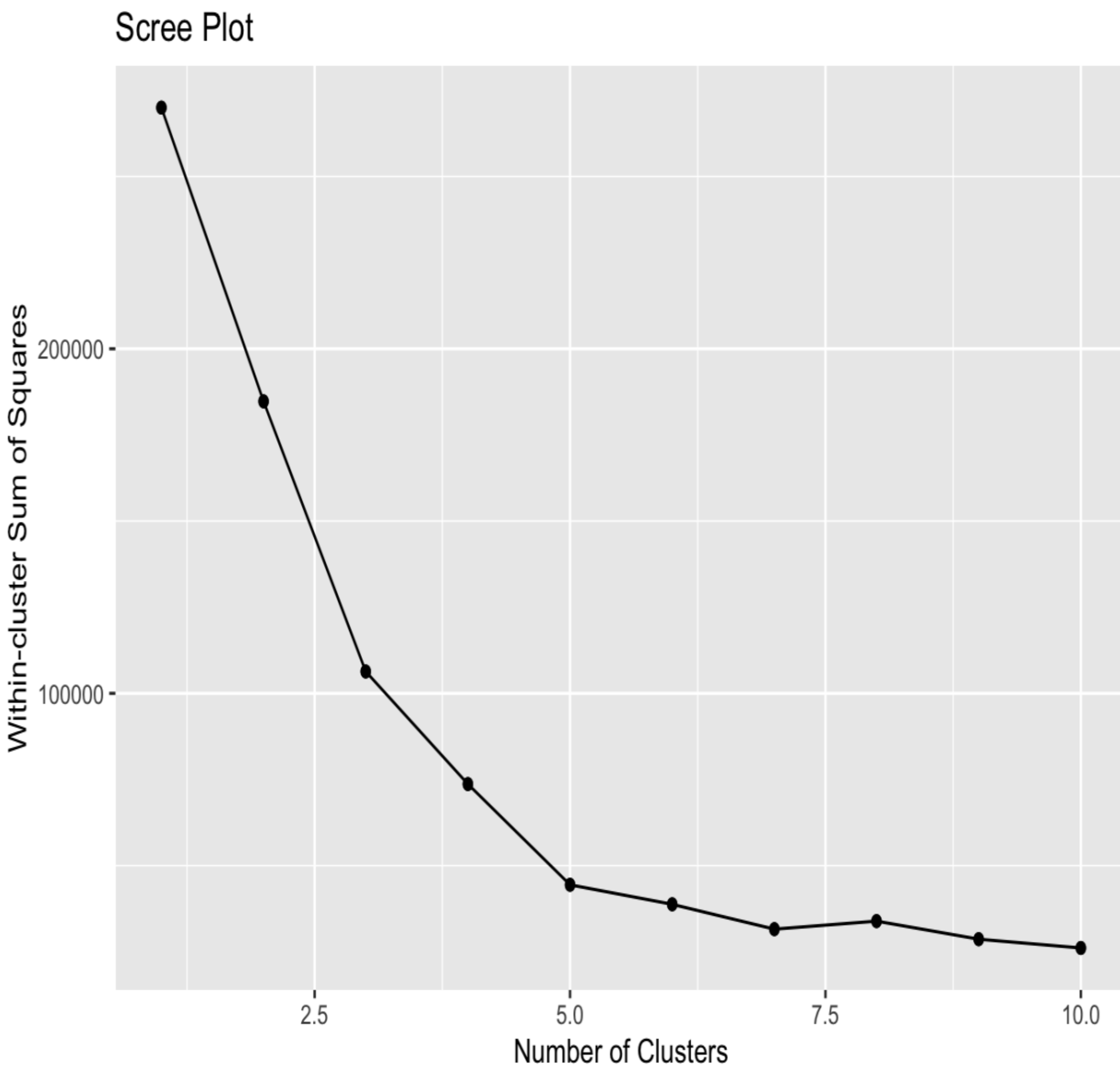
7.2 Repeat the exercise from (1) using different numbers of clusters k between {1, ..., 10}.

For each result, extract the within-cluster sum of squares using …$tot.withinss. Create a scree plot (i.e., plot the sum of squares against the number of clusters) to identify the ideal number of clusters. How many clusters do you suggest we should use to group our customers?

```
set.seed(1)
# Create an empty vector
wss = 0

# Use for loop to aggregate the sum of squares for 1 - 10 cluster centers
for(i in 1:10) {
  km.out <- kmeans(customers[,3:4], centers = i , nstart=1)
    #Save total within sum of squares to wss variable
     wss[i] = km.out$tot.withinss
    # For each clusters k from 1 to 10, extract the within-cluster sum of square
s
    print(wss[i])}
```

```
## [1] 269981.3
## [1] 184740.4
## [1] 106348.4
## [1] 73679.79
## [1] 44448.46
## [1] 38788.46
## [1] 31573.82
## [1] 33908.15
## [1] 28662.93
## [1] 26115.87
```

```
# Plot a scree plot shows the total within sum of squares vs. number of clusters
qplot(1:10, wss) + geom_point() +
  geom_line() +
  xlab("Number of Clusters") +
  ylab("Within-cluster Sum of Squares") +
  ggtitle("Scree Plot")
```




```
# Set k equal to the number of clusters corresponding to the elbow location
k = 5
```

The ideal number of clusters is the one that is located at the elbow location, which is 5.

Same results as we determined the most optimal number of cluster using silhouette method in 7.1, the result here is consistent. Therefore, I would highly suggest using 5 clusters to group customers.

7.3 In order to visualize clusters, we must reduce the dimensionality of the data. Use principal

component analysis to generate two variables out of the four present in the dataset (ignore customer id as a variable). Find a suitable name for the variables you have generated

```r
#Perform Principal Component Analysis
str(customers)
## 'data.frame':    200 obs. of  5 variables:
##  $ Gender       : Factor w/ 2 levels "Female","Male": 2 2 1 1 1 1 1 1 2 1 ...
##  $ Age          : int  19 21 20 23 31 22 35 23 64 30 ...
##  $ AnnualIncome : int  15 15 16 16 17 17 18 18 19 19 ...
##  $ SpendingScore: int  39 81 6 77 40 76 6 94 3 72 ...
##  $ ClusterNumber: int  3 4 3 4 3 4 3 4 3 4 ...
customers$Gender = as.numeric(customers$Gender)
pcclust<-prcomp(customers[, 1:4], scale=FALSE)

#Checking the summary of the PCA model
summary(pcclust)
## Importance of components:
##                            PC1     PC2     PC3     PC4
## Standard deviation     26.4625 26.1597 12.9317 0.49548
## Proportion of Variance  0.4512  0.4409  0.1077 0.00016
## Cumulative Proportion   0.4512  0.8921  0.9998 1.00000
# Applying the PCA model on the data
pcclust$rotation[, 1:2]
##                         PC1          PC2
## Gender         0.0003327282  0.001578712
## Age            0.1889772912  0.130961404
## AnnualIncome  -0.5886227558  0.808388308
## SpendingScore -0.7860093664 -0.573894557
```

Results from the PCA show that components 1 and 2 (PC1 and PC2) contribute the most variance to the data. The high correlation between PC1 and spending score (-0.786) and PC2 and annual income (0.808) show that annual income and spending income are the 2 major components of the data.

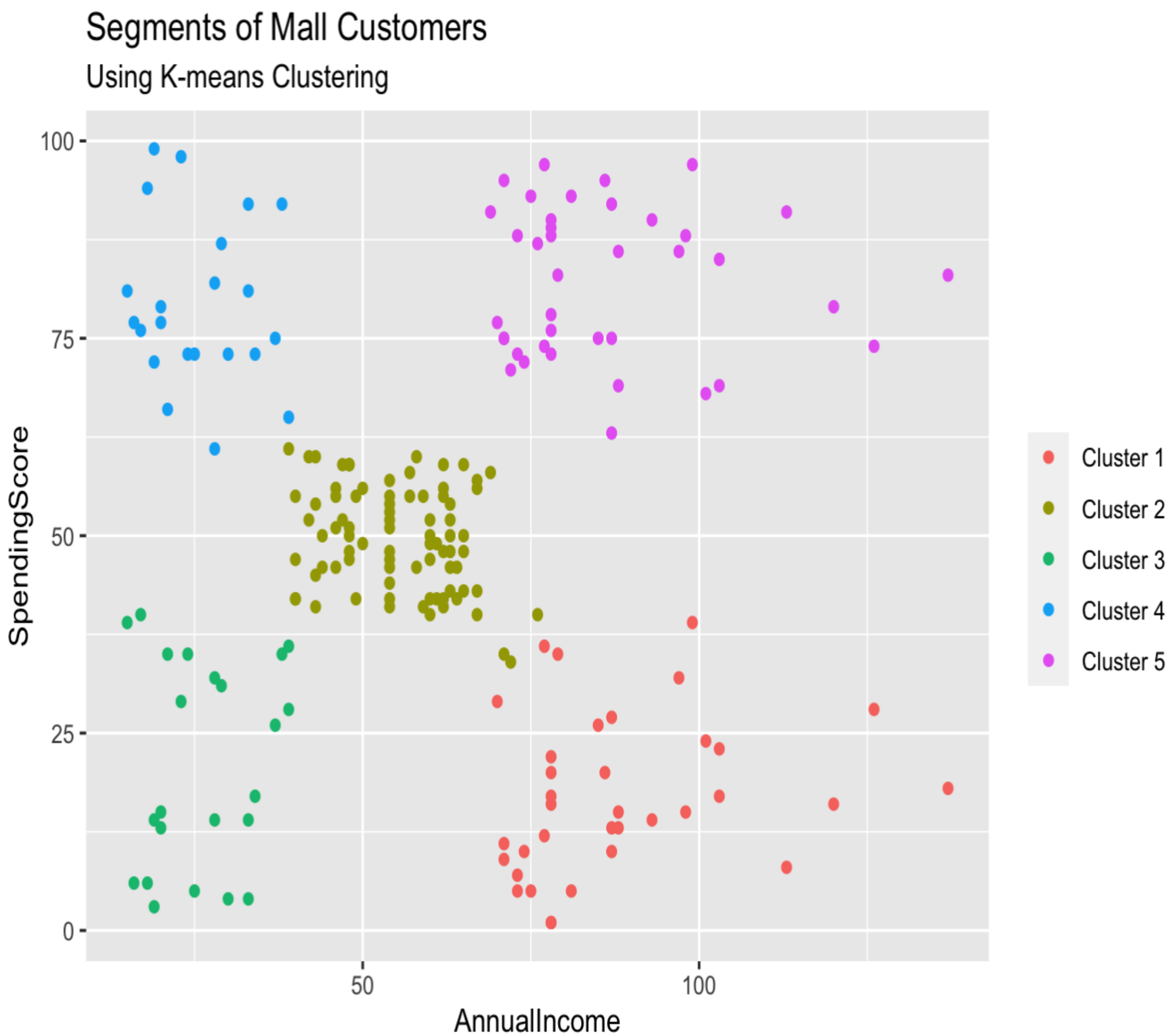
These newly generated variables from PCA have got the new names in 7.1 which are AnnualIncome and SpendingScore.

7.4 Identify the clusters made up of the most valuable consumers.

Plot the customer segments based on results from the cluster analysis and PCA.

```r
# Set seed to 1
```

```r
set.seed(1)

#Create a plot of the customers segments
ggplot(customers, aes(x = AnnualIncome , y = SpendingScore)) +
  geom_point(stat = "identity", aes(color = as.factor(km$cluster))) +
  scale_color_discrete(name = " ",
                       breaks=c("1", "2", "3", "4", "5"),
                       labels=c("Cluster 1", "Cluster 2", "Cluster 3",
                                "Cluster 4", "Cluster 5")) +
  ggtitle("Segments of Mall Customers",
          subtitle = "Using K-means Clustering")
```




```
#Create a more informative plot of the customers segments
library(ggplot2)
ggplot(customers, aes(x = AnnualIncome , y = SpendingScore)) +
  geom_point(stat = "identity", aes(color = as.factor(km$cluster))) +
  scale_color_discrete(name = " ",
                       breaks=c("1", "2", "3", "4", "5"),
                       labels=c("High Income, Low Spending", "Medium Income, Medium Spending", "Low Income, Low Spending", "Low Income, High Spending","High Income, High Spending")) +
  labs(x="Annual Income", y="Spending Score") +
  ggtitle("Segments of Mall X Customers",
```

```
        subtitle = "Using K-means Clustering")
```

As the graph shown, the clusters that made up for the most valuable customers are Cluster 4,5 or the Segments “Low Income, High Spending” and “High Income, High Spending” as High Spending contributes to better revenue and profit for the business.

```
tinytex::install_tinytex()
max_print_line = 10000
```