Customer Segmentation and Profiling

Rushikesh Naidu

March 15, 2018

### Introduction

#### The aim of the assignment is to segment customer based on their spendings in different types of goods. The dataset provides division of customers based on region and channel. Information about the dataset can be found on the Readme document.

##### The dataset is not clean with some missing values. We clean the dataset first followed by removing the Channel and Region from our cleaned dataset since it is not required to cluster the data.

MyData <- read.csv(file="C:/Users/Admin/Desktop/MIS584/Wholesale customers data-1.csv", header=TRUE, sep=",")  
head(MyData)

## ï..Channel Region Fresh Milk Grocery Frozen Detergents\_Paper Delicassen  
## 1 2 3 12669 9656 7561 214 2674 1338  
## 2 2 3 7057 9810 9568 1762 3293 1776  
## 3 2 3 6353 8808 7684 2405 3516 7844  
## 4 1 3 13265 1196 4221 6404 507 1788  
## 5 2 3 22615 5410 7198 3915 1777 5185  
## 6 2 3 9413 8259 5126 666 1795 1451

summary(MyData)

## ï..Channel Region Fresh Milk   
## Min. :1.000 Min. :1.000 Min. : 3 Min. : 55   
## 1st Qu.:1.000 1st Qu.:2.000 1st Qu.: 3128 1st Qu.: 1533   
## Median :1.000 Median :3.000 Median : 8504 Median : 3627   
## Mean :1.323 Mean :2.543 Mean : 12000 Mean : 5796   
## 3rd Qu.:2.000 3rd Qu.:3.000 3rd Qu.: 16934 3rd Qu.: 7190   
## Max. :2.000 Max. :3.000 Max. :112151 Max. :73498   
## Grocery Frozen Detergents\_Paper Delicassen   
## Min. : 3 Min. : 25.0 Min. : 3.0 Min. : 3.0   
## 1st Qu.: 2153 1st Qu.: 742.2 1st Qu.: 256.8 1st Qu.: 408.2   
## Median : 4756 Median : 1526.0 Median : 816.5 Median : 965.5   
## Mean : 7951 Mean : 3071.9 Mean : 2881.5 Mean : 1524.9   
## 3rd Qu.:10656 3rd Qu.: 3554.2 3rd Qu.: 3922.0 3rd Qu.: 1820.2   
## Max. :92780 Max. :60869.0 Max. :40827.0 Max. :47943.0

cleandata <- MyData  
cleandata <- na.omit(cleandata)  
cleandata$Channel <- NULL  
cleandata$Region <- NULL

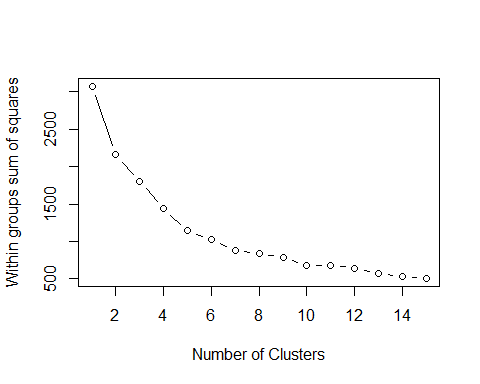
### Task

#### 1.Segmentation: You will first use the data related to spending on the six products (columns c - h) to segment (cluster) the clients. You can simply use Tableau (and Excel if needed) to perform the segmentation process; however, if you are so inclined, you can use other tools and languages such as R or Python to implement a clustering algorithm and answer the questions. If you use Tableau, you can use the automatically determined number of clusters by Tableau (or you can manually change the number of clusters if you have a reason for that). Then, answer the following questions:

##### a. How many segments are created?

###### The within group sum of squares does not show any specific drop after 6 clusters due to which we select the number of clusters in the dataset as 6. We will see in the next plot that this assignment answer 66% of variability of the dataset.

cleandata <- scale(cleandata)  
within\_sum\_of\_squares <- (nrow(cleandata)-1)\*sum(apply(cleandata,2,var))  
for (i in 2:15) within\_sum\_of\_squares[i] <- sum(kmeans(cleandata,   
 centers=i)$withinss)  
plot(1:15,within\_sum\_of\_squares, type="b", xlab="Number of Clusters",  
 ylab="Within groups sum of squares")

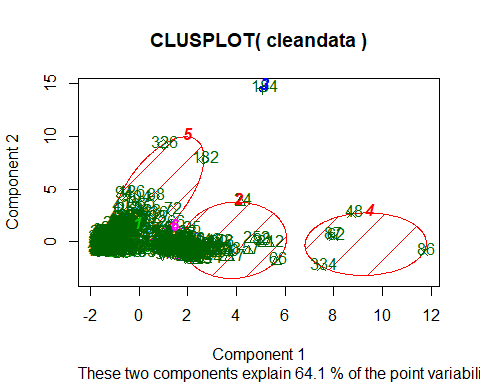


#This does not give a strong elbow, but intuition shows that there is no significant drop after 6 clusters.   
fit <- kmeans(cleandata, 6) # fit the model  
aggregate(cleandata,by=list(fit$cluster),FUN=mean) # get cluster means

## Group.1 ï..Channel Fresh Milk Grocery Frozen  
## 1 1 -0.6895122 -0.1975378 -0.39060871 -0.4634379 -0.13297154  
## 2 2 1.4470045 -0.4711133 1.44151485 1.7909649 -0.25838840  
## 3 3 -0.6895122 1.9645810 5.16961846 1.2857533 6.89275382  
## 4 4 1.4470045 1.0755395 5.10330749 5.6319063 -0.08979632  
## 5 5 -0.6440544 1.7608360 -0.05218697 -0.2255801 1.48397129  
## 6 6 1.4470045 -0.2332161 0.16639766 0.3152949 -0.33590095  
## Detergents\_Paper Delicassen  
## 1 -0.4402180 -0.19371497  
## 2 1.8264760 0.24070203  
## 3 -0.5542311 16.45971129  
## 4 5.6823687 0.41981740  
## 5 -0.4292139 0.45423892  
## 6 0.3558642 -0.00193479

cleandata <- data.frame(cleandata, fit$cluster) #append cluster assignment

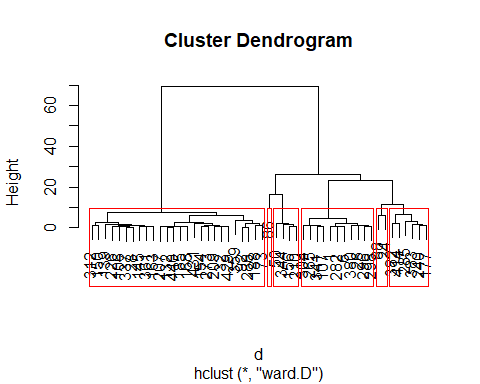
library(cluster)  
clusplot(cleandata, fit$cluster, color=TRUE, shade=TRUE, labels=2, lines=0)



sampled <- sample(1:dim(cleandata)[1], 50)  
Sample\_final <- cleandata[sampled,]  
Sample\_final$Region <- NULL  
Sample\_final$Channel <- NULL  
d <- dist(Sample\_final, method = "euclidean") # distance matrix  
fit <- hclust(d, method="ward")

## The "ward" method has been renamed to "ward.D"; note new "ward.D2"

plot(fit) # display dendogram  
groups <- cutree(fit, k=6) # cut tree into 6 clusters  
rect.hclust(fit, k=6, border="red") # draw dendogram with red borders around the 6 clusters



###### We can see that the cluster explain 66% of the variability in the dataset. The dendograms generated show our division of datasets in 6 different clusters. The rest of the assignment is carried out under tableau.