Class Discovery and Class Prediction by Gene Expression Monitoring

Depeng Xu, Xiahan Tang, James Willbanks and Md Kamrul Hasan Khan

Introduction: We analyzed gene expression data, retrieved from a paper by Golub et al. (1999), using unsupervised and supervised methods to find possible ways to classify if a leukemia patient has acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML). This data consists of 47 cases of ALL and 25 cases of AML which is divided into 38 training and 34 testing points. The presence of ALL and AML is represented by 0 and 1 respectively. Each observation contains 7129 human genes. Dudoit et al. (2002) described how to preprocess this data. They did it in four steps:

- **Thresholding:** floor of 100 and ceiling of 16,000
- **Filtering:** exclusion of genes with max/min ≤ 5 or (max – min) ≤ 500
- **Transformation:** base 10 logarithmic transformation
- **Standardization:** standardization is done in a way so that the observations have mean 0 and variance 1 across the variables (genes)

After processing this data using these four steps, we have found 3051 important genes. Since, the number of genes is large yet, we use Wilcoxon Rank Sum and Signed Rank Tests to find genes which has different effect on two groups of response. At 5% level of significance we have found 263 important genes. Then we use some unsupervised and supervised methods on this data.

Unsupervised Methods:

K-means

We began our analysis using an unsupervised method. The k-means algorithm is popular due to its fast time complexity and relative accuracy for clusters of similar shape and size. Using only the training data, we ran the k-means algorithm initially for 2 clusters. For comparison, the samples that were classified as "ALL" have been given a value of 0 and the samples that were classified as "AML" have been given a value of 1. This makes a side by side comparison simpler. The clustering results and actual classifications for each of the 38 samples is as follows:

```
1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 2 2 2 2 2 2 2 2 2 2 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1
```

It is clear that the 1st cluster that k-means found is representative of the "ALL" group, and the 2nd cluster is representative of the "AML" group. Only two samples were misclassified, specifically the 12th and 25th samples. Since this clustering algorithm is using all 3051, it is possible that these two samples were misclassified because of some strange shape in very high dimension. So, we decided to run the k-means algorithm again using 3 clusters to see if this improved the model. The results are as follows:

```
1 3 3 1 1 3 1 1 3 3 3 2 1 3 1 1 1 1 1 1 1 1 3 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1
```

Now, the 1st cluster has been broken into two smaller clusters, and the 2nd cluster has remained the same. This has allowed us to correctly assign one of the errant assignments, namely the 25th sample, from the first attempt at this method. It is worth noting that the algorithm did not correctly assign the 25th sample by putting it in its own cluster, but has in fact clustered it with several other points of the same classification. With the success of this increase of clusters, we decided to increase the clusters to four to see if we get another similar improvement in the model. The results are as follows:

```
2 4 4 2 3 4 2 2 4 4 4 2 3 4 3 3 3 2 3 3 3 2 4 3 2 3 2 1 1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1
```

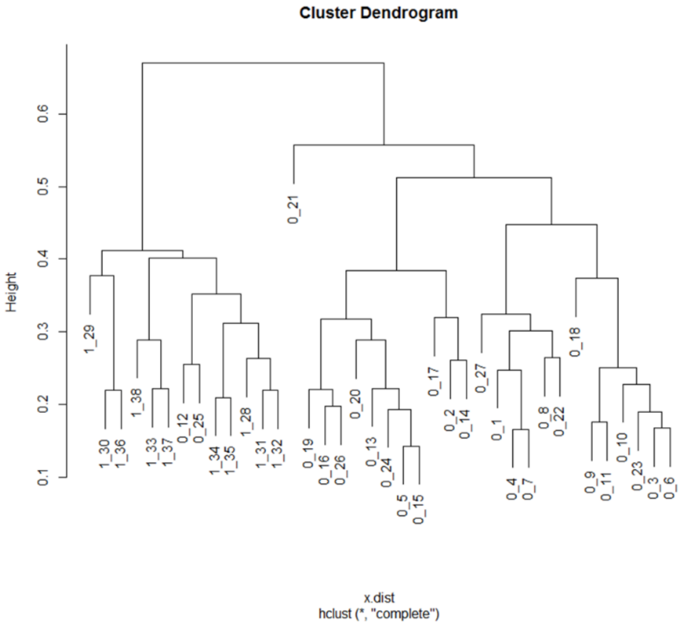
Once again, we see that the samples not classified into the “AML” cluster have been split into more clusters. However, we now have no samples that are misclassified. We take what are now labeled clusters 2, 3, and 4 to be samples classified as “ALL,” and the cluster labeled 1 to be the samples classified as “AML.” Once again, the new clusters all have several members. This is a good indication that the algorithm isn’t just overfitting the single errant values from previous attempts, but that the 4 clusters all have several members that more efficiently model the data. The clusters can be summarized in the following tables:

K = 2	AML	ALL	K = 3	AML	ALL	K = 4	AML	ALL
1	25	0	1	18	0	1	0	11
2	2	11	2	1	11	2	9	0
			3	8	0	3	10	0
						4	8	0

We have previously mentioned that the error in the first two models could be due to some odd shape in high dimension. However, k-means also typically works best for clusters of similar size. Our training data has one cluster of size 11 and another of size 27. This may be the cause of the error in the first model. As more clusters are added, the size of the clusters becomes more uniform. This may be the cause of our eventual perfect modeling.

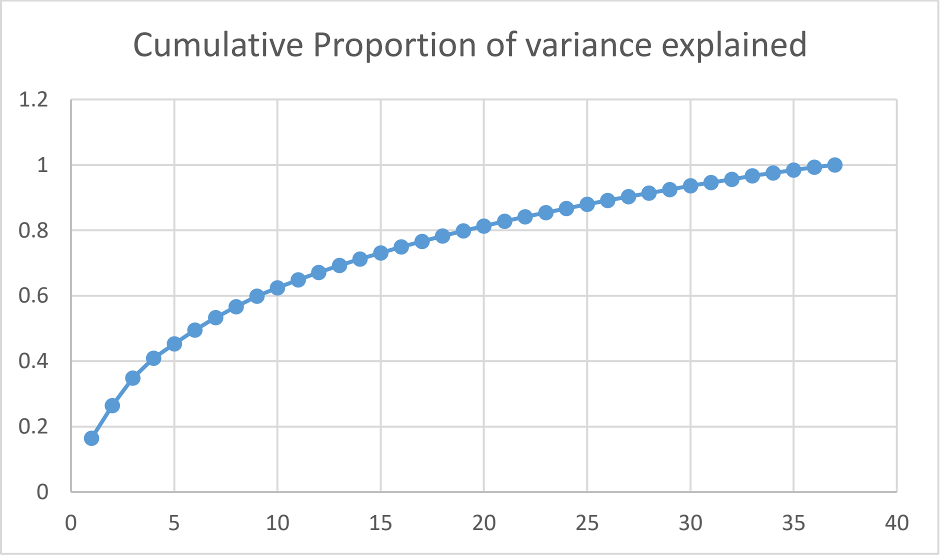
Hierarchical Clustering

First Hierarchical method was used, instead of using Euclidean distance to measure the correlation between two observations, Spearman’s rank correlation was implemented. The hierarchical diagram is provided in Figure 1. We see that the 38 observations were classified into 2 groups (group 1 and group 0). Most of the 38 observations were classified correctly except 2. Observation 12 and observation 25 were supposed to be in cluster 0 but instead they were in 1. This result is consistent with using k-means with k=4, where also these 2 observations were classified into the wrong cluster.



PCA:

Next Principle Component Analysis is implemented. The result is plotted in Figure 2. We see that there is no obvious transition point at which the variance explained has a sudden increase. We also see that when we choose to have 5 principle components, less than half of the total variance is explained. Therefore, there is no need to look further because we can conclude that PCA is not an efficient method for classification in this case.



Supervised Methods:

K nearest neighbor

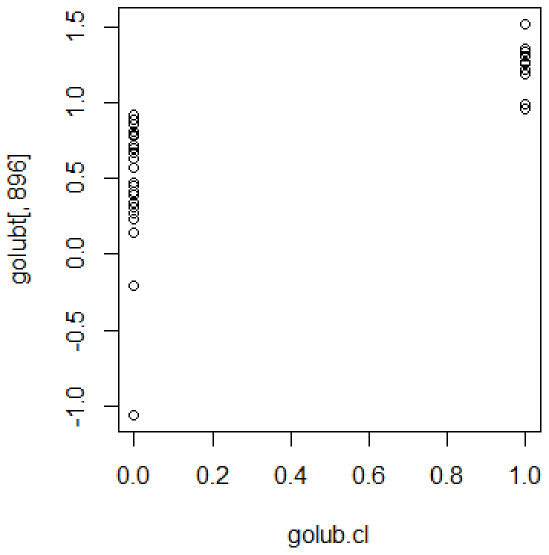
We next moved on to the k-nearest neighbor algorithm. We used a leave out one cross validation algorithm on the training data. This uses the true classifications of the data to determine the clusters. The algorithm trains a model, leaving one out, and then classifies the one left out. The reported numbers are how many were classified into each cluster based on the number k of nearest neighbors used in the algorithm. We report k values from 3 to 10 as values outside of this range are prone to error:

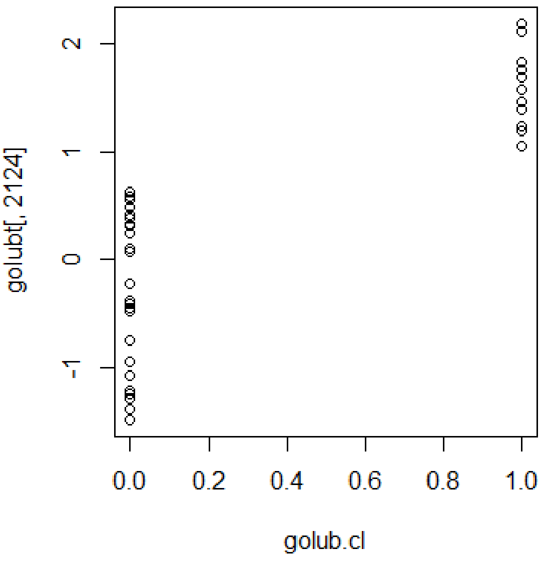
	K = 3	K = 4	K = 5	K = 6	K = 7	K = 8	K = 9	K = 10
AML	26	26	26	27	27	29	27	27
ALL	12	12	12	11	11	9	11	11

The k nearest neighbor algorithm classifies all of the points correctly for k equal to 6, 7, 8, and 10. It is worth noting that the point that is misclassified when k is equal to 3, 4, and 5 is the same 12th sample that is misclassified during the k-means algorithm. Once again, we are able to perfectly model the sample data, and it appears that the 12th sample may have some significant differences from the other samples that are classified as AML.

Decision Trees

We next decided to look at some decision trees to see if there are any genes that may be driving these clustering algorithms. If there are any genes that have close values that split between the two classifications, then these are likely to be the largest contributors for both the shortest Euclidean distance used in k-means and the nearest neighbors. We found two genes that split perfectly between the two classifications. Their gene levels are graphed by their classification:





Ridge, Lasso & Elastic Net:

Ridge Regression

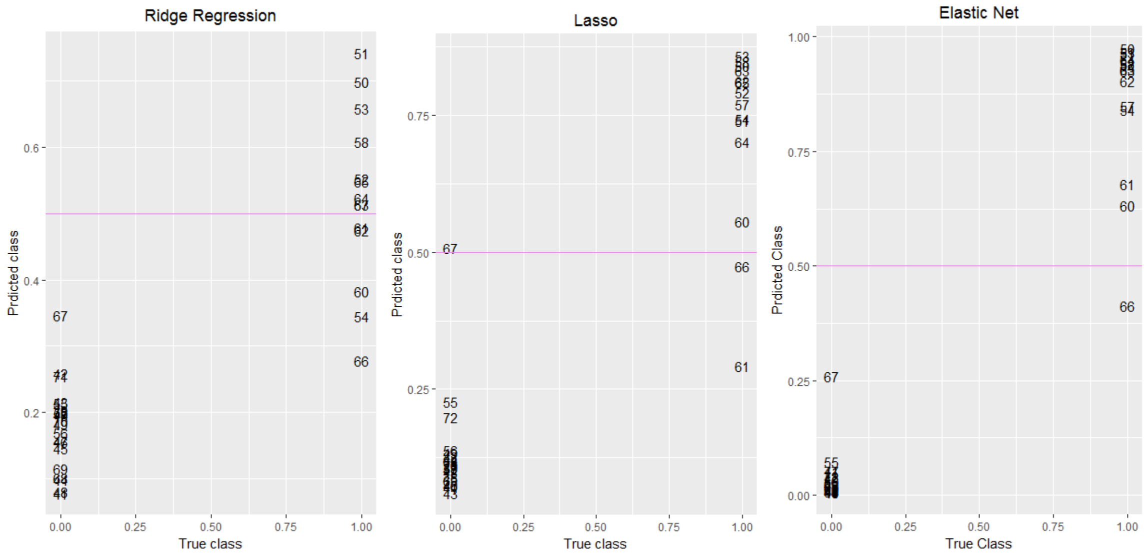
		True Class	
		0	1
Predicted Class	0	20	5
	1	0	9

Lasso

		True Class	
		0	1
Predicted Class	0	19	2
	1	1	12

Elastic Net

		True Class	
		ALL	AML
Predicted Class	ALL	20	1
	AML	0	13

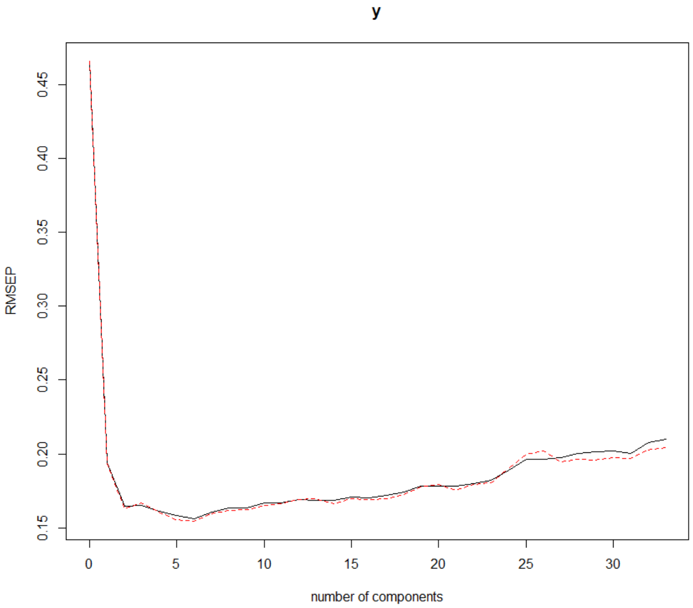


	Number of non-zero β
Ridge	263
Lasso	8
Elastic Net	44

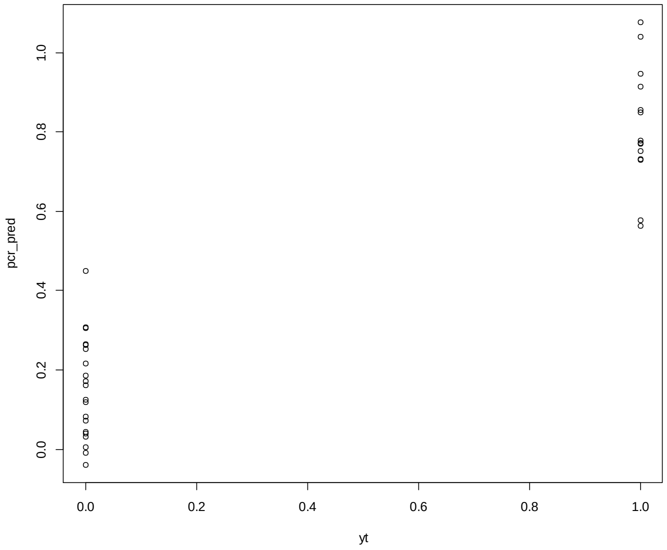
Interpretation: For a grid of α in Elastic Net we find best α = 0.6666667 and for that α the best λ = 0.02483311. The proportion of misclassification is 2.94%. On the other hand, in Ridge regression with best λ = 17.34371 the proportion of misclassification is 14.7% and in Lasso regression with best λ = 0.1281825 the proportion of misclassification is 8.8%. Therefore, we can conclude that Elastic Net is best here.

PCR:

Another supervised method we used to classify AML and ALL is Principal component regression (PCR). We used pls package in R to perform PCR. We first ran PCR on Training dataset with cross validation. We used rooted mean square error of prediction (RMSEP) as criteria in cross validation. The cross-validation plot is shown below:



From the cross-validation plot, we can see PCR has the lowest training error when the number of principal components is 6. Using 6 PCs can explain 67.37 % variance in x and 90.84 % variance in y. We use the PCR model with 6 PCs to predict class label for testing dataset. The prediction result is shown below:

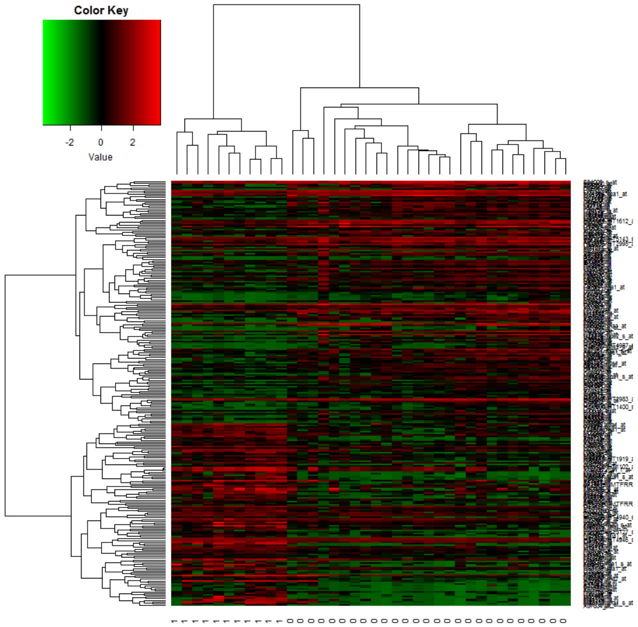


From the plot, we can see the 0s and 1s can be clearly separated by 0.5 threshold. This 6 PCs model can predict all 20 ALL samples as 0 and all 14 AML samples as 1. It has 100% accuracy, which is higher than other supervised methods.

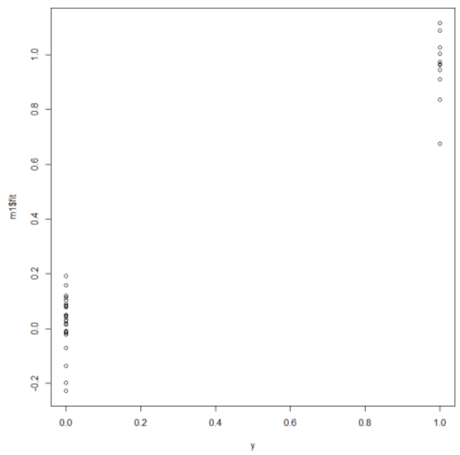
Others:

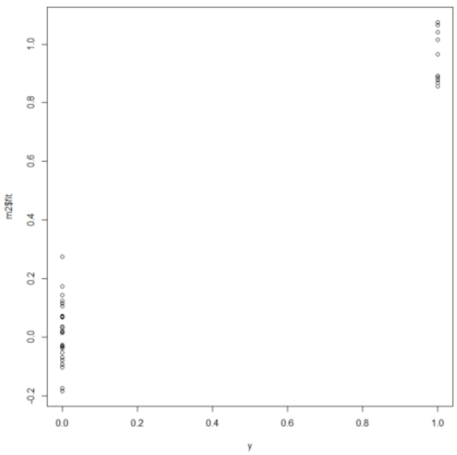
Approximate best set variable selection:

We also explored how to do approximate best set variable selection. For example, we want to select a best 20 variable set for regression. The computational complexity is 20^N (N=#of genes to select from), which is impossible to do. One naïve way to do the selection of 20 genes is to select the best 20 genes based on lowest p-values of one-covariate regressions. This has a computation time of only O(N). The problem of this is that it doesn't consider correlation among genes. However, if we look at the heatmap of gene expressions. We can see genes are clustered into different groups. Each group has genes in a similar expression pattern. For example, the top half clusters have low expression in AML but high expression in ALL, whereas the bottom half clusters have high expression in AML but low expression in ALL. So we can divide all genes into 20 clusters (based on hierarchical clustering), then select 1 characteristic gene from each group. The computation time of selection is also O(N). But this time, we can assume the characteristic gene can representing all other genes in its gene group, and they are less correlated to each other because they are clustered into different groups at the beginning based on gene expression profile.



We can build two regression models using two approximate best set variable selection methods. We can see from the results below: on the left it is the naïve top 20 genes selection, the right is characteristic genes of 20 gene groups. Even though both of them has 100% accuracy on prediction, the characteristic genes may have higher prediction power (as the points are more clustered together) and have more biological meanings.

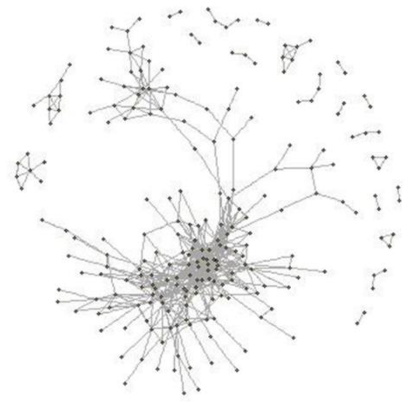




Network Analysis:

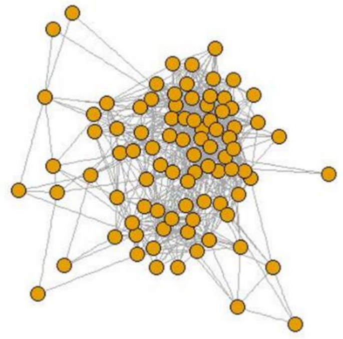
We also tried network analysis on this dataset. We use 47 ALL samples and 25 AML samples to comstructe gene expression networks respecctively. The network is constructed based on Random matrix theory (RMT)- based approach. Each node in the network representing a gene, and the link between two nodes means the two genes have related gene expressions (positively or negatively). The networks are shown below:

acute lymphoblastic leukemia (ALL)



Cutoff=0.68, #nodes=234, #links=640

acute myeloid leukemia (AML)



Cutoff=0.56, #nodes=88, #links=640

We can see from the networks that: The ALL network has a more complicated structure and more genes related to each other. The AML network is more compact and less genes linked together. From the basic topological properties, we can explain the result we founded earlier in k-means. In k-means, the ALL samples can only be fully captured by the model after k=4. It takes at least 3 different clusters of samples to describe ALL class. The reason may be the gene expression patterns is more complicated in ALL network. There are more genes linked in the network, and there are more than 1 modules in the network.