Block 3: Clustering

Huizhi Lin (88112518130) & Jan Orsel (950608630010)

November 14, 2017

First, we read the data.

skin.df <- read.table('get\_normal\_vs\_tumor2\_RAW\_Skin.out',sep=' ',header=TRUE,stringsAsFactors=FALSE)

Then we created the transposed dataset for further analsys.

skin.tdf <- data.frame(t(skin.df[,-2562]))  
colnames(skin.tdf)<-paste0(skin.df$tissue,1:72)

## Task 1: Clustering the genes

We used the tranposed data frame ### K-means clustering First we tried k-mean clustering with k=2

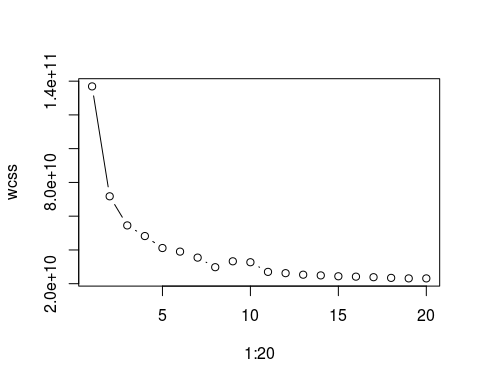
dim(skin.tdf)

## [1] 2561 72

km.gene <- kmeans(skin.tdf,2,nstart=20)

Then we used Elbow Method to check the opimal k value.

set.seed(6)  
wcss <- vector()  
for (i in 1:20) wcss[i] <- sum(kmeans(skin.tdf,i)$withinss)  
plot(1:20,wcss,type='b')

 Based on this plot, we chose k=8 as the optimal k value. Bucasue after 8 there is no siginificant improment on the clustering. Then we proformed k-mean clustering again with k=8.

km.gene8 <- kmeans(skin.tdf,8,nstart=20)

We also compared the total withinss value between these two k-mean clustering results.

km.gene$tot.withinss

## [1] 71774033699

km.gene8$tot.withinss

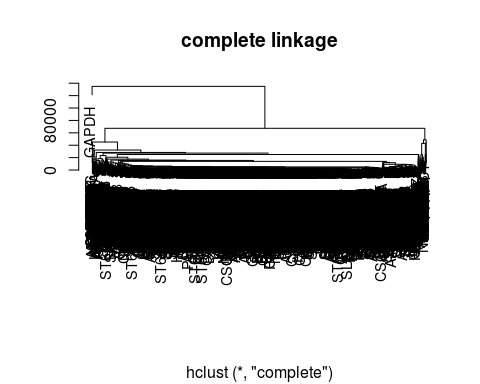
## [1] 29741890462

Base on there two numbers, we conclude that 8 clusters method is better than the 2 clusters method.

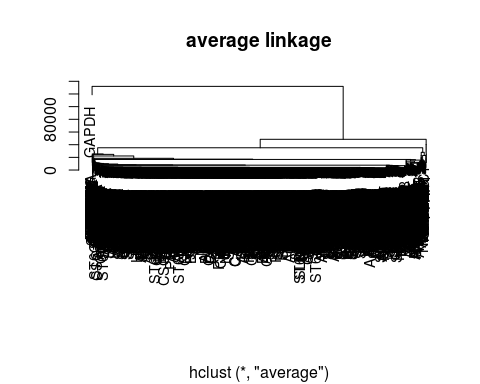
### hierarchical clustering

First, we perfrm hierachical clustering of the genes based on Euclidean distance. We tried both complete and average linkage.

# use complete linkage  
hc.gene.com <- hclust(dist(skin.tdf),method="complete")  
# use average linkage  
hc.gene.avg <- hclust(dist(skin.tdf),method="average")  
#create denrogram  
plot(hc.gene.com,main="complete linkage",xlab='',ylab='',cex=.9)

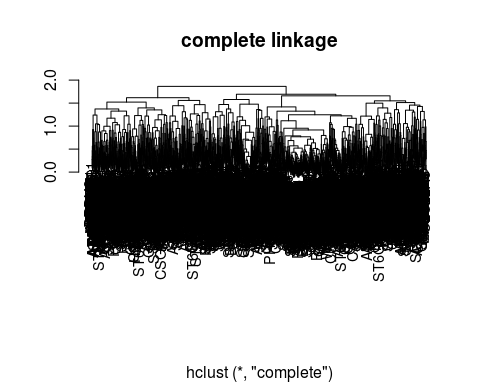


plot(hc.gene.avg,main="average linkage",xlab='',ylab='',cex=.9)

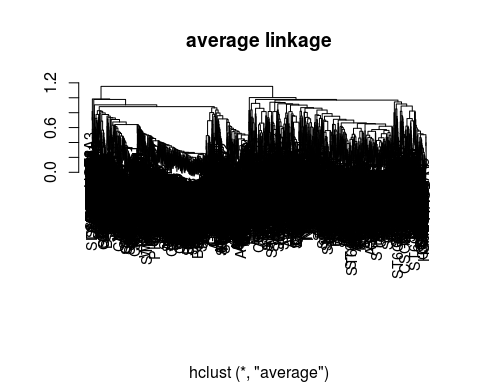


Base on the denrograms, Euclidean distance ma not be suitable here. Therefore, we used a correlation based distance for hc clustering. And this time, we also used both omplete and average linkage.

# calculate the correlation based distance  
skin.tdf.cordist <- as.dist(1-cor(skin.df[,1:2561]))  
# use complete linkage  
hccor.gene.com <- hclust(skin.tdf.cordist,method="complete")  
# use average linkage  
hccor.gene.avg <- hclust(skin.tdf.cordist,method="average")  
#create denrogram  
plot(hccor.gene.com,main="complete linkage",xlab='',ylab='',cex=.9)



plot(hccor.gene.avg,main="average linkage",xlab='',ylab='',cex=.9)

 When clustering genes, test using different definitions of distance. What do you observe? There are big differences in results from using different definitions of distance. Based on the denrograms we got, we concluded that using complete linkage is more suitable here. Because by using complete linage, there are less variances (differences) within the same cluster group.

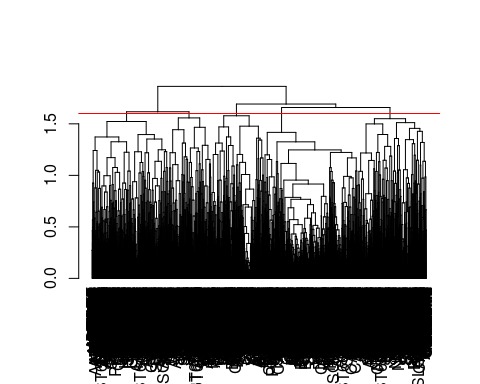
Therefore we used correlation based distance with the complete linkage for futher analsys. We deciede to cut the tree into 5 clusters.

hc.gene=cutree(hccor.gene.com,5)  
table(hc.gene)

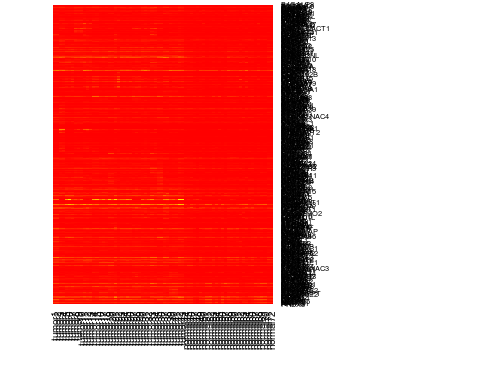
## hc.gene  
## 1 2 3 4 5   
## 527 589 717 403 325

Visualize the gene expression levels in the different samples from genes in at least two clusters. Use a visualization that enables to compare the similarity between expression levels of genes that are clustered together. Comment on what you observe.

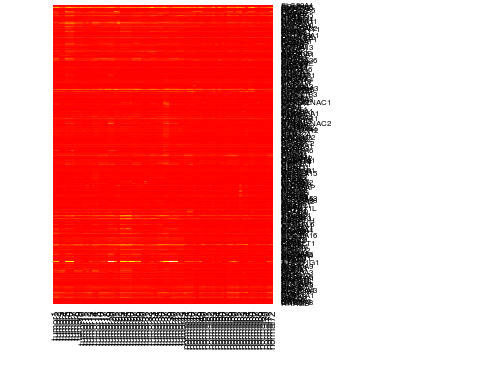
hcld.hccor <- as.dendrogram(hccor.gene.com)  
plot(hcld.hccor, cex=.3)  
abline(h=1.6, col="red")



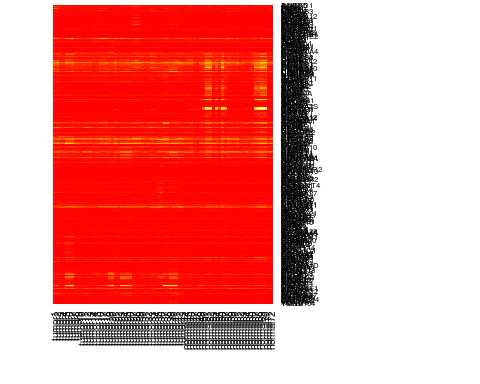
cuthcd = cut(hcld.hccor, h=1.6)  
# Visualize the gene expression levels in the different samples in clusters 1   
hc1 <- data.matrix(skin.tdf[unlist(cuthcd$lower[[1]]),])  
heatmap(hc1,Rowv=NA,Colv=NA,scale='none',col=heat.colors(256),margins=c(5,10))



# Visualize the gene expression levels in the different samples in clusters 2  
hc2 <- data.matrix(skin.tdf[unlist(cuthcd$lower[[2]]),])  
heatmap(hc2,Rowv=NA,Colv=NA,scale='none',col=heat.colors(256),margins=c(5,10))



# Visualize the gene expression levels in the different samples in clusters 3  
hc3 <- data.matrix(skin.tdf[unlist(cuthcd$lower[[3]]),])  
heatmap(hc3,Rowv=NA,Colv=NA,scale='none',col=heat.colors(256),margins=c(5,10))



## Task 2: Clustering the samples

We used the skin.df data frame for this task.

### K-means clustering

First, we used k-means clustering to seperated smaples into two clusters.

dim(skin.df)

## [1] 72 2562

#create a data frame without the labels  
skin.df.nolab <-skin.df[,1:2561]  
km.sample <- kmeans(skin.df.nolab,2,nstart=20)  
km.sample$cluster

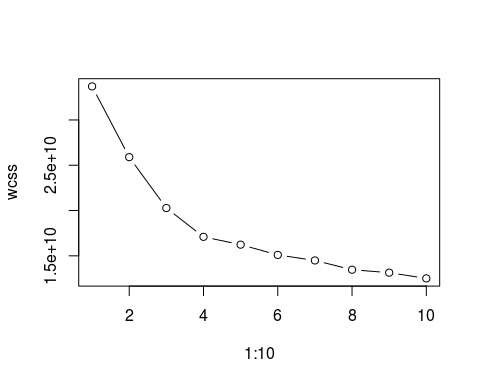
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [36] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 1 2 1 2 2 1 1 1 1 1 1 1 1 1 2 2 2 2  
## [71] 1 1

table(km.sample$cluster)

##   
## 1 2   
## 63 9

Then we used the Elbow Method to detecte the optimal k value

set.seed(6)  
wcss <- vector()  
for (i in 1:10) wcss[i] <- sum(kmeans(skin.df.nolab,i)$withinss)  
plot(1:10,wcss,type='b')

 As the plot shows, the optimal k value is around 6. Then we preformed the k-means clustering again with k=6. And we also compared the results form clustering with k=6 to clustering with k=2.

km.sample6 <- kmeans(skin.df.nolab,6,nstart=20)  
km.sample6$cluster

## [1] 4 4 3 3 2 2 2 6 6 6 3 3 3 6 6 6 6 6 2 2 4 6 2 2 2 2 4 4 4 6 6 6 6 6 6  
## [36] 6 2 2 2 2 2 6 6 5 5 5 5 5 5 5 1 1 5 1 5 1 1 5 5 5 5 5 5 5 5 5 1 1 1 1  
## [71] 5 5

# compare the 2 clusters and 6 clusters  
km.sample$tot.withinss

## [1] 25894461717

km.sample6$tot.withinss

## [1] 14958470497

table(km.sample$cluster,km.sample6$cluster)

##   
## 1 2 3 4 5 6  
## 1 0 14 5 6 20 18  
## 2 9 0 0 0 0 0

As the table shows, with 2 clusters, the smaller cluster (with 9 samples) stayed the same in 6 clusters. The bigger cluster was futher being divided into 4 sub clusters. Then we compare clustering results obtained with different approaches with the known labels.

table(km.sample$cluster,skin.df$tissue)

##   
## normal tumor  
## 1 20 43  
## 2 9 0

table(km.sample6$cluster,skin.df$tissue)

##   
## normal tumor  
## 1 9 0  
## 2 0 14  
## 3 0 5  
## 4 0 6  
## 5 20 0  
## 6 0 18

We also tried to cluster the samples into 3 clusters, which result into 1 cluster contain only tumor samples, and other 2 cluster contain only normal samples.

km.sample3 <- kmeans(skin.df.nolab,3,nstart=20)  
km.sample3$cluster

## [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  
## [36] 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 3 3 2 3 2 3 3 2 2 2 2 2 2 2 2 2 3 3 3 3  
## [71] 2 2

table(km.sample3$cluster)

##   
## 1 2 3   
## 43 20 9

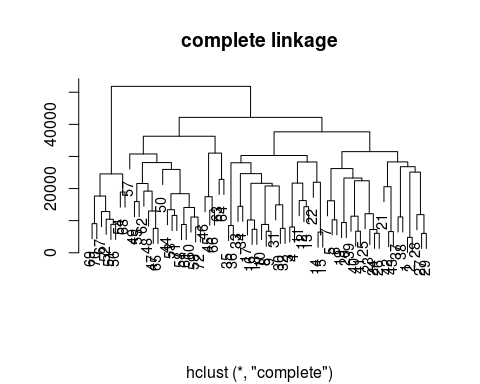
# compare clustering results with the known labels  
table(km.sample3$cluster,skin.df$tissue)

##   
## normal tumor  
## 1 0 43  
## 2 20 0  
## 3 9 0

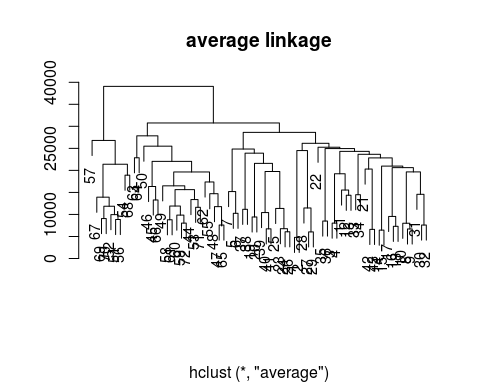
### hierarchical clustering

First, we perfrm hierachical clustering of the samples based on Euclidean distance. We tried both complete and average linkage.

# use complete linkage  
hce.sample.com <- hclust(dist(skin.df.nolab),method="complete")  
# use average linkage  
hce.sample.avg <- hclust(dist(skin.df.nolab),method="average")  
#create denrogram  
plot(hce.sample.com,main="complete linkage",xlab='',ylab='',cex=.9)

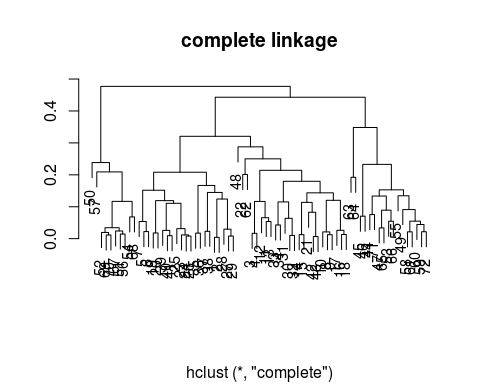


plot(hce.sample.avg,main="average linkage",xlab='',ylab='',cex=.9)

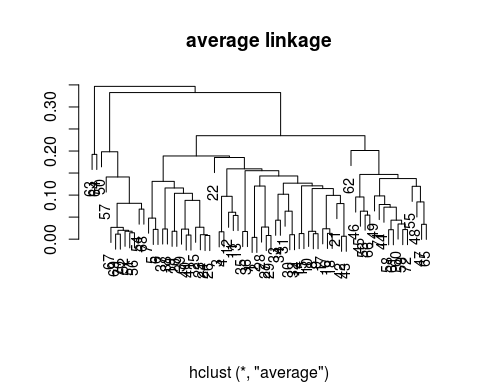


Then we used a correlation based distance for hc clustering. And this time, we also used both complete and average linkage.

# calculate the correlation based distance  
skin.df.cordist <- as.dist(1-cor(t(skin.df.nolab)))  
# use complete linkage  
hccor.sample.com <- hclust(skin.df.cordist,method="complete")  
# use average linkage  
hccor.sample.avg <- hclust(skin.df.cordist,method="average")  
#create denrogram  
plot(hccor.sample.com,main="complete linkage",xlab='',ylab='',cex=.9)



plot(hccor.sample.avg,main="average linkage",xlab='',ylab='',cex=.9)



There are differences in results from using different definitions of distance. However it is difficult to concluded which method is more suitable here. Therefore, we cut all 4 trees and compare the results to the know label.

# Euclidean distance, complete linkage  
euc.com=cutree(hce.sample.com,5)  
table(euc.com,skin.df$tissue)

##   
## euc.com normal tumor  
## 1 0 22  
## 2 0 21  
## 3 16 0  
## 4 5 0  
## 5 8 0

# Euclidean distance, average linkage  
euc.avg=cutree(hce.sample.avg,3)  
table(euc.avg,skin.df$tissue)

##   
## euc.avg normal tumor  
## 1 0 43  
## 2 20 0  
## 3 9 0

# Correlation base distance, complete linkage  
cor.com=cutree(hccor.sample.com,5)  
table(cor.com,skin.df$tissue)

##   
## cor.com normal tumor  
## 1 0 21  
## 2 2 22  
## 3 15 0  
## 4 10 0  
## 5 2 0

# Correlation base distance, average linkage  
cor.avg=cutree(hccor.sample.avg,4)  
table(cor.avg,skin.df$tissue)

##   
## cor.avg normal tumor  
## 1 0 43  
## 2 17 0  
## 3 10 0  
## 4 2 0

When clustering samples, compare clustering results obtained with different approaches with the known labels for these samples. What do you learn from this? Can you say something on how many clusters there are in this dataset?