Customer Segmentation

DATA SCIENCE INTERNSHIP - EXPOSYS DATA LABS

Aarushi Gupta

Table of Contents

Introduction

What is Customer Segmentation?

Customer segmentation is the process of dividing customers into groups based on common characteristics so companies can market to each group effectively and appropriately.

In business-to-business marketing, a company might segment customers according to a wide range of factors, including:

- Industry
- Number of employees
- Products previously purchased from the company
- Location

In business-to-consumer marketing, companies often segment customers according to demographics that include:

- Age
- Gender
- Marital status
- Location (urban, suburban, rural)
- Life stage (single, married, divorced, empty-nester, retired, etc.)

Why Segment Customers?

Segmentation allows marketers to better tailor their marketing efforts to various audience subsets. Those efforts can relate to both communications and product development. Specifically, segmentation helps a company:

- Create and communicate targeted marketing messages that will resonate with specific groups of customers, but not with others (who will receive messages tailored to their needs and interests, instead).
- Select the best communication channel for the segment, which might be email, social media posts, radio advertising, or another approach, depending on the segment.
- Identify ways to improve products or new product or service opportunities.
- Establish better customer relationships.

- Test pricing options.
- Focus on the most profitable customers.
- Improve customer service.
- Upsell and cross-sell other products and services.

How to Segment Customers

Customer segmentation requires a company to gather specific information – data – about customers and analyze it to identify patterns that can be used to create segments.

Some of that can be gathered from purchasing information – job title, geography, products purchased, for example. Some of it might be gleaned from how the customer entered your system. An online marketer working from an opt-in email list might segment marketing messages according to the opt-in offer that attracted the customer, for example. Other information, however, including consumer demographics such as age and marital status, will need to be acquired in other ways.

Typical information-gathering methods include:

- Face-to-face or telephone interviews
- Surveys
- General research using published information about market categories
- Focus groups

Using Customer Segments

Common characteristics in customer segments can guide how a company markets to individual segments and what products or services it promotes to them. A small business selling hand-made guitars, for example, might decide to promote lower-priced products to younger guitarists and higher-priced premium guitars to older musicians based on segment knowledge that tells them that younger musicians have less disposable income than their older counterparts. Similarly, a meals-by-mail service might emphasize convenience to millennial customers and “tastes-like-mother-used-to-make” benefits to baby boomers.

Customer segmentation can be practiced by all businesses regardless of size or industry and whether they sell online or in person. It begins with gathering and analyzing data and ends with acting on the information gathered in a way that is appropriate and effective.

Methodology

Clustering

Clustering is one of the most common exploratory data analysis technique used to get an intuition about the structure of the data. It can be defined as the task of identifying subgroups in the data such that data points in the same subgroup (cluster) are very similar while data points in different clusters are very different. In other words, we try to find homogeneous subgroups within the data such that data points in each cluster are as similar as possible according to a similarity measure such as euclidean-based distance or correlation-based distance. The decision of which similarity measure to use is application-specific.

Clustering analysis can be done on the basis of features where we try to find subgroups of samples based on features or on the basis of samples where we try to find subgroups of features based on samples. We'll cover here clustering based on features. Clustering is used in market segmentation; where we try to find customers that are similar to each other whether in terms of behaviours or attributes, image segmentation/compression; where we try to group similar regions together, document clustering based on topics, etc.

Unlike supervised learning, clustering is considered an unsupervised learning method since we don't have the ground truth to compare the output of the clustering algorithm to the true labels to evaluate its performance. We only want to try to investigate the structure of the data by grouping the data points into distinct subgroups.

In this post, we will cover only **Kmeans** which is considered as one of the most used clustering algorithms due to its simplicity.

Kmeans Algorithm

Kmeans algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to **only one group**. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The way kmeans algorithm works is as follows:

1. Specify number of clusters *K*.
2. Initialize centroids by first shuffling the dataset and then randomly selecting *K* data points for the centroids without replacement.

3. Keep iterating until there is no change to the centroids. i.e. assignment of data points to clusters isn't changing.

- Compute the sum of the squared distance between data points and all centroids.
- Assign each data point to the closest cluster (centroid).
- Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.

Applications

- kmeans algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc. The goal usually when we undergo a cluster analysis is either:
- Get a meaningful intuition of the structure of the data we're dealing with.
- Cluster-then-predict where different models will be built for different subgroups if we believe there is a wide variation in the behaviours of different subgroups. An example of that is clustering patients into different subgroups and build a model for each subgroup to predict the probability of the risk of having heart attack.

Implementation

Including all the needed libraries:

```
library(ggplot2)
library(RColorBrewer)
library(purrr)
library(cluster)
library(gridExtra)
library(grid)
library(NbClust)
library(factoextra)
```

Reading the customer dataset into RStudio:

```
customer_data <- read.csv("Mall_Customers.csv")
```

Data Exploration:

```
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 ...
```

```
names(customer_data)
```

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

```
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
```

```
summary(customer_data)
```

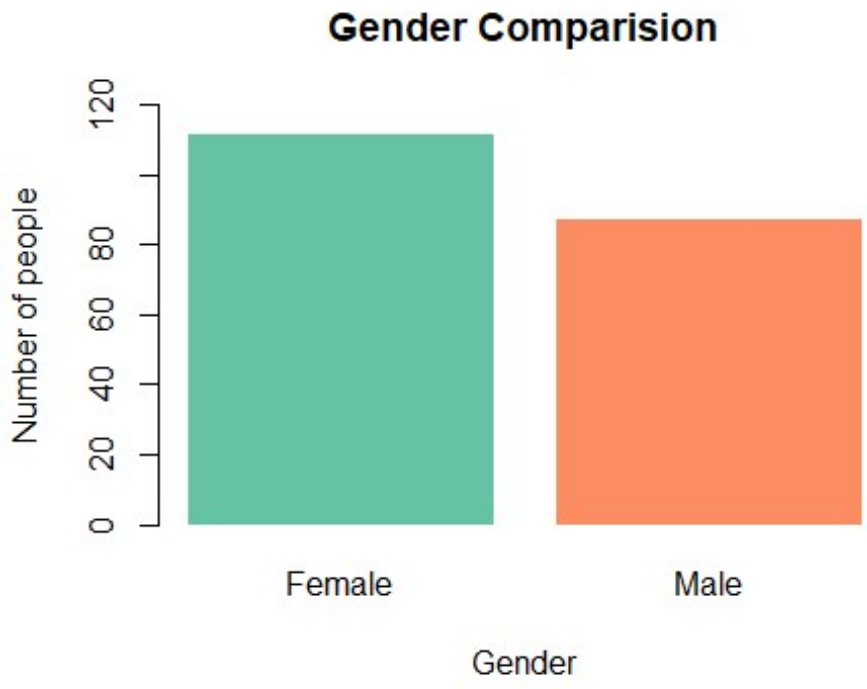
```
##    CustomerID        Gender               Age        Annual.Income..k..
##  Min.   :  1.00   Length:200         Min.   :18.00   Min.   : 15.00
```

```
##  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 Visualization

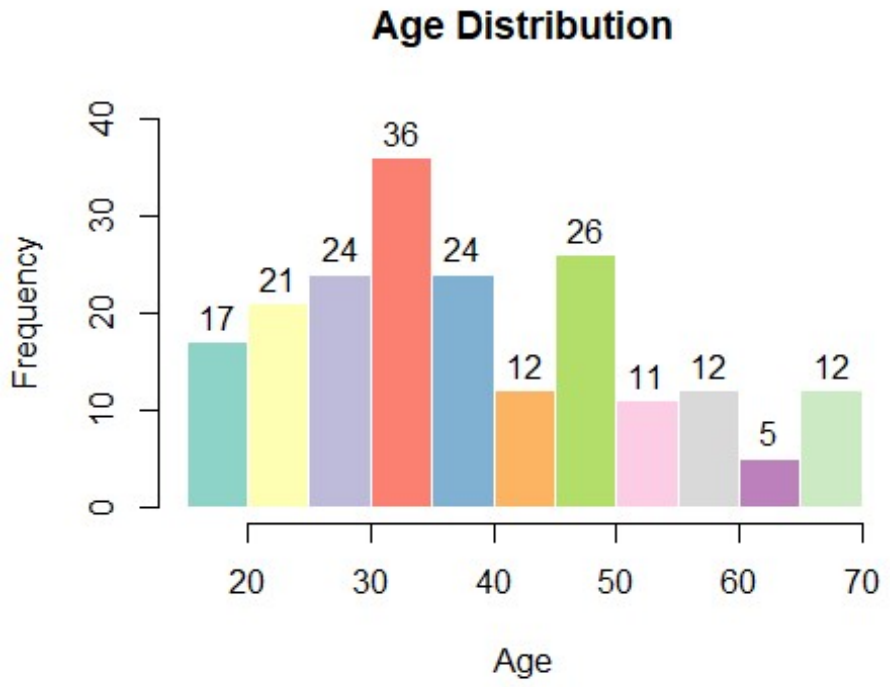
Customer gender distribution:

```
barplot(table(customer_data$Gender), main = "Gender Comparision",
        xlab = "Gender",
        ylab = "Number of people",
        ylim = c(0,120),
        col = brewer.pal(3, "Set2"),
        border = "white")
```



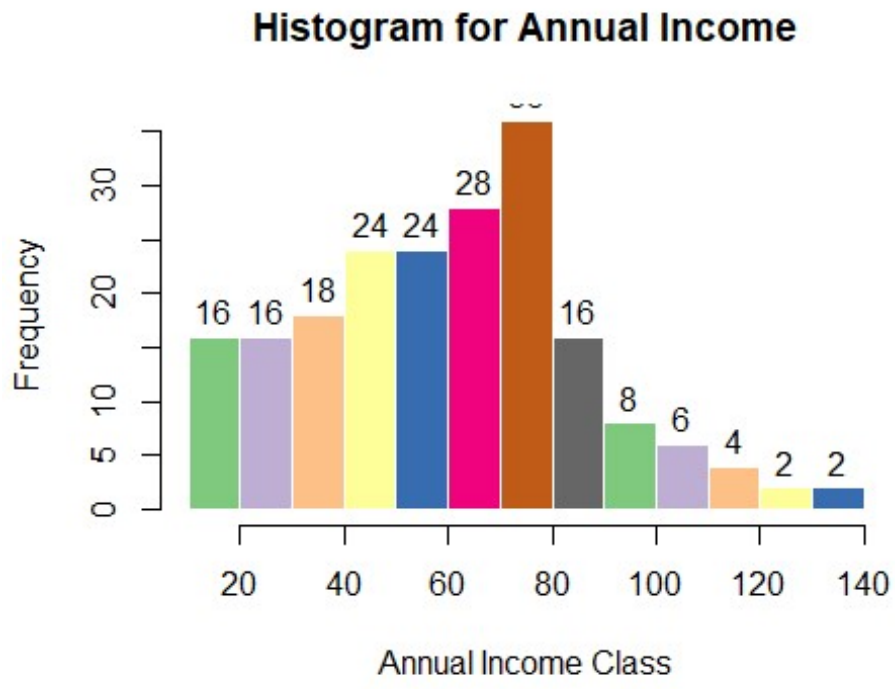
Customer Age Distribution:

```
hist(customer_data$Age, main = "Age Distribution",
     xlab = "Age",
     ylab = "Frequency",
     ylim = c(0,40),
     col = brewer.pal(11, "Set3"),
     border = "white",
     labels = T)
```

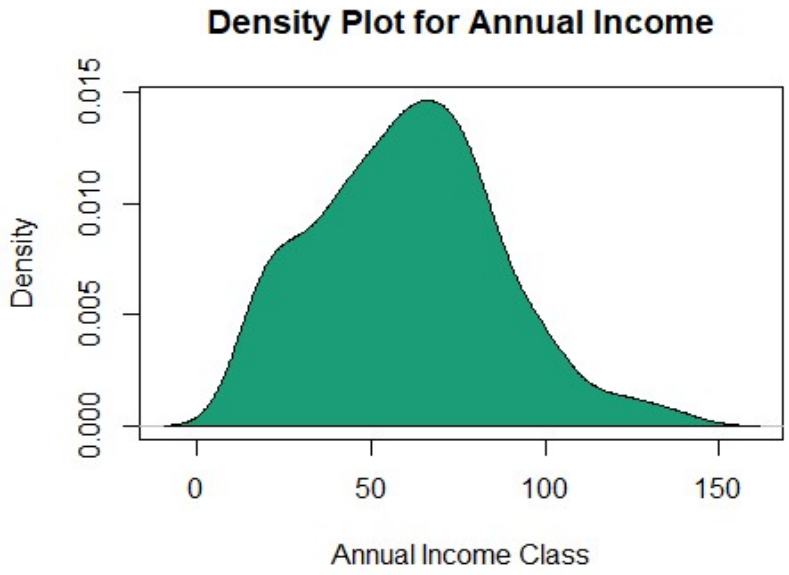


Customer Annual Income Distribution:

```
hist(customer_data$Annual.Income..k..,
  col=brewer.pal(8,"Accent"),
  main="Histogram for Annual Income",
  xlab="Annual Income Class",
  ylab="Frequency",
  border = "white",
  labels=TRUE)
```

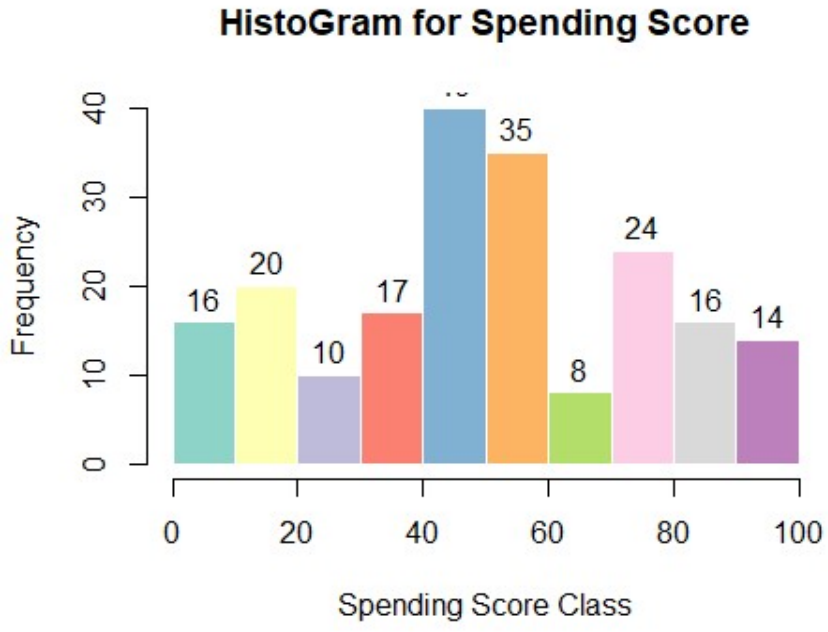



```
plot(density(customer_data$Annual.Income..k..),
    col="yellow",
    main="Density Plot for Annual Income",
    xlab="Annual Income Class",
    ylab="Density")
polygon(density(customer_data$Annual.Income..k..),
        col=brewer.pal(3,"Dark2"))
```



Customer Spending Score Distribution:

```
hist(customer_data$Spending.Score..1.100.,
     main="HistoGram for Spending Score",
     xlab="Spending Score Class",
     ylab="Frequency",
     col=brewer.pal(11,"Set3"),
     border = "white",
     labels=TRUE)
```



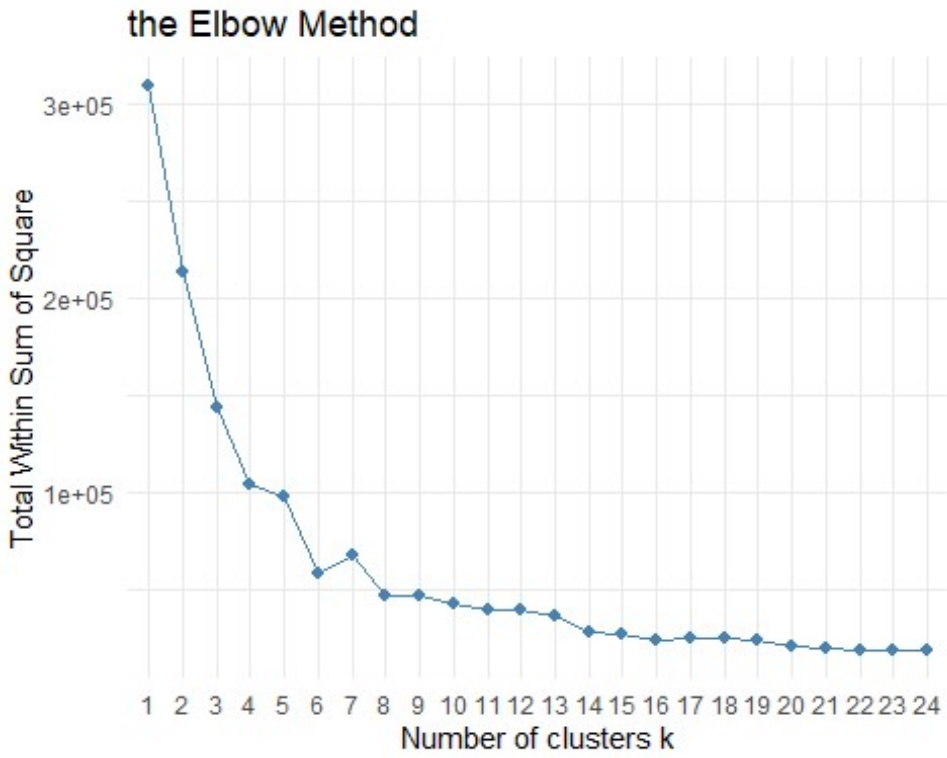
K Means Clustering

Determining optimal number of clusters:

The "Elbow" Method:

Probably the most well-known method, the elbow method, in which the sum of squares at each number of clusters is calculated and graphed, and the user looks for a change of slope from steep to shallow (an elbow) to determine the optimal number of clusters. This method is inexact, but still potentially helpful.

```
fviz_nbclust(customer_data[,3:5], kmeans, method = "wss", k.max = 24) + theme
_minimal() + ggtitle("the Elbow Method")
```



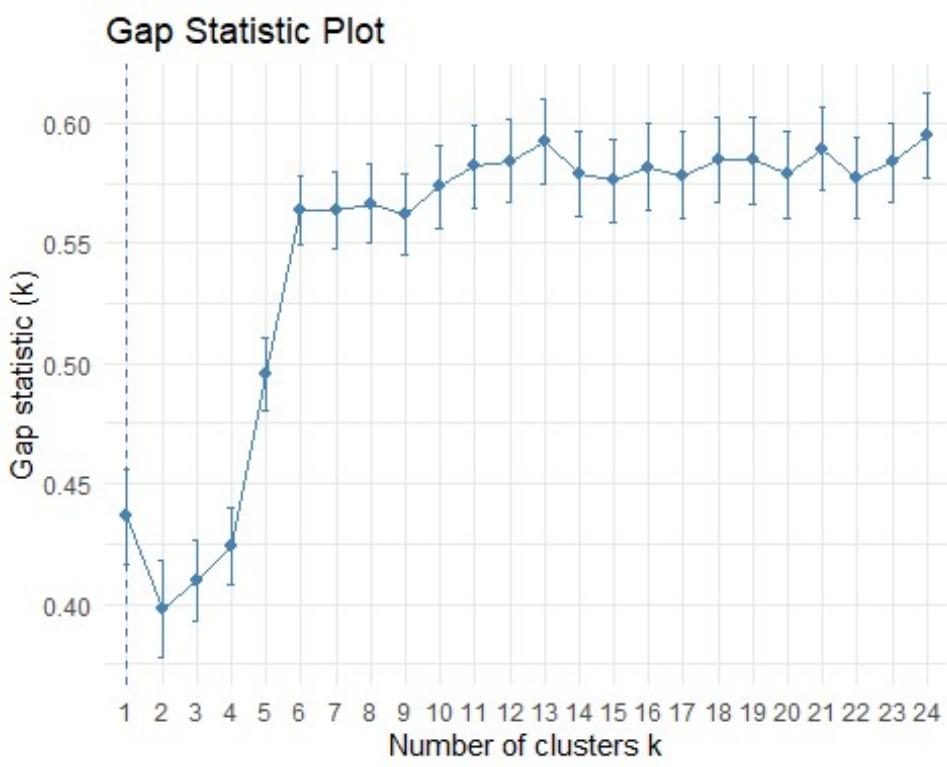
The Elbow Curve method is helpful because it shows how increasing the number of the clusters contribute separating the clusters in a meaningful way, not in a marginal way. The bend indicates that additional clusters beyond the fifth have little value.

The Elbow method is fairly clear, if not a naïve solution based on intra-cluster variance.

The Gap Statistic:

The gap statistic compares the total within intra-cluster variation for different values of k with their expected values under null reference distribution of the data. The estimate of the optimal clusters will be value that maximize the gap statistic (*i.e.,* that yields the largest gap statistic). This means that the clustering structure is far away from the random uniform distribution of points.

```
gap_stat <- clusGap(customer_data[,3:5], FUN = kmeans, nstart = 30, K.max = 2
4, B = 50)
fviz_gap_stat(gap_stat) + theme_minimal() + ggtitle("Gap Statistic Plot")
```

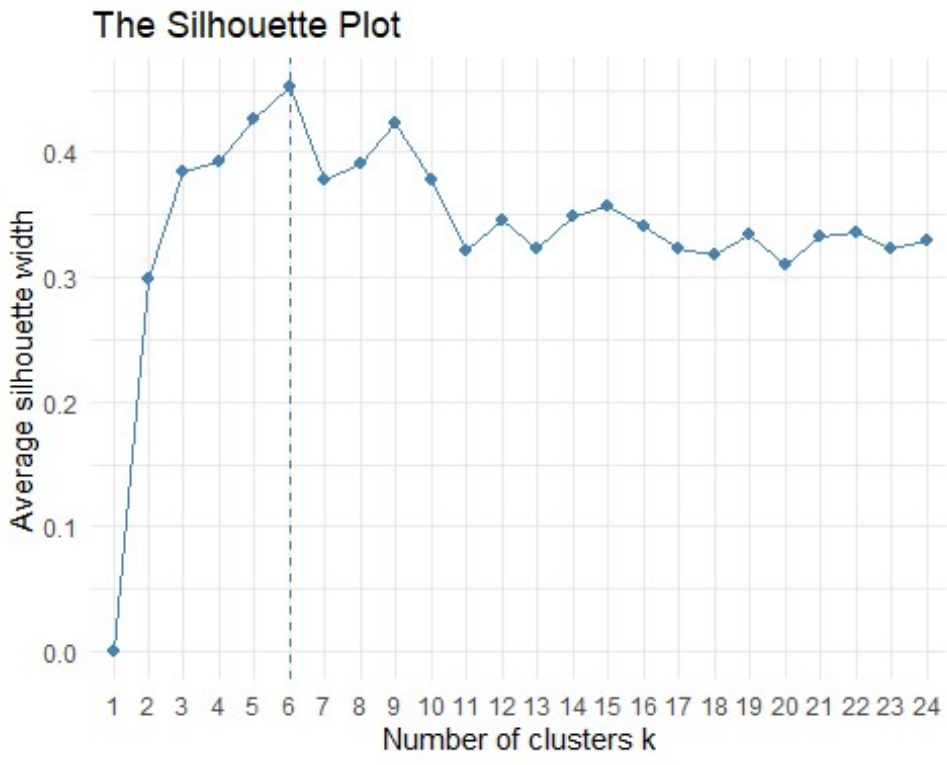


The gap stats plot shows the statistics by number of clusters (k) with standard errors drawn with vertical segments and the optimal value of k marked with a vertical dashed blue line.

The Silhouette Method:

Another visualization that can help determine the optimal number of clusters is called the a silhouette method. Average silhouette method computes the average silhouette of observations for different values of k. The optimal number of clusters k is the one that maximize the average silhouette over a range of possible values for k.

```
fviz_nbclust(customer_data[,3:5], kmeans, method = "silhouette", k.max = 24)
+ theme_minimal() + ggtitle("The Silhouette Plot")
```



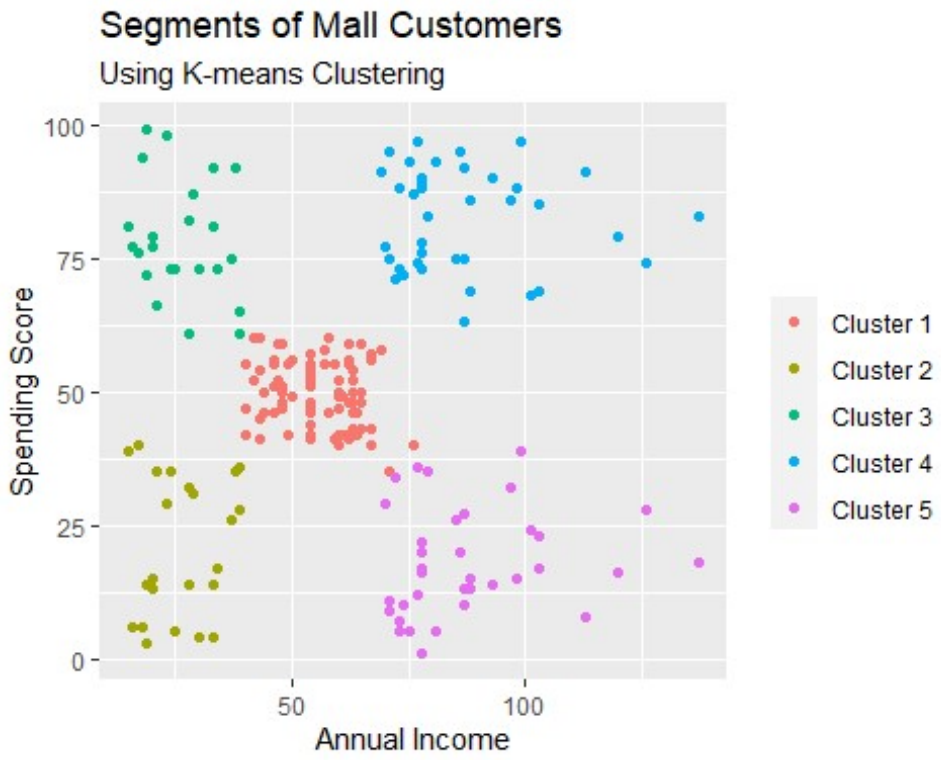
This suggests an optimal of 6 clusters.

Visualizing the Identified Clusters:

Based on the above technique outcomes along with trial and error, it is decided to take 5 clusters for this data set.

```
k5<-kmeans(customer_data[,3:5],5,iter.max=100,nstart=50,algorithm="Lloyd")
k5

## K-means clustering with 5 clusters of sizes 79, 23, 23, 39, 36
##
## Cluster means:
##        Age Annual.Income..k.. Spending.Score..1.100.
## 1 43.08861           55.29114               49.56962
## 2 45.21739           26.30435               20.91304
## 3 25.52174           26.30435               78.56522
## 4 32.69231           86.53846               82.12821
## 5 40.66667           87.75000               17.58333
##
## Clustering vector:
##   [1] 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
2 3 2
##  [38] 3 2 3 2 3 2 3 2 3 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
##  [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 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 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
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] 30138.051  8948.609  4622.261 13972.359 17669.500
##  (between_SS / total_SS =  75.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withi
nss"
## [6] "betweenss"    "size"         "iter"         "ifault"

pcclust=prcomp(customer_data[,3:5],scale=FALSE)
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[,1:3]

##                                PC1        PC2         PC3
## Age                         0.1889742 -0.1309652 0.973209570
## Annual.Income..k..         -0.5886410 -0.8083757 0.005516668
## Spending.Score..1.100. -0.7859965  0.5739136 0.229853647

set.seed(1)
ggplot(customer_data, aes(x =Annual.Income..k.., y = Spending.Score..1.100.))
+
  geom_point(stat = "identity", aes(color = as.factor(k5$cluster))) +
  scale_color_discrete(name=" ",
              breaks=c("1", "2", "3", "4", "5"),
              labels=c("Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4", "C
luster 5")) +
  ggtitle("Segments of Mall Customers", subtitle = "Using K-means Clustering"
) + labs(x = "Annual Income", y = "Spending Score")
```



Cluster 1 – This cluster consists of customers with a medium annual income and a medium spending score.

Cluster 2 – This cluster consists of customers with a low annual income and a low spending score.

Cluster 3 – This cluster consists of customers with a low annual income and a high spending score.

Cluster 4 – This cluster consists of customers with a high annual income and a high spending score.

Cluster 5 – This cluster consists of customers with a high income and a low spending score.

Conclusion

The customers were divided into 5 groups as listed below.

CLUSTER	ANNUAL INCOME	SPENDING SCORE	SIZE
CLUSTER 1	**Medium**	**Medium**	**79**
CLUSTER 2	**Low**	**Low**	**23**
CLUSTER 3	**Low**	**High**	**23**
CLUSTER 4	**High**	**High**	**29**
CLUSTER 5	**High**	**Low**	**36**

The first group is the largest, having customers with medium annual income and medium spending score. The store must concentrate their business strategies keeping it around this group to gain maximum profits.

The second and third groups consisting of low income customers form the most minor part of the stores customers set and so they need not be given the most importance.