**K Means Clustering: An Illustration**

**Data: i)** OfferInformation.csv – contains info about 32 distinct offers made to customers. ii) Transactions.csv – contains customer last name and offer #(s) purchased by each customer.

**Requirement:** To segment the customers based on the offers bought by them.

**R Script with Explanation and pictorial outputs:**

# 1. Set the folder path and read in the data

offers<-read.csv(file="OfferInformation.csv")

transactions<-read.csv(file="Transactions.csv")

head(offers)

head(transactions)

# Distinct customer count=100

# 2. Combine both tables in a format that can be used for clustering: Since customers are to be segmented, make # customer name as first column, and all offers as column 2:33.

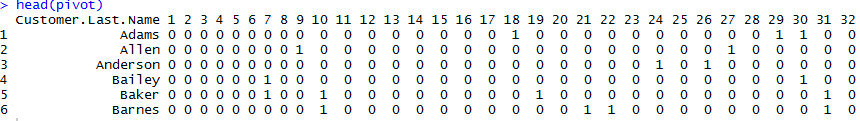
library(reshape)

# Melt transactions, cast customers by offers

pivot<-melt(transactions[1:2])

pivot<-(cast(pivot,Customer.Last.Name~value,fill=0,fun.aggregate=length))

head(pivot)



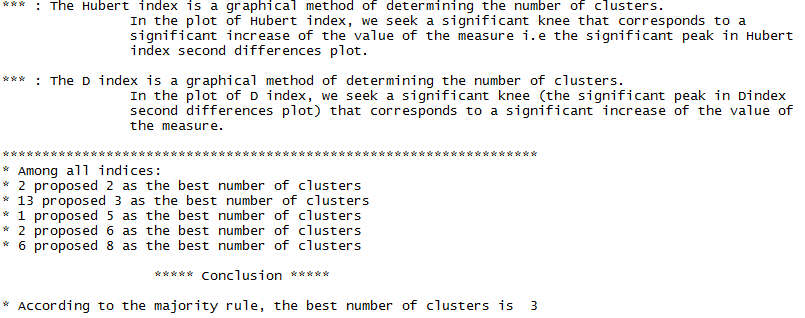
# Note: Only columns 2 to 33 are passed into the kmeans function.

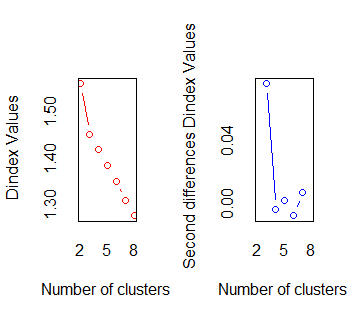
# 3. Determining optimal number of clusters: This can be selected subjectively based on business context. There are some statistical tools also which can help us determine it. Some of these tools can be subjective.

**Method A.** using NbClust package. It creates multiple indices and the value of K favored by maximum indices is the output.

library(NbClust)

temp<-NbClust(data=pivot[-1],min.nc=2,max.nc = 8,method = "kmeans")



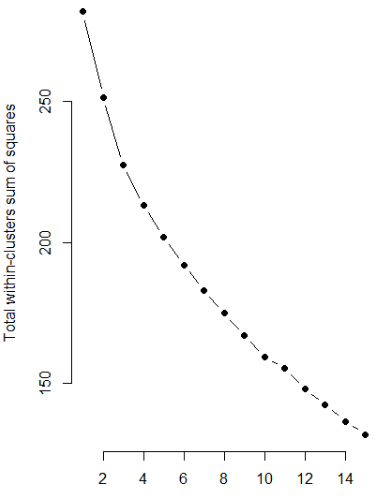


**Method B. Elbow Method:** Calculate WSS(within cluster sum of squares) for a range of K values. Plot it. The preferred K value is the elbow value, after which the change is WSS is less or not significant. This is a subjective decision.

k.max<-15data<-pivot[-1]

wss<-sapply(1:k.max,function(k){kmeans(data, k, nstart=10 )$tot.withinss})

plot(1:k.max, wss, type="b", pch = 19, frame = FALSE, xlab="Number of clusters K", ylab="Total within-clusters sum of squares")



# Here, we may take K=3 or 4. Lets go with K=3.

# 4. Creating Clusters: there are many variations of kmeans but we use vanilla implementation of K means here.

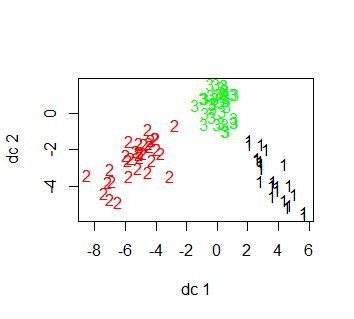
library(stats) ; library(fpc) ; library(reshape)

set.seed(101)

txn.cluster<-kmeans(pivot[-1],3)

txn.cluster$size

plotcluster(pivot,txn.cluster$cluster) # Plot the clusters



# 5. Assigning cluster # to each txn in original dataset.

cust.cluster<-cbind(pivot[1],txn.cluster$cluster)

cluster.offers<-merge(transactions,cust.cluster,by="Customer.Last.Name")

colnames(cluster.offers)<-c("CustomerLastName","Offer","Cluster")

# 6. Exploring the clusters: finding top offers within each cluster. Write the output in excel.

clustertopdeals<-melt(cluster.offers,id=c("Offer","Cluster"))

clustertopdeals<-cast(clustertopdeals,Offer~Cluster,fun.aggregate = length)

head(clustertopdeals)

clustertopdeals<-cbind(offers,clustertopdeals[-1])

write.csv(file="topdeals.csv",clustertopdeals,row.names=F)

# 7. **Inference:** Sort each cluster by descending order of number of customers who purchased a particular offer.

On examining the excel:

**Cluster 1**(size=26): 16/26 customers each for offers 29(Pinot Grigio) and Offer30(Malbec), followed by 12/26 customers each for Offer18&8(Espumant). So we can conclude Cluster 1 customers have high inclination for products Pinot Grigio and Malbec.

**Cluster 2**(size=32): 21 customers for offer22(champagne) and 17 customers for offer31(champagne). We can safely conclude these customers prefer champagne.

**Cluster 3**(size=42): 12 customer bought Offer24(pinot noir), 14 customers for Offer26(Pinot noir). Thus, these customers can be targeted for selling pinot noir.

# 8. Next Steps: We can always run the whole process for 4 clusters or any other value f K and check if we get more distinct cluster definitions or not.