MAS8404 Practical Report 2

Mansoor Hamid

11/9/2019

### Question 1:

#### Part(a)

genexpr\_data=read.csv("C:/Users/User/Documents/RWorkSpace/data/Ch10Ex11.csv",header=TRUE)  
#Check dimesion of the loaded data  
dim(genexpr\_data)

## [1] 999 40

#transpose once to correct the data  
genexpr\_data = t(genexpr\_data)

##### Creating correlation distance matrix

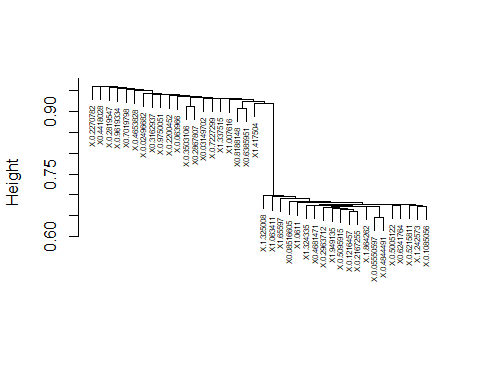
#Transpose the matrix in order to use cor function  
genexpr\_data\_transposed = t(genexpr\_data)  
  
#Compute the inter-observation correlations  
r\_obs = cor(genexpr\_data\_transposed)  
#Create the distance matrix  
d\_cor = 1-r\_obs  
  
#convert the distance matrix to dis object  
d\_cor = as.dist(d\_cor)

##### Perform hierarchical clustering using hclust with metod specified as “single” to indicate single-linkage

hc\_c = hclust(d\_cor, method = "single")

##### Plot the dendrogram

plot(hc\_c, cex=0.5,main="",sub="",xlab="")



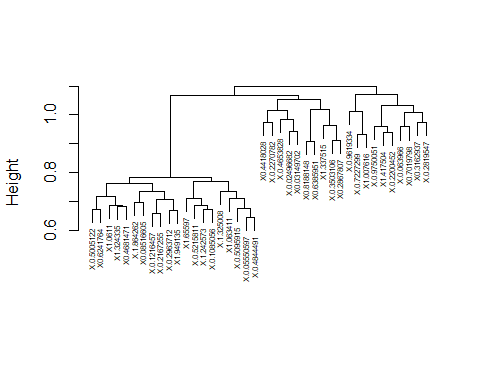
#### Part(b)

##### Perform hierarchical clustering using hclust with metod specified as “complete” and “average”

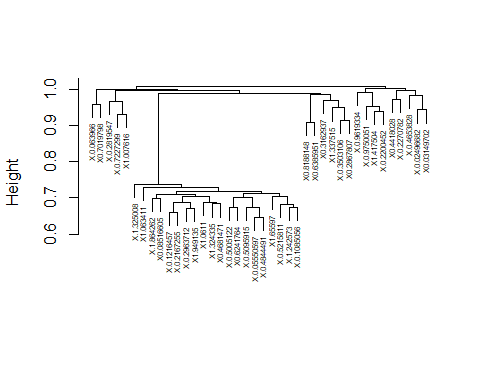
hc\_c\_complete = hclust(d\_cor, method = "complete")  
hc\_c\_average = hclust(d\_cor, method = "average")

##### Plot the dendrograms

plot(hc\_c\_complete, cex=0.5,main="",sub="",xlab="")



plot(hc\_c\_average, cex=0.5,main="",sub="",xlab="")



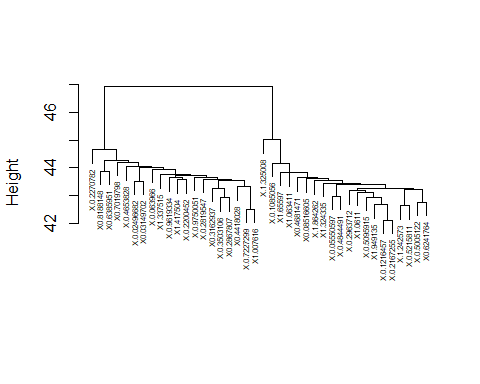
#### Part(c)

###### Determine Euclidean distance and perform hierarchical clustering with single-linkage

genexpr\_data\_euc = dist(genexpr\_data)  
hc\_c\_euc\_single = hclust(genexpr\_data\_euc, method = "single")

##### Plot the dendrogram

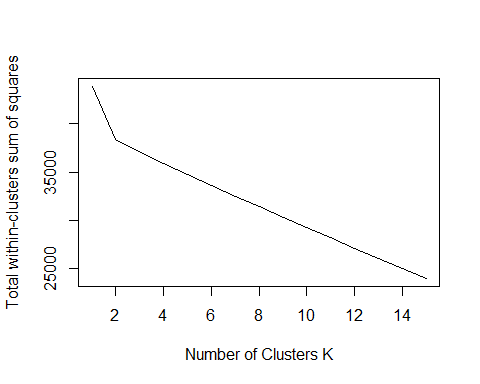
plot(hc\_c\_euc\_single, cex=0.5,main="",sub="",xlab="")



### Question 2:

#### Part(a)

Kmax = 15  
#store with-cluster variation  
SS\_W = numeric(Kmax)  
  
genexpr\_km\_fit = list()  
  
for(K in 1:Kmax){  
 genexpr\_km\_fit[[K]] = kmeans(genexpr\_data,K,iter.max = 50,nstart=20)  
 SS\_W[K] = genexpr\_km\_fit[[K]]$tot.withinss  
}  
plot(1:Kmax,SS\_W, type="l",xlab="Number of Clusters K",ylab = "Total within-clusters sum of squares")



sort(genexpr\_km\_fit[[2]]$cluster)

## X.0.9619334 X0.4418028 X.0.9750051 X1.417504 X0.8188148   
## 1 1 1 1 1   
## X0.3162937 X.0.02496682 X.0.063966 X0.03149702 X.0.3503106   
## 1 1 1 1 1   
## X.0.7227299 X.0.2819547 X1.337515 X0.7019798 X1.007616   
## 1 1 1 1 1   
## X.0.4653828 X0.6385951 X0.2867807 X.0.2270782 X.0.2200452   
## 1 1 1 1 1   
## X.1.242573 X.0.1085056 X.1.864262 X.0.5005122 X.1.325008   
## 2 2 2 2 2   
## X1.063411 X.0.2963712 X.0.1216457 X0.08516605 X0.6241764   
## 2 2 2 2 2   
## X.0.5095915 X.0.2167255 X.0.05550597 X.0.4844491 X.0.5215811   
## 2 2 2 2 2   
## X1.949135 X1.324335 X0.4681471 X1.0611 X1.65597   
## 2 2 2 2 2

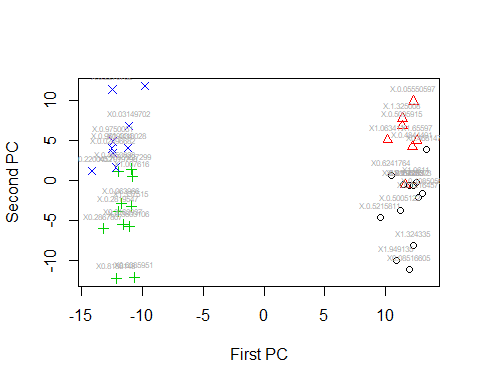
##### Conclusion:

The result suggests that 2 is the optimal number of clusters as there is a bend in the knee at K=2

#### Part(b)

Perfom K-Means clustering with K=4:

genexpr\_km = kmeans(genexpr\_data,4,iter.max = 50,nstart = 20)  
#Perform the PCA  
genexpr\_pca = prcomp(x=genexpr\_data)  
#Plot the first PC against the second PC  
plot(genexpr\_pca$x[,1],genexpr\_pca$x[,2], xlab = "First PC", ylab = "Second PC", col=genexpr\_km$cluster, pch=genexpr\_km$cluster)  
text(genexpr\_pca$x[,1],genexpr\_pca$x[,2],labels = rownames(genexpr\_data),cex=0.5,pos=3,col = "darkgrey")



##### Conclusion:

Overall, clusters are separated