June Kang,
Nagaswaroop Kengunte Nagaraj
03/08/2017
Prof. Jeff Solka, Ph.D.
Binf 702 Project

Use Golub Gene Expression Data to Classify AML and ALL

Abstract

Golub data is the first classical datasets used in bioinformatics. Golub consists of gene expression values for 3051 genes from 38 leukemia patients. The 38 leukemia patients in the golub dataset are consist of 27 patients with acute lymphoblastic leukemia (ALL) and 11 patients with acute myeloid leukemia (AML). The purpose of this study is to successfully classify AML and ALL. Hierarchical clustering showed that 13 out of 20 genes of interest have a somewhat separate clustering. 9 out of the 13 genes are negatively expressed. k -mean shows that these cluster structures separates well. Bootstrap showed that most of the expression lies in the 95% confidence interval. These genes showed biological significance in the AML and ALL. These genes should be further studied in the future due to its significant variance was shown.

Background and Objectives

Identifying cancer classes are important precursors for treating cancers in the long run. Detailed subclassification not only helps doctors prescribe specific targeted treatments, it's also beneficial to any future cancer research. Golub data is a classic example of cancer classification analysis that can be applied to many statistical ideas. Golub data has aided the process of discovering and predicting cancer class(Golub et al. 1999).

The golub data contains the gene expression for 6817 genes from 38 patients. These 38 patients have either the acute myeloid leukemia (AML) or acute lymphoblastic leukemia (ALL). The ALL contains two different cancer cells, T-cell leukemia, and B-cell leukemia. Both the ALL cancer types received treatment. The AML patients are divided into the control and the experimental group. The control group didn't receive any treatment and the disease has run its course. The experimental group received generic leukemia treatments to slow the course of the disease(Golub et al. 1999).

The difference between ALL and AML arise from their surface molecule differences. The ALL has a lymphoid precursor and the AML are arising from a lymphoid precursor. These differences were confirmed by the antibiotics experiments conducted in the 1970s (Tsukimoto, Wong, and Lampkin 1976; Schlossman et al. 1976; Roper et al. 1983; Pesando et al. 1979). Some acute leukemia can be divided into subtypes based on the specific chromosomal translocations (Golub et al. 1996; McLean et al. 1996; Shurtleff et al. 1995; Romana et al. 1995; Rowley 1973). Distinguish the difference between AML and ALL is critical for the treatment. There are no test diagnostic that can distinguish the difference between AML and ALL(Golub et al. 1999).

Cells samples were obtained from the peripheral blood (10) and the bone marrow (24) of the Leukemia patients. Hybridized RNA were prepared from the bone marrow mononuclear cells and used on the probes. Affymetrix microarray was applied to human acute leukemias to generate the quantitative expression data (Golub et al. 1999).

The golub data is divided into two dimensions. There are 6817 rows that consist of the gene expressions and 38 columns that matched each Leukemia patients. 27 columns are consists of the ALL patients' data and 11 are AML patients' data. The data is collected in the multtest package and it's part of the Bioconductor. The tumor class for ALL is represented by 0 and for AML is coded as 1 in numeric vector golub.cl. The columns correspond to three layers, the gene index, gene ID and the golub name("R: Gene Expression Dataset from Golub et al." 2017).

The challenge of cancer treatment can be minimized by figuring out the specific target therapies that can be used for each cancer class. By analyzing the golub data, the identification of cancer classed can be improved significantly and benefit the medical research community. The goal of this project is to analyze and determine the difference between the AML and ALL use R and statistics.

Computational Methods

Microarray experiments in the fields include image analysis, experimental design, cluster and discriminant analysis. Multiple hypothesis testing raises numerous statistical questions. In this project, we focus on the classification of tumors in the golub data using gene expression values. Tumor classification is associated with three main types of statistical problems. Identification of new tumor classes using gene expression profiles, cluster analysis, and unsupervised learning. Classify malignancies into known classes by use discriminant analysis and supervised learning. And the identification of "marker" genes that can characterize the different tumor classes through variable selection(Dudoit, Fridlyand, and Speed 2002).

Principal Component Analysis

Principal component analysis is a descriptive method that can be used to analyze correlations between variables. The principal component analysis help find new directions in the data at where the maximal variation is. Principal component analysis is performed on the golub data. Percent of variance explained by the components were generated. The weights of the eigenvectors were analyzed and considered. The first ten overexpressed genes and last ten underexpressed genes were computed.

Cluster Analysis

Cluster analysis or clustering is the task of grouping a set of objects into groups where each group has a unique characteristic. The groups are structured in such a way that the objects are similar in the same group are dissimilar when compared with another group("Clustering - Introduction" 2017).

Clustering algorithms can be classified as:

- Exclusive Clustering (K-Means clustering)
- Overlapping Clustering (Fuzzy C-Means clustering)
- Hierarchical Clustering
- Probabilistic Clustering (Mixture of Gaussian)

Cluster analysis is usually based on Distance measure that could determine the similarity between two genes. Cluster algorithm is used to determine the similarity between two clusters.

Some of the distance measures include Euclidean distance, Manhattan distance, and Correlation distance, etc. Correlation distance is used to measure trends and relative differences. Euclidean and Manhattan distance measure absolute differences between two points. Manhattan distance is more robust against outliers than correlation distance and euclidean distance. The two cluster algorithms that we are proposing for the project are K-Means clustering and Hierarchical clustering("Clustering - Introduction" 2017; Sánchez, n.d.).

Hierarchical clustering

Hierarchical clustering will be performed to group genes with similar expression patterns in a series of clustering tree branches. The distance between two expression vectors is calculated to determine the similarity between two genes. Linkage method is used to determine the similarity between clusters ("Microarray Data Analysis" 2017). The tree can be built in two distinct ways, bottom-up and top-down. The bottom-up algorithm involves the agglomerative clustering. And top-down is determined by the divisive clustering. Bottom-up hierarchical clustering is more suitable for the golub data analysis. Bottom-up algorithm starts with n clusters. The two closest clusters were merged using the measurements between the cluster dissimilarity, which reflects the shape of the clusters. Distance calculation between two clusters is based on the pairwise distances between members of the clusters (Sánchez, n.d.).

There are three ways to calculate the distance between clusters. Single-linkage clustering is the shortest distance from one cluster to another cluster. Complete-linkage clustering is the greatest distance from any member of one cluster to any member of the other cluster.

Average-linkage clustering is the average distance from any member of one cluster to any member of the other cluster ("Clustering - Introduction" 2017).

K-Means clustering

K-means clustering classifies a given data set through a number of k clusters. k centroids need to be defined for each cluster. These centroids should be placed in a calculative way. The points from the data set are associated with the closest centroid. The first step and an early groupage are complete when no points are pending. The k new centroids as barycenters of the clusters resulting from the previous step were calculated. After we have these k new centroids, a new binding is formed between the same points and the nearest newly formed centroid. A loop has been generated. k centroids will change the location step by step until no more changes are done.

The algorithm is applied to the golub data as follows:

1. Specify K number of clusters. These represent an initial number of centroids.
2. Randomly Assign each gene value to the group that has the closest centroid.
3. When all gene values have been assigned, calculate the centroid (mean expression profile) for each cluster.
4. Shuffle genes among clusters such that the gene is in the cluster whose centroid (mean expression profile) is the closest to the gene expression profile.
5. Repeat Steps 3 and 4 until the centroids no longer move. This produces a separation of the gene values into groups from which the metric to be minimized can be calculated("Clustering - Introduction" 2017).

Bootstrap

One can estimate statistical parameters (like mean of a population and its confidence interval) using Bootstrap method which follows the principle that these parameters are estimated from the sample by means of resampling with replacement. Bootstrapping does not make assumption (like whether the data is normally distributed so that it can be characterized using mean and variance) about the distribution of the sample like most of the parametric approaches do. A bootstrapped sample is obtained from the sample by resampling with replacement. This can be explained as: each observation from the sample has the same probability of being selected in the bootstrapped sample, and as it is with replacement observations have the probability to be selected subsequently in the bootstrapped sample. This process of resampling is applied multiple times to obtain a number of bootstrapped samples. Mean of all the bootstrapped samples are calculated and then all the bootstrapped entities are sorted which would lead us to find the overall mean, confidence interval, standard deviation, 2.5th percentile and 97.5th percentile.

Results and Discussion

We can obtain the golub data and perform the statistical procedures to analyze the golub data using principal component analysis, hierarchical clustering, k-mean, bootstrap and explain the biological significance.

Principal component analysis

Principal component analysis (PCA) were performed on the golub data. Principal component analysis is a descriptive method used to analyze correlations between variables. PCA component 1 and 2 explained 72.41% of variance (Figure 1). All the weights of the first

eigenvector on component 1 are between 0.13 and 0.17. They are all positive and very similar in size. The first component is very close to the sum of the variables and the correlations are equal to 0.9999. The second principal components have 27 positive weights and 11 negative weights. This is correspond to the ALL and AML patients number. Thus principal components 2 is a lot more interesting than component 1. The layout of the genes and patients in the two components can be visualize in the biplot (Figure 2).

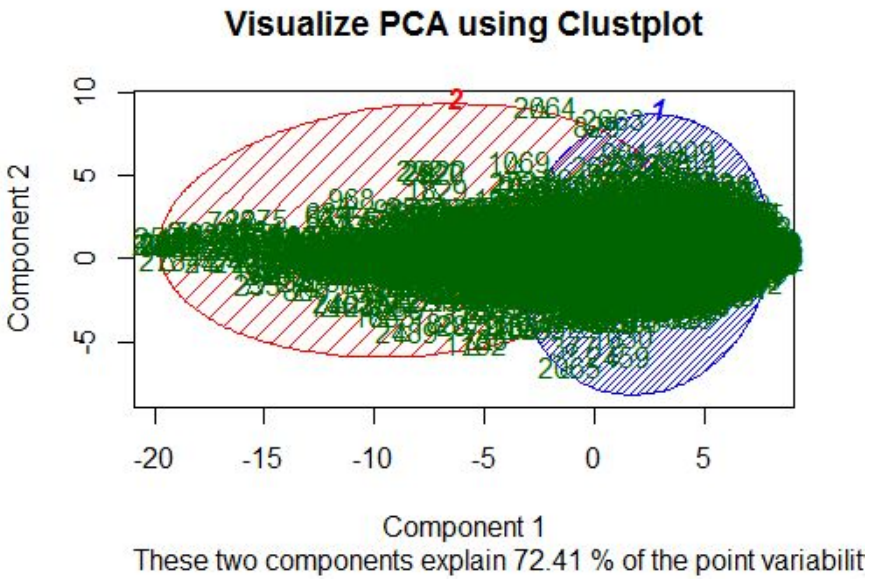

Figure 1. Visualization of the PCA component 1 and 2 using k-mean fit. The first two components explain 72.41% of the variance.

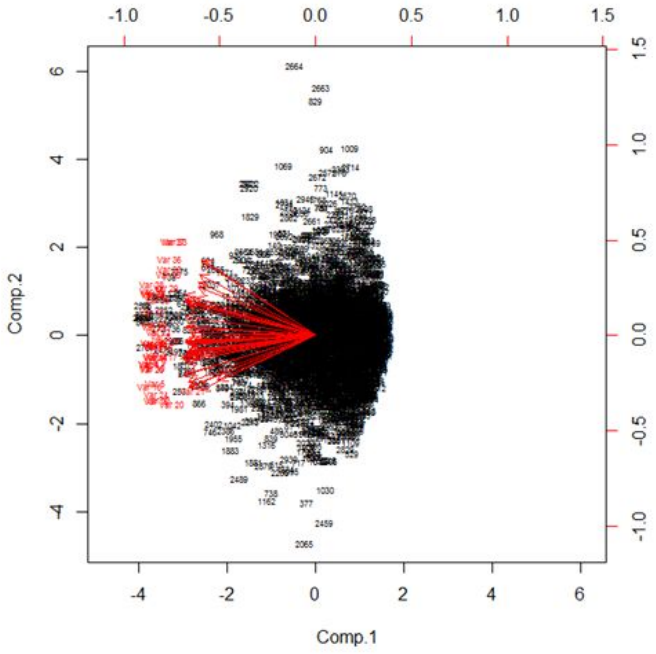

Figure 2. Biplot of selected genes and patients from the golub data. The left and bot axis refer to the component scores. The top and right refer to the patient score.

The first ten overexpressed genes and last ten underexpressed genes were computed and analyzed using hierarchical clustering.

#Overexpressed genes
[1] "TCL1 gene (T cell leukemia) extracted from H.sapiens mRNA for Tcell leukemia/lymphoma 1"
[2] "CD24 signal transducer mRNA and 3' region"
[3] "GB DEF = (lambda) DNA for immunoglobin light chain"
[4] "MB-1 gene"
[5] "Terminal transferase mRNA"
[6] "IGB Immunoglobulin-associated beta (B29)"
[7] "C-myb gene extracted from Human (c-myb) gene, complete primary cds, and five complete alternatively spliced cds"
[8] "Adenosine triphosphatase, calcium"
[9] "RETINOBLASTOMA BINDING PROTEIN P48"
[10] "Cytoplasmic dynein light chain 1 (hdlc1) mRNA"

#Underexpressed gene

[1] "GB DEF = Cystic fibrosis antigen mRNA"
[2] "CYSTATIN A"
[3] "LYZ Lysozyme"
[4] "MACROPHAGE INFLAMMATORY PROTEIN 1-ALPHA PRECURSOR"
[5] "GRO2 GRO2 oncogene"
[6] "Azurocidin gene"
[7] "LGALS3 Lectin, galactoside-binding, soluble, 3 (galectin 3) (NOTE: redefinition of symbol)"
[8] "DF D component of complement (adipsin)"
[9] "CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)"
[10] "Interleukin 8 (IL8) gene"
[11] "INTERLEUKIN-8 PRECURSOR"

Hierarchical clustering

Clustering algorithms are applied to the golub data where the gene expression data are used to identify new classes of biological subtypes and to identify batch effects or outliers. It can also identify groups of possibly co-regulated genes, spatial or temporal patterns (e.g. in cell cycle or analysis of different brain areas), and reduce redundancy (e.g. for variable selection in predictive models). The hierarchical clustering for all 20 genes are computed using single linkage cluster with euclidian distance matrix with the nearest neighbor algorithm. 13 out of 20 of these genes have a somewhat clear separation among clusters (Figure 3). 9 out of the 13 genes are negatively expressed.

There are a lot of advantages in using clustering. Clustering leads to readily interpretable figures. Clustering can increase the signal when gene averages are calculated k(Eisen). Clustering can be helpful for identifying patterns in time or space. Clustering is essential when seeking new subclasses of tumor samples (Sánchez, n.d.).

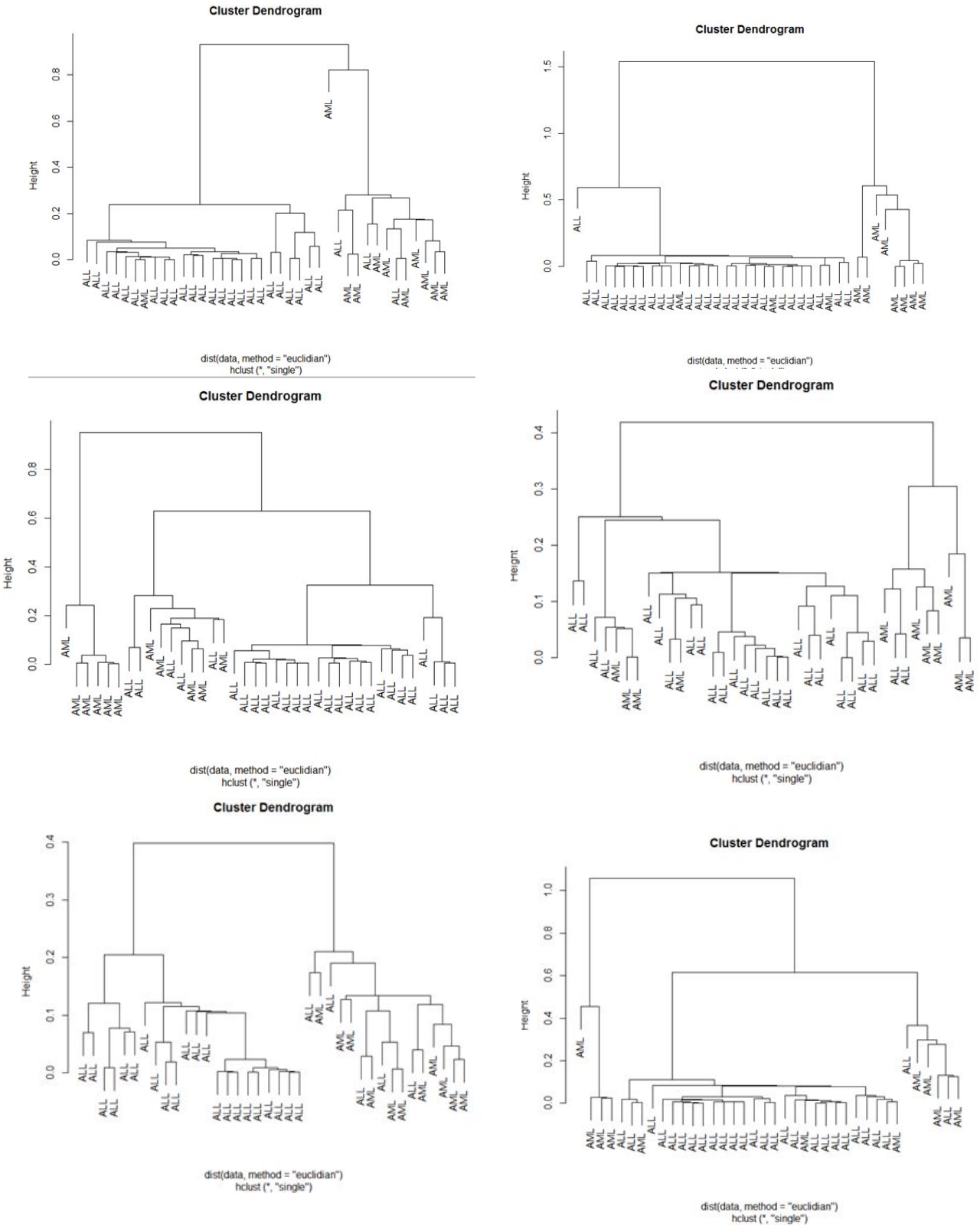

Figure 3. Example Clusters shown CST3 Cystatin C, interleukin 8(IL8) gene, DF D component

of complement, Adenosine triphosphatase, cytoplasmic dynein light chain 1 (hdlc1) mRNA and GRO2 oncogene, respectively.

K-mean clustering

K-Mean clustering is an algorithm to group the data into coherent subsets. There are two steps for K-Means clustering. The first step is to randomly initialize the centroids (K number of clusters we are interested in on the given data set). Here K signifies the number of clusters the data is to be grouped into. The next step is an iterative process where we have two subtasks under it. The first being assign the cluster to the nearest centroid. Second, calculate the mean of the group and move the centroid to the position of the mean. These two subtasks are repeated until the centroids do not move further.

Choosing the number of clusters (K) is a challenging problem when we do not have a clear idea about how the data is classified. However, this can be solved by one of the few methods, popularly called “Elbow method”. Run the K-means with K=1 and compute the distortion function, next run it with K=2, and calculate the distortion function again. Continuing this process by increasing the value of K and plotting a graph of the distortion function relative to K value we obtain a curve as shown in the Figure 4. The graph shows how distortion decreases as the K value increases.

The K number of cluster can be determined using this graph. As seen from the graph we see a rapid decrease in the distortion until we reach K=2 after which the distortion becomes relatively constant. At K=2 it sort of creates an elbow shaped curve and hence the name of the method. Thus we can come to a conclusion and select K=2 as the value for the number of

clusters our data needs to be divided. Since we already know about the golub data we can be sure that this method is a good approximation to calculate K number of clusters.

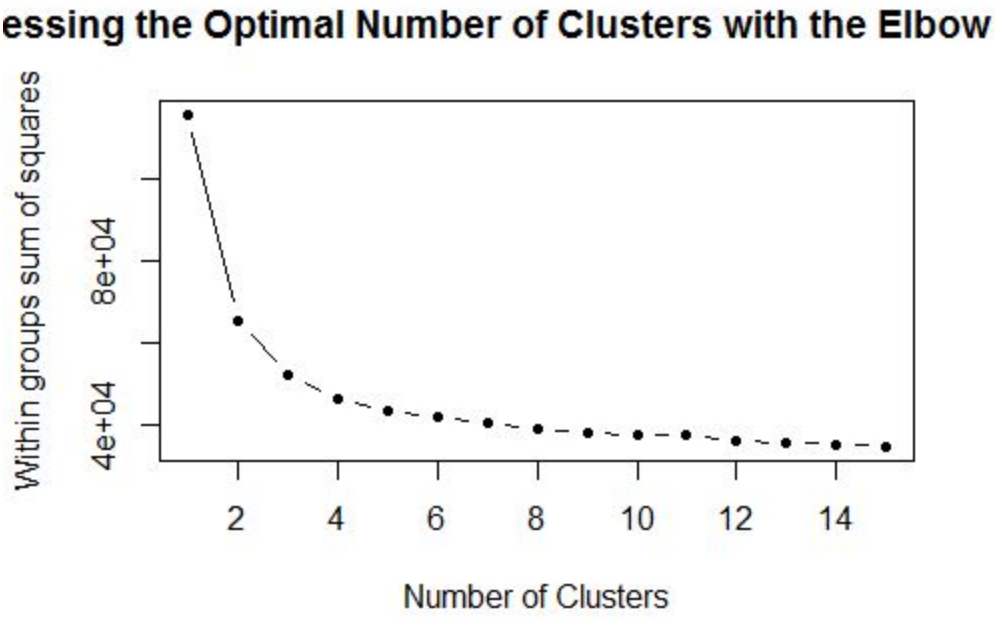

Figure 4. Assessing the optimal number of clusters with the elbow method.

Applying K-Means algorithm to the whole of golub data with a K value equal to 2 i.e dividing the data into two clusters. According to the golub paper we can say that the two groups belong to the ALL and AML class of cancer. The Figure 5 shows the K-Means applied on the entire gene expression values present in the golub data.

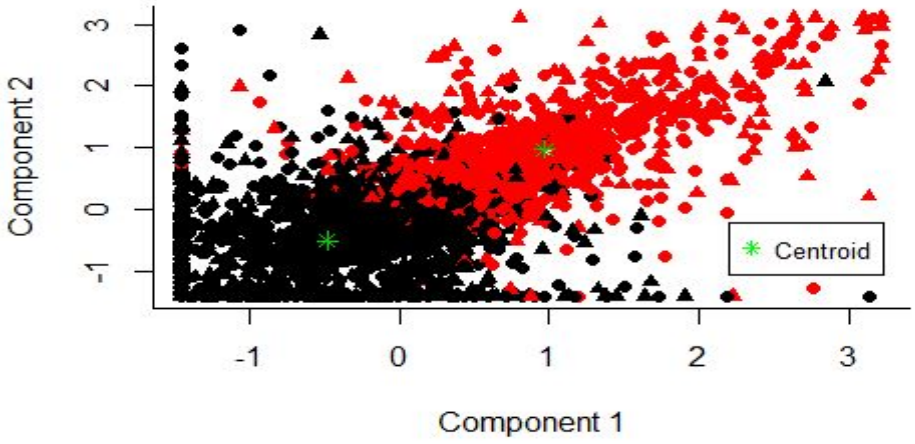

Figure 5. K-Means applied on the gene expression values of the golub data.

The above figure depicts that the data is divided into two clusters (Groups) and these can be categorized into two different types of cancer as they classified upon the gene expression levels. Figure 6 shows the K-Means cluster plot of a single gene CST3 Cystatin C, a gene predicted to show clear separation of clustering.

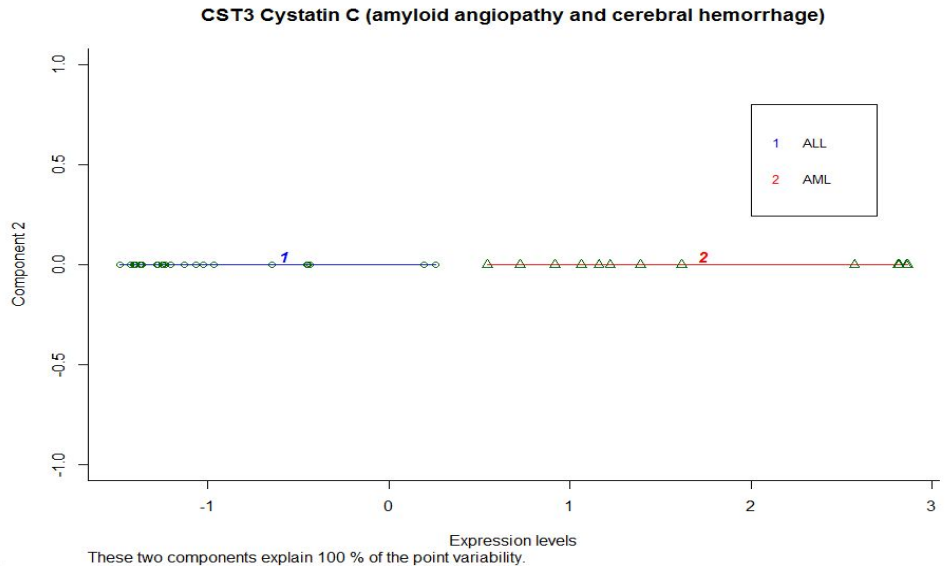

Figure 6. Plot of K-means clustering on CST3 Cystatin C gene expression level.

As we can see from Figure 6 that when we consider a single gene it is much evident on the power of using a K-Mean algorithm to detect the presence of two types of cancer. Here since we know from the Golub paper, there are two class of cancer present in the data plus noticing that the CST3 Cystatin C gene has been expressed more by AML class patients and hence giving us a clear indication to depict the two classes.

K-Means can be applied in various ways as well. There are a numerous packages available in R which makes understanding K-Means better and also to apply K-Means to understand the data in a clear view. One such package we have used here is “factoextra”. By using this we obtain Figure 7.

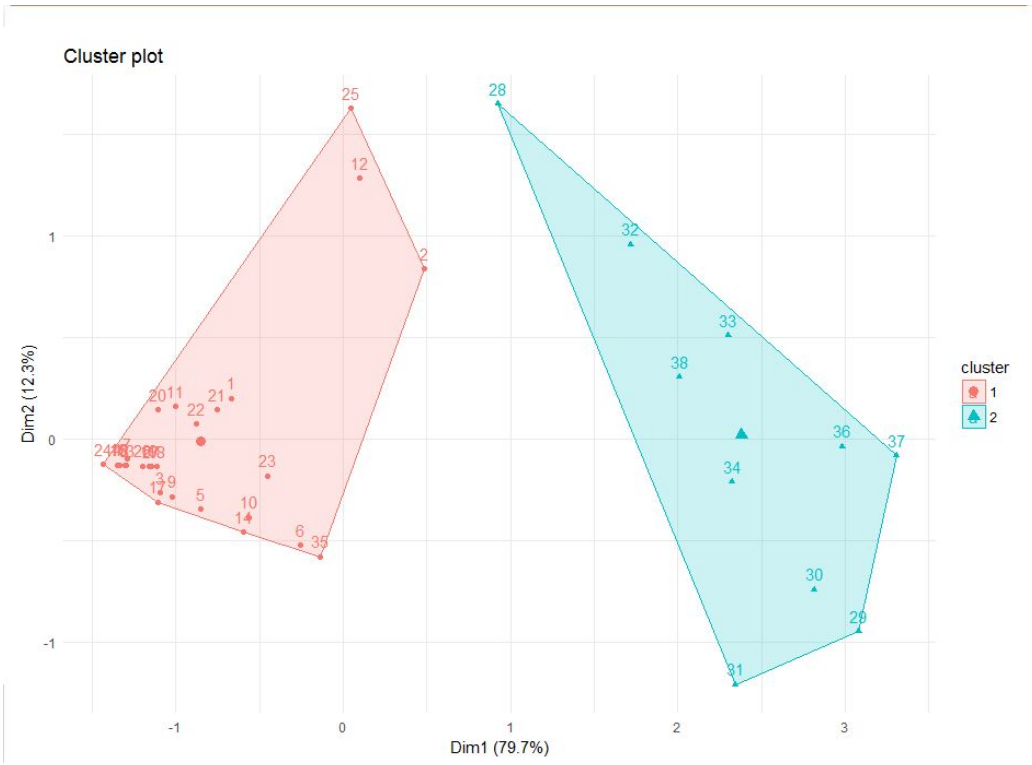

Figure 7. Applying the K-Means to CST3 Cystatin C, Interleukin 8, DF D genes using the "Factoextra" package in R.

In Figure 7 we are analyzing how the 3 genes i.e CST3 Cystatin C, Interleukin 8, DF D can be first correlated using PCA and then plotted into two cluster. This plot is obtained by using "Factoextra" package. Here the data is divided into 2 groups and by noticing the gene expression values of all the three genes and from the golub paper that the last 11 samples belonged to the AML patients, we can conclude that the cluster 2 belongs to the AML patients and cluster 1 belongs to the ALL group of patients.

Similarly we have plotted a few more differentiation examples as seen in Figure 8 using the genes which have proven to be the more significant genes in determining the classes as example.

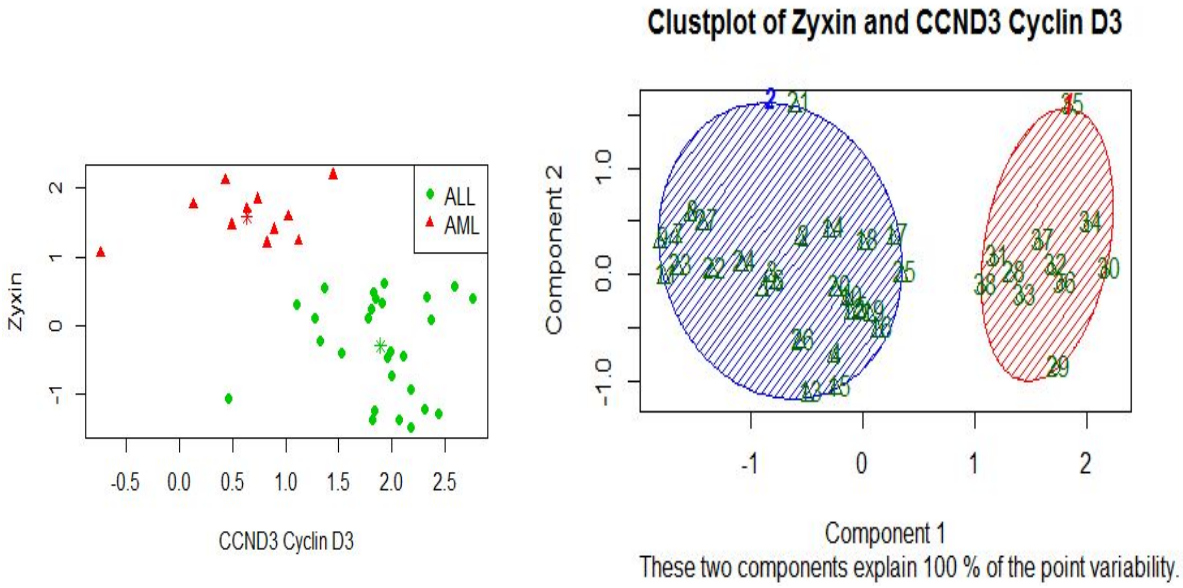

Figure 8. Comparison of Zyxin and CCND3 genes.

Bootstrap

A bootstrapping sample is an approximation of the population distribution. Therefore we can be sure that the bootstrapping distribution provides a 95% confidence interval for the middle 95% values of the population.

In order to estimate the population mean better, we use the confidence interval of the bootstrapped sample. Consider the mean from bootstrap sample, and then find the confidence interval. The mean of the bootstrapped sample is just an estimate of the actual population mean. Since the mean is based on sample data and not the population data, it is unlikely that the sample mean is equal to population mean. Hence it is better to use the confidence interval to estimate population mean.

Confidence intervals are based on the sampling distribution of a statistic. If a statistic has no bias as an estimator of a parameter, its sampling distribution is centered at the true value of the parameter. The confidence interval helps you assess the practical significance of your estimate for the population parameter.

From the results as shown in Figure 9, the estimate for the population mean after applying K-Means for the gene CST3 Cystatin C is approximately -1.016898 for cluster 1 and 1.815716 for cluster 2. You can be 95% confident that the population mean is between approximately -1.2298863 and -0.7258075 for cluster 1 and 1.181629 and 2.450780 for cluster 2.

Applying bootstrap to the CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)

```
data= data.frame(golub[829,])
initial <-as.matrix(tapply(golub[829,],gol.fac,mean),nrow
=2, ncol=1)
table(cl$cluster,gol.fac)
n <- nrow(data); nboot<-1000
boot.cl <- matrix(0,nrow=nboot,ncol = 2)


for (i in 1:nboot) {
  dat.star <- data[sample(1:n,replace=TRUE),]
  cl <- kmeans(dat.star, initial, nstart = 10)
  boot.cl[i,] <- c(cl$centers[1,],cl$centers[2,])
}
```

```
> cl <- kmeans(data1, 2, nstart = 25)
> cl
K-means clustering with 2 clusters of
sizes 14, 24

Cluster means:
 golub.829...
1    -1.016898
2     1.815716

> mean(boot.cl[,1])
[1] -1.01304
> mean(boot.cl[,2])
[1] 1.80983
> quantile(boot.cl[,1],c(0.025,0.975))
     2.5%      97.5%
-1.2298863 -0.7258075
> quantile(boot.cl[,2],c(0.025,0.975))
   2.5%    97.5%
1.181629 2.450780
```

Figure 9. Results of Bootstrap on gene CST3 Cystatin C.

From Figure 9 we can conclude that the K-means mean and the bootstrap mean are almost equal and also it can be observed that the estimation is quite precise because the 95% bootstrap confidence intervals are fairly small.

Figure 10 below shows us how the means of the bootstrapped sample of gene expression values are distributed for both cluster 1 and cluster 2 using a histogram.

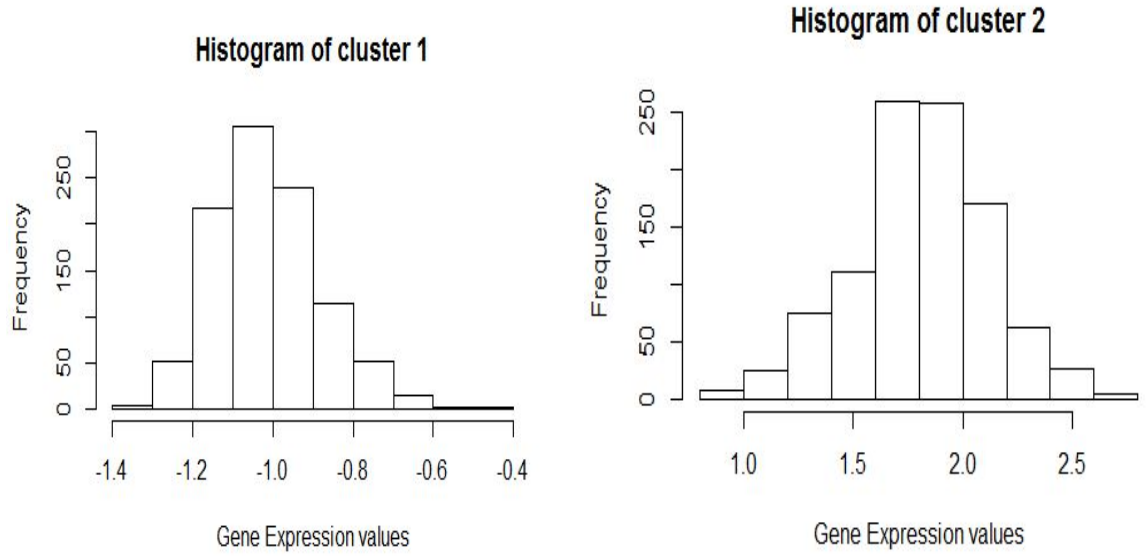

Figure 10. Histogram of bootstrapped sample means of cluster 1 and 2.

Biological significance

As we have seen from the above hierarchical clustering that 13 of the genes proved to be significantly expressive in determining the classification of ALL and AML. So it is important to consider the biological significance of these, it is listed as follows :

- C-myb gene extracted from Human (c-myb) gene- The myb gene family is significant in cell growth, differentiation and apoptosis.
- ATPase is a solute enzyme that pumps sodium out of cells while pumping potassium into cells, both against their concentration gradients. This pumping is active (i.e. it uses energy from ATP) and is important for cell physiology.

- Retinoblastoma Binding Protein P48 is a core histone-binding subunit that targets chromatin assembly factors, chromatin remodeling factors and histone deacetylases. They are regulated by nucleosomal DNA.
- Cytoplasmic dynein light chain 1 (hdlc1) mRNA- Binding of this protein destabilizes the neuronal nitric oxide synthase dimer4
- Cystatin-A- Has an important role in desmosome-mediated cell-cell adhesion in the lower levels of the epidermis
- Macrophage inflammatory protein-1 alpha (MIP-1α)- Primarily associated with cell adhesion and migration.
- GRO2 oncogene- growth-regulated oncogene (gro) found in human colon carcinoma growth and metastasis.
- Azurocidin 1- important multifunctional inflammatory mediator.
- SYN1 synapsin I [Homo sapiens (human)]- This gene is a member of the synapsin gene family.This member of the synapsin family plays a role in regulation of axonogenesis and synaptogenesis.
- Cystatin C- As an inhibitor of cysteine proteinases, this protein is thought to serve an important physiological role as a local regulator of this enzyme activity.
- Interleukin 8 (IL8) gene- IL-8 is a chemotactic factor that attracts neutrophils, basophils, and T-cells, but not monocytes. It is also involved in neutrophil activation.

Conclusions

In medical settings, patients are often diagnosed into classes corresponding to types of diseases. In bioinformatics, the question arises whether the diagnosis of a patient can be predicted by gene expression values, or which genes play an important role in the prediction of classes. The purpose of clustering analysis of biological data is to gain insight into the underlying structure in the complex data.

A key question when analyzing high throughput data is whether the information provided by the measured biological entities is related to the experimental conditions, or, rather, to some interfering signals, such as experimental bias or artefacts. To better understand the underlying structure of the data in a 'blind' (unsupervised) way. PCA is particularly powerful if the biological question is related to the highest variance.

Brief Description

There are many approaches that are focused on just clustering and thus forgetting the gene function or rather the biological knowledge of the classified genes. Most of the genomic data consists of high level of noise and thus makes the clustering alone an unreliable method of classifying. Thus it becomes important to understand the gene functionality and the biological knowledge of the gene set while applying clustering.

Some of the well known methods have been suggested by Hanisch *et al*. (2002) who proposed incorporating a metabolic pathway while Cheng *et al*. (2004) incorporating the Gene Ontology (GO) (Ashburner *et al*., 2000) into clustering. Both approaches work by first defining a distance metric based on a biological network, either a pathway or GO. Then this metric is

combined with the usual expression-based metric using their average, which is used in a clustering algorithm.

Au et al. presented a particular attribute validation technique for clustering. They selected a subset of top genes from each obtained cluster to make up a gene pool, and then they run classification experiments on the selected genes to see whether or not the results are backed by the ground truth and which method performs the best. Thus, they exploit class information on samples to validate the results of gene clustering. The good accuracy reached by selecting few genes from the clusters reveals that the good diagnostic information existing in a small set of genes can be effectively selected by the algorithm. It is an interesting new way of clustering validation, by integrating clustering and feature selection.

Appendix

```
source("https://bioconductor.org/biocLite.R")
biocLite(“multtest”)

library(multtest); data(golub)
library(MASS)
gol.fac <- factor(golub.cl,levels=0:1, labels= c("ALL","AML"))

data <- golub; p <- ncol(data); n <- nrow(data) ; nboot<-1000
eigenvalues <- array(dim=c(nboot,p))
for (i in 1:nboot){dat.star <- data[sample(1:n,replace=TRUE),]
eigenvalues[i,] <- eigen(cor(dat.star))$values}
for (j in 1:5) cat(j,as.numeric(quantile(eigenvalues[,j],
                                         +
c(0.025,0.975))),"\n" )
sum(eigen(cor(golub))$values[1:2])/38*100

-eigen(cor(golub))$vec[,1:2]
pca <- princomp(golub, center = TRUE, cor=TRUE, scores=TRUE)
o <- order(pca$scores[,2])
Golub.gnames[o[1:10],2]
Golub.gnames[o[3041:3051],2]
```

#example code for Hierarchical clustering

```
grep("MPO Myeloperoxidase", golub.gnames[,2])
grep("CST3 Cystatin C", golub.gnames[,2])
grep("INTERLEUKIN-8 PRECURSOR", golub.gnames[,2])
grep("Interleukin 8", golub.gnames[,2])
grep("TCL1 gene", golub.gnames[,2])

data= data.frame(golub[2664,])
zyxinclus <-hclust(dist(data, method="euclidian"),
method="single")
plot(zyxinclus, labels= gol.fac)

data= data.frame(golub[2734,])
zyxinclus <-hclust(dist(data, method="euclidian"),
method="single")
plot(zyxinclus, labels= gol.fac)
```

#K-mean
#Assessing the K number of cluster

```
mydata <- golub
wss <- (nrow(mydata)-1)*sum(apply(mydata,2,var))
for (i in 2:15) wss[i] <- sum(kmeans(mydata,
                                     centers=i)$withinss)
plot(1:15, wss, type="b", xlab="Number of Clusters",
     ylab="Within groups sum of squares",
     main="Assessing the Optimal Number of Clusters with the
Elbow Method",
     pch=20, cex=1)
```

#K-Means on the complete golub dataset

```
fit <- kmeans(mydata,2)
plot(mydata,col=fit$cluster,pch=16:17,xlab='Component
1',ylab='Component 2',bty='L')
points(fit$centers,col='51',pch=8)
legend(2.2,-0.2,legend="Centroid",pch=8,col ="green",cex=0.8)

#K-Means on the gene CST3 Cystatin C
data1= data.frame(golub[829,])
initial <-as.matrix(tapply(golub[829,],gol.fac,mean),nrow =2,
ncol=1)
cl <- kmeans(data1,initial, nstart=10)
cl
library("cluster")
```

```
clusplot(data1, cl$cluster,xlab="Expression
levels",main=golub.gnames[829,2], color=TRUE, shade=TRUE,
labels=4, lines=0,bty='L')
legend(2,0.8,legend=c("ALL","AML"),pch=49:50,col
=c("blue","red"),cex=0.8)

#K-Means on the genes CST3 Cystatin C, Interleukin 8, DF D
data2=data.frame(golub[1009,])
initial <-as.matrix(tapply(golub[1009,],gol.fac,mean),nrow =2,
ncol=1)
dfd <- kmeans(data2,initial, nstart=10)
dfd
data3= data.frame(golub[2663,])
initial <-as.matrix(tapply(golub[2663,],gol.fac,mean),nrow =2,
ncol=1)
il8 <- kmeans(data3,initial, nstart=10)
il8
data4<-data.frame(golub[829,],golub[2663,],golub[1009,])
km.res <- kmeans(data4, 2, nstart = 25)
#instal.packages(“factoextra”)
library("factoextra")
fviz_cluster(km.res, data = data4, frame.type = "convex")+
  theme_minimal()

#zyxin n ccnd3 plot code

data_z_c <- data.frame(golub[1042,],golub[2124,])
zyx_ccnd3 <- kmeans(data_z_c, 2,nstart = 10)
zyx_ccnd3

plot(data_z_c,col=zyx_ccnd3$cluster+1,pch = c(16,
17)[as.numeric(gol.fac)])
legend("topright",legend=c("ALL","AML"),pch=16:17,col =
c('51','red'))
points(cl$centers, pch=8,col=c('red','51'))
clusplot(data_z_c, cl$cluster, main="Clustplot of Zyxin and
CCND3 Cyclin D3" ,color=TRUE, shade=TRUE, labels=2, lines=0)


#bootstrap
data= data.frame(golub[829,])
table(cl$cluster,gol.fac)
n <- nrow(data); nboot<-1000
boot.cl <- matrix(0,nrow=nboot,ncol = 2)
```

```
for (i in 1:nboot) {
  dat.star <- data[sample(1:n,replace=TRUE),]
  cl <- kmeans(dat.star, initial, nstart = 10)
  boot.cl[i,] <- c(cl$centers[1,],cl$centers[2,])


}

hist(boot.cl[,1],main="Histogram of cluster 1",xlab="Gene
Expression values")
hist(boot.cl[,2],main="Histogram of cluster 2",xlab="Gene
Expression values")

quantile(boot.cl[,1],c(0.025,0.975))
quantile(boot.cl[,2],c(0.025,0.975))
mean(boot.cl[,1])
mean(boot.cl[,2])
```

Reference

"Clustering - Introduction." 2017. Accessed March 20. https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/index.html.

Dudoit, Sandrine, Jane Fridlyand, and Terence P. Speed. 2002. "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data." *Journal of the American Statistical Association* 97 (457): 77–87. doi:10.1198/016214502753479248.

Golub, T R, A Goga, G F Barker, D E Afar, J McLaughlin, S K Bohlander, J D Rowley, O N Witte, and D G Gilliland. 1996. "Oligomerization of the ABL Tyrosine Kinase by the Ets Protein TEL in Human Leukemia." *Molecular and Cellular Biology* 16 (8): 4107–16.

Golub, T. R., D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, et al. 1999. "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring." *Science (New York, N.Y.)* 286 (5439): 531–37.

Krijnen, Wim. 2009. *Applied Statistics for Bioinformatics Using R*. The Netherlands: Hanze University, Institute for Life Science and Technology.

McLean, Thomas W., Sarah Ringold, Donna Neuberg, Kimberly Stegmaier, Ramana Tantravahi, Jerome Ritz, H. Phillip Koeffler, et al. 1996. "TEL/AML-1 Dimerizes and Is Associated with a Favorable Outcome in Childhood Acute Lymphoblastic Leukemia." *Blood* 88 (11): 4252–58.

"Microarray Data Analysis." 2017. Accessed March 20. http://compbio.uthsc.edu/microarray/lecture1.htm.

Pesando, J. M., J. Ritz, H. Lazarus, S. B. Costello, S. Sallan, and S. F. Schlossman. 1979. "Leukemia-Associated Antigens in ALL." *Blood* 54 (6): 1240–48.

"R: Gene Expression Dataset from Golub et Al. (1999)." 2017. Accessed March 18. http://svitsrv25.epfl.ch/R-doc/library/multtest/html/golub.html.

Romana, S. P., H. Poirel, M. Leconiat, M. A. Flexor, M. Mauchauffé, P. Jonveaux, E. A. Macintyre, R. Berger, and O. A. Bernard. 1995. "High Frequency of t(12;21) in Childhood B-Lineage Acute Lymphoblastic Leukemia." *Blood* 86 (11): 4263–69.

Roper, M., W. M. Crist, R. Metzgar, A. H. Ragab, S. Smith, K. Starling, J. Pullen, B. Leventhal, A. A. Bartolucci, and M. D. Cooper. 1983. "Monoclonal Antibody Characterization of Surface Antigens in Childhood T- Cell Lymphoid Malignancies." *Blood* 61 (5): 830–37.

Rowley, J. D. 1973. "Identificaton of a Translocation with Quinacrine Fluorescence in a Patient with Acute Leukemia." *Annales De Genetique* 16 (2): 109–12.

Sánchez, Alex. n.d. "Finding Patterns in Genes/Samples: Clustering Methods for Class Discovery." Statistics Department. University of Barcelona. http://www.ub.edu/stat/docencia/bioinformatica/microarrays/ADM/slides/Clustering%20microarray%20data.pdf.

Schlossman, S. F., L. Chess, R. E. Humphreys, and J. L. Strominger. 1976. "Distribution of Ia-like Molecules on the Surface of Normal and Leukemic Human Cells." *Proceedings of the National Academy of Sciences* 73 (4): 1288–92.

Shurtleff, S. A., A. Buijs, F. G. Behm, J. E. Rubnitz, S. C. Raimondi, M. L. Hancock, G. C. Chan, C. H. Pui, G. Grosveld, and J. R. Downing. 1995. "TEL/AML1 Fusion Resulting from a Cryptic t(12;21) Is the Most Common Genetic Lesion in Pediatric ALL and Defines a Subgroup of Patients with an Excellent Prognosis." *Leukemia* 9 (12): 1985–89.

Tsukimoto, Ichiro, Kwan Y. Wong, and Beatrice C. Lampkin. 1976. "Surface Markers and Prognostic Factors in Acute Lymphoblastic Leukemia." *New England Journal of Medicine* 294 (5): 245–48. doi:10.1056/NEJM197601292940503.

The Gene Ontology Consortium, Ashburner M, Ball CA, et al. Gene Ontology: tool for the unification of biology. Nature genetics. 2000;25(1):25-29. doi:10.1038/75556.
Overview. (n.d.). A Primer on Mapping Class Groups, 1-14. doi:10.1515/9781400839049-003

Garrett-Mayer, E. (2007). 18. Clustering Gene Expression Data. Data Clustering: Theory, Algorithms, and Applications, 323-340. doi:10.1137/1.9780898718348.ch18

Ronan, T., Qi, Z., & Naegle, K. M. (2016). Avoiding common pitfalls when clustering biological data. Science Signaling, 9(432). doi:10.1126/scisignal.aad1932

Yao, F., Coquery, J., & Cao, K. L. (2012). Independent Principal Component Analysis for biologically meaningful dimension reduction of large biological data sets. BMC Bioinformatics, 13(1), 24. doi:10.1186/1471-2105-13-24

Cluster Analysis in R - Unsupervised machine learning. (n.d.). Retrieved May 01, 2017, from http://www.sthda.com/english/wiki/cluster-analysis-in-r-unsupervised-machine-learning

Kumar, A. (2017, February 10). Implementing K-means Clustering on Bank Data Using R. Retrieved May 01, 2017, from https://www.edureka.co/blog/clustering-on-bank-data-using-r/

Au, W., Chan, K., Wong, A., & Wang, Y. (2005). Attribute Clustering for Grouping, Selection, and Classification of Gene Expression Data. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2(2), 83-101. doi:10.1109/tcbb.2005.17

Desheng Huang, Wei Pan; Incorporating biological knowledge into distance-based clustering analysis of microarray gene expression data. Bioinformatics 2006; 22 (10): 1259-1268. doi: 10.1093/bioinformatics/btl065