COSC 681 Bioinformatics

Project 3 Gene Expression Analysis

2016/12/15

William Gillespie

**Part 1: Supervised Classification**

1. **Apply holdout method to divide samples into training set and testing set.**

I chose 3 different holdouts for the train/test parts: 50/50, 65/35, 75/25. I chose these because they each give a decent amount of training data while still having at lease 10 samples in the test data.

1. **Apply SAM to select informative genes**

I used the MeV software to select 78 informative genes. I chose 78 because that was the lowest number above 75 that I could select with the slider bar. The requirements were to select around 25-75. I wanted to select as many as possible while still being around that range.

1. **Use KNN classification methods; repeat 3 times with different holdout**

I used the standard KNN classifier in Matlab. The code for this are in the Part1\_Supervised\_Classification directory under KNN\_Pearson and KNN\_Euclidean. I performed two distance metrics with the three holdouts totaling 6 different classification runs. The distances I chose were ‘correlation’ and ‘Euclidean’.

1. **Summarize your analyzing results – at lease errors and consistency**

I will list the error percentages in the following table:

|  |  |  |
| --- | --- | --- |
|  | Error Percentages for KNN |  |
|  | **Distance: Pearson** | **Distance: Euclidean** |
| Holdout 50/50 | 16.13 % | 19.35 % |
| Holdout 65/35 | 18.18 % | 18.18 % |
| Holdout 75/25 | 18.75 % | 18.75 % |

For the most part, these are fairly low error rates. I’m not sure what the optimal error rates are, but I think this is quite remarkable given that the data started with 2000 features.

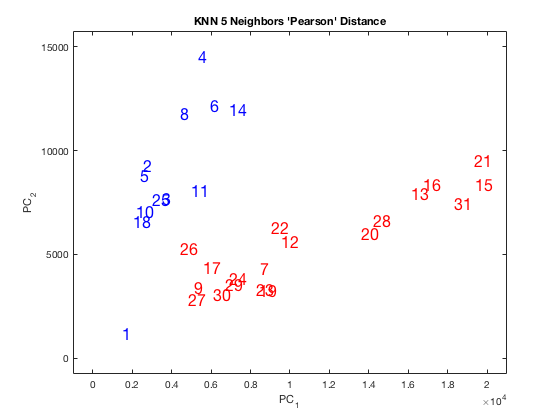
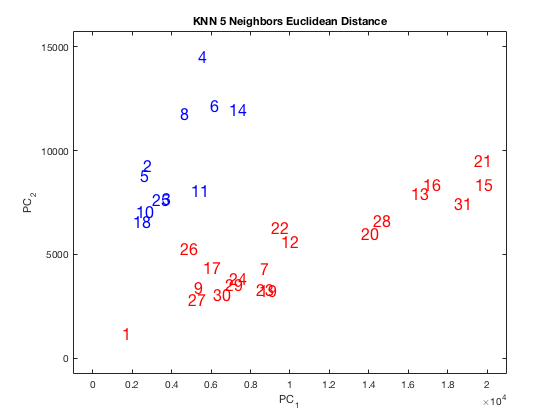
One can see that for the holdouts of 65/35 and 75/25, the Euclidean and Pearson distance measurements give the same error percentage. For the holdout of 50/50, Pearson gets a better error percentage. Such a difference is not that significant because it is due to one or two record misclassifications. With such a small test set, differences of the error percentages that small are not significant.

One thing that is strange is that for the Pearson distances, the error percentage goes up as the training data grows. This should not happen. If anything, the error percentage should go down as the training data grows. This is no doubt a symptom of the small test set. If the small test set has some data points that are hard to classify, then that will bump up the error percentage.

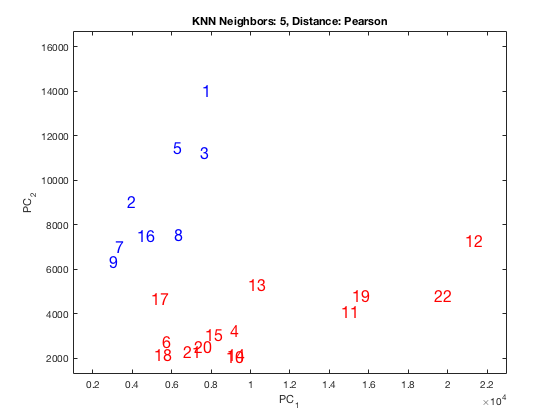
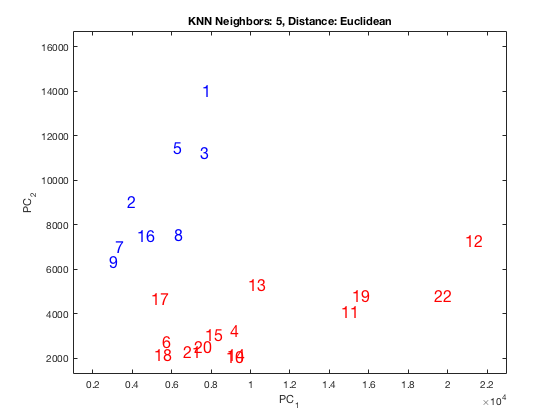
1. **Display at least tree view and colored scatter plot**

**Scatter plots (Euclidean on first column, Pearson on second column):**

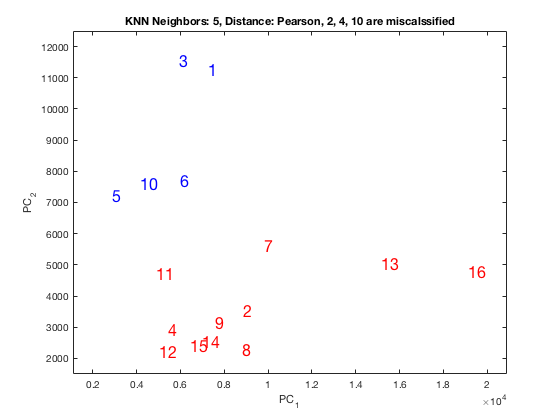
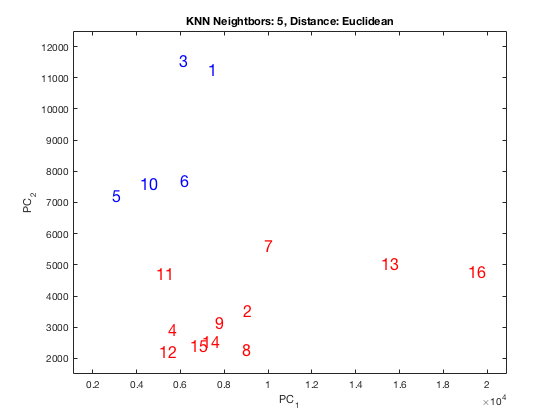
Holdout 50/50

****

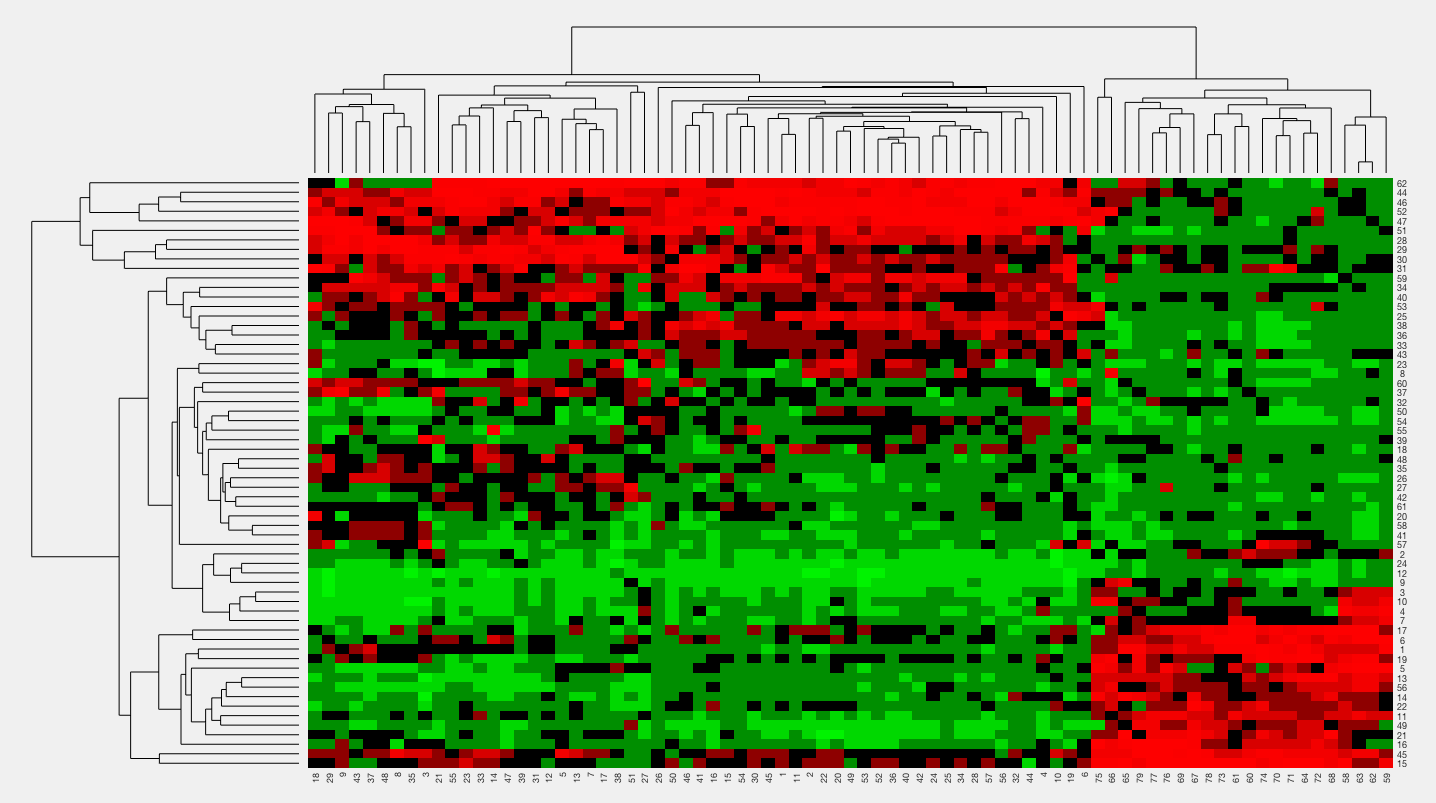
Holdout 65/35

****

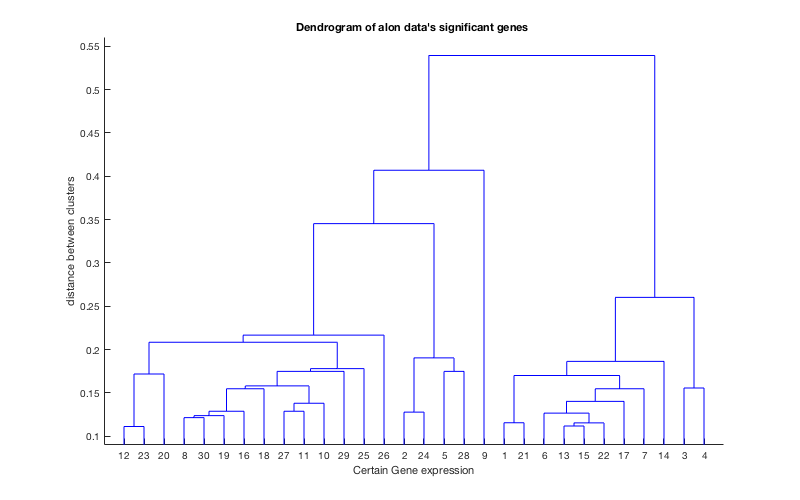
Holdout 75/25

****

**Tree View**



Dendrogram (Just for fun)



**Part 2: Unsupervised Clustering**

1. **Perform HC and compare to the supervised results. (use 3 linkages, 2 distances)**

I performed hierarchical clustering with 6 combinations of linkages and distances. I then compared the cluster labels I got with the clustering, compared them to the official labels of the data set, and calculated the errors. Below is a table summarizing my results.

|  |  |  |
| --- | --- | --- |
| Error Percentages of HC: cluster labels compared to real labels | | |
|  | Distance: Pearson | Distance: Euclidean |
| Linkage: Single | 32.26% | 33.87% |
| Linkage: Complete | 16.13% | 33.87% |
| Linkage: Average | 16.13% | 43.55% |

1. **Compare the six experiments and briefly address which ones are possibly superior than others and why.**

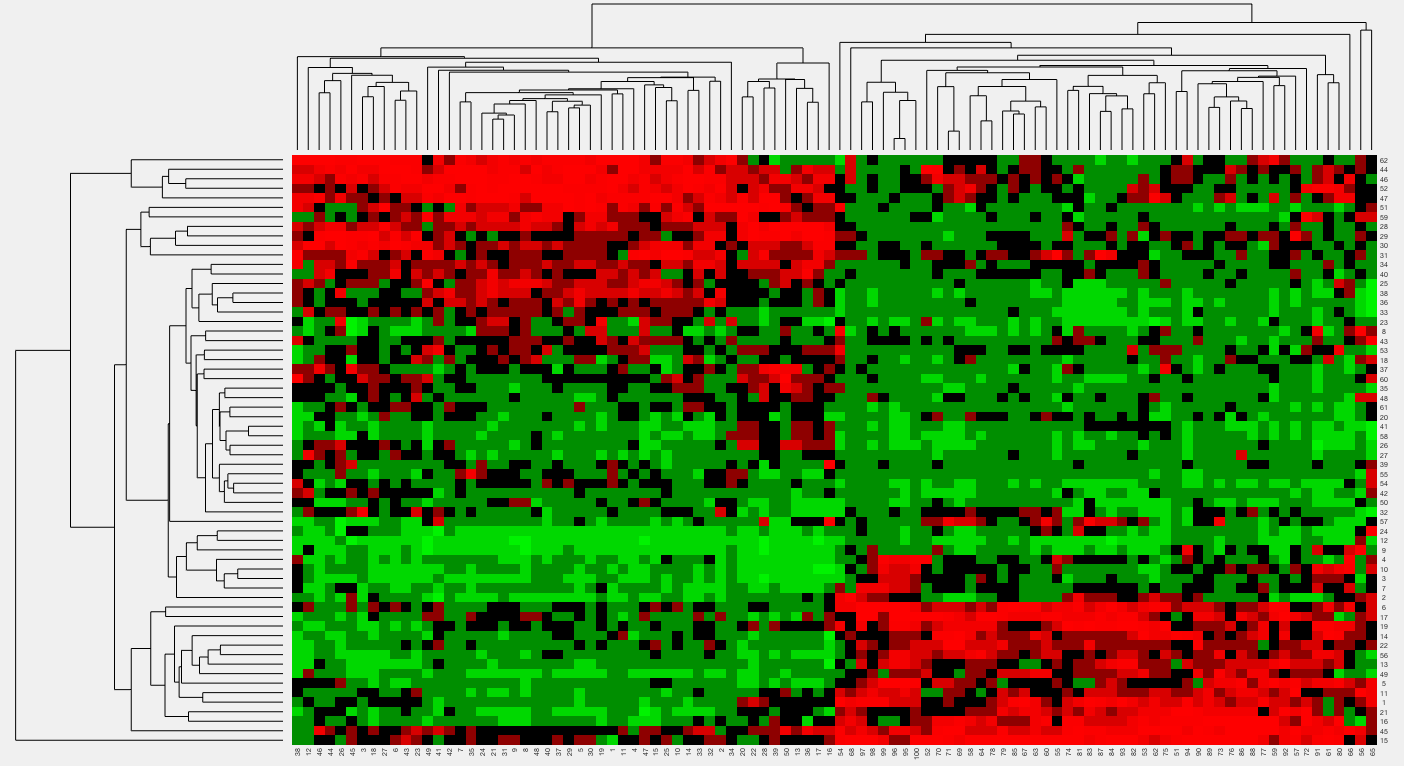
From the table, we can see that two of the combinations did significantly better than the other ones: Complete and Average Linkage with Pearson Distance. Single linkage and Euclidean distance performed poorly. I can see why single linkage performed poorly. Its nature of merging two clusters based on the two closest elements tends to suck up one point at a time and merge it with a giant cluster. This gives the end result of one enormous cluster and one tiny cluster. Complete and average linking tend to build up smaller clusters before merging the smaller ones into larger ones. This can be seen in the dendrograms below.

The strange thing is that the Euclidean distance metric performed so poorly. I am not sure why Pearson distance performed so much better than Euclidean distance. **ADD MOAR**

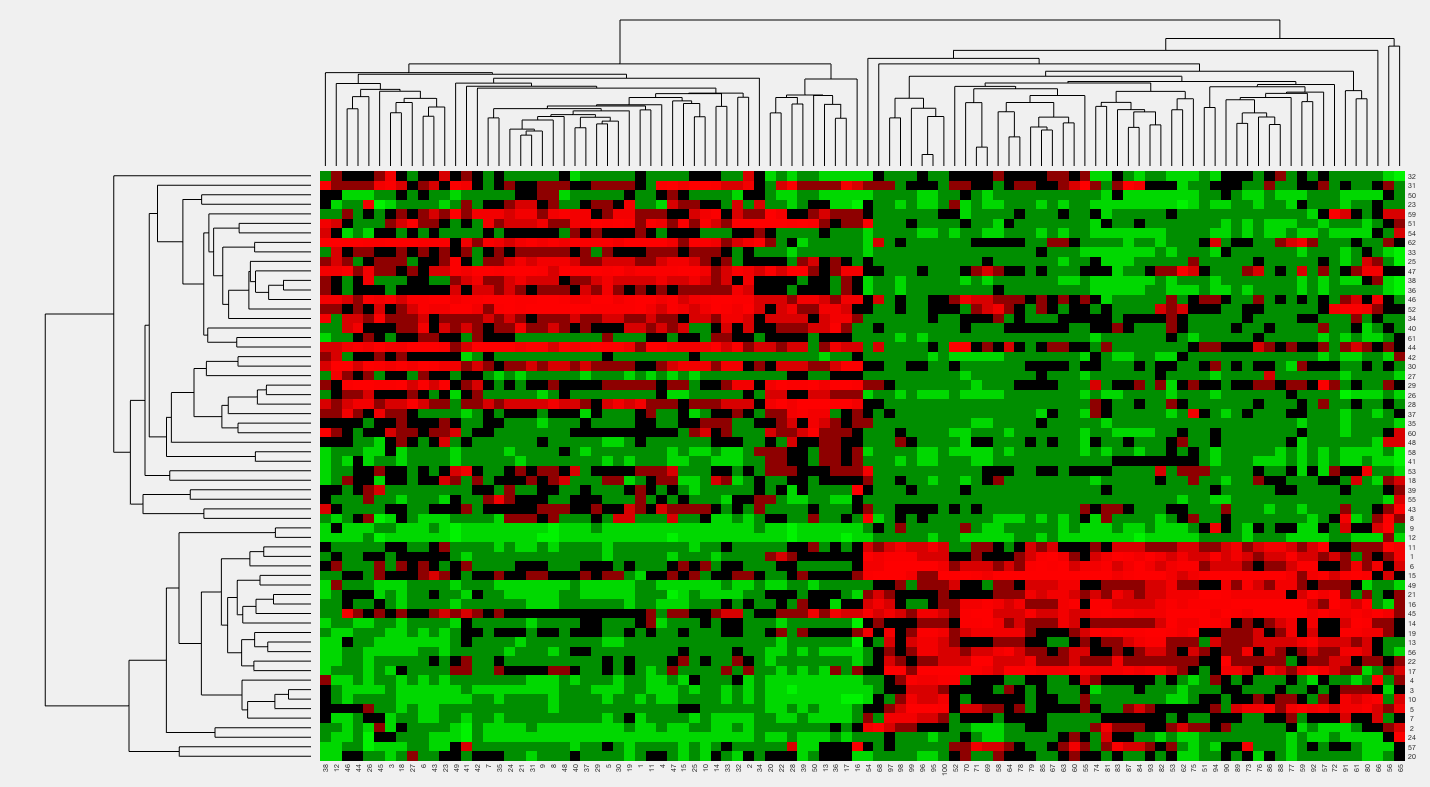
1. **Display the tree views and colored scatter plots**

Below I show all of the tree views:

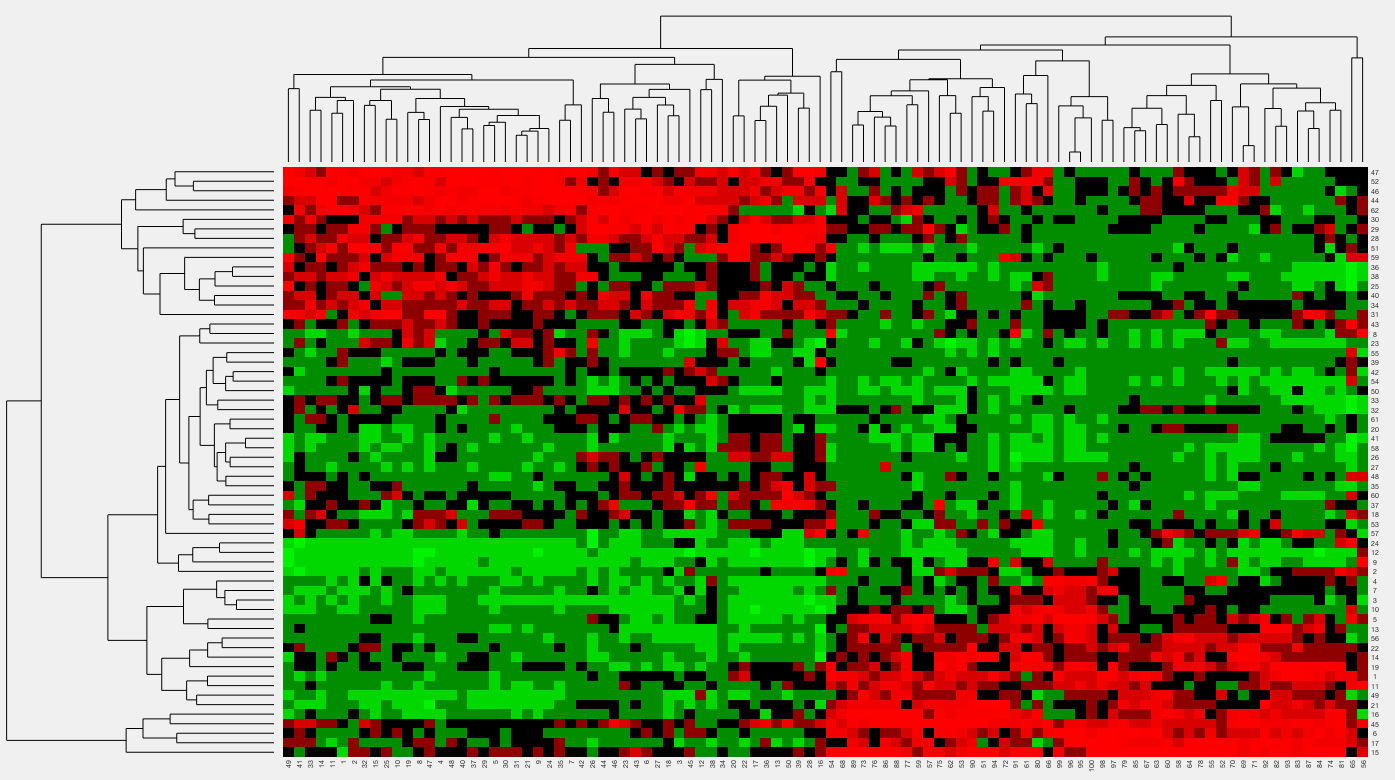
Distance: Euclidean, Linkage: Average



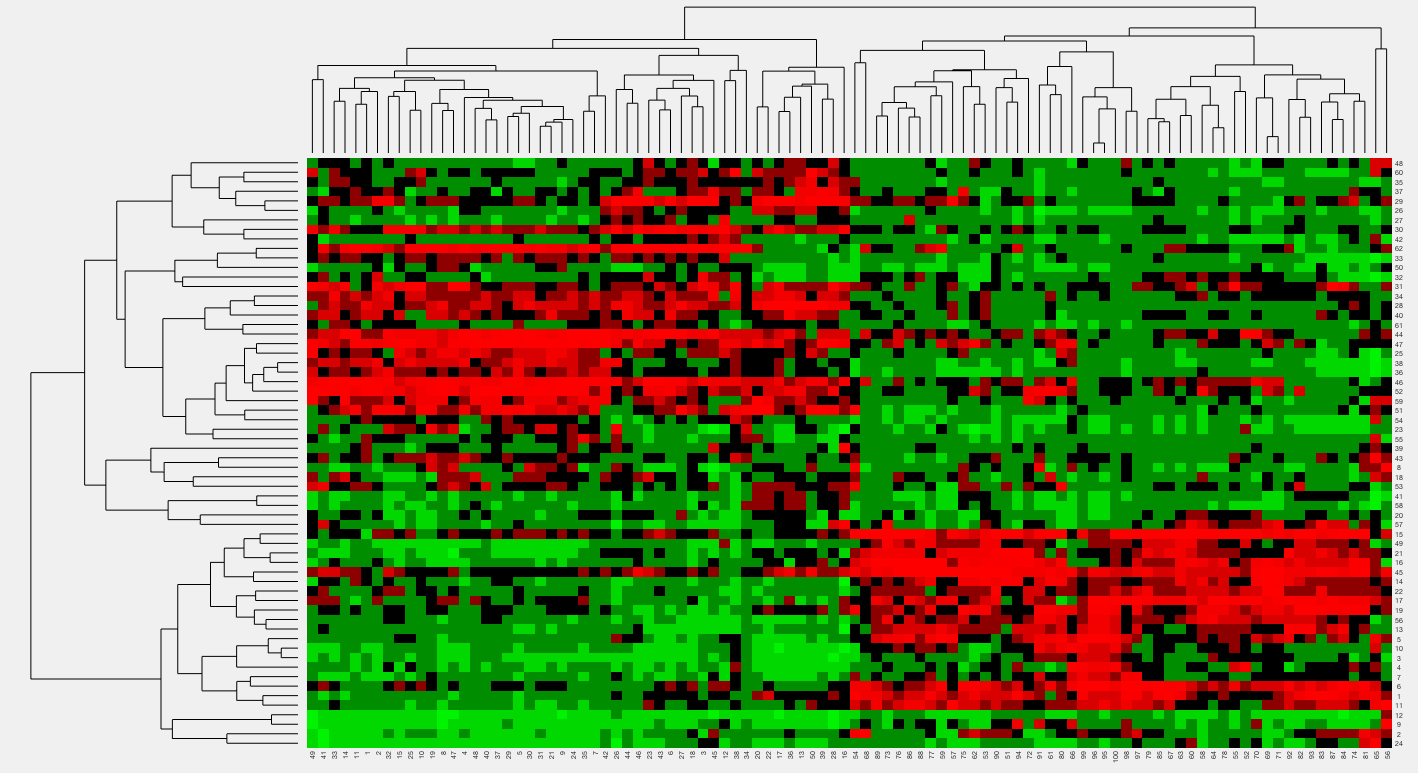
Distance: Pearson, Linkage: Average



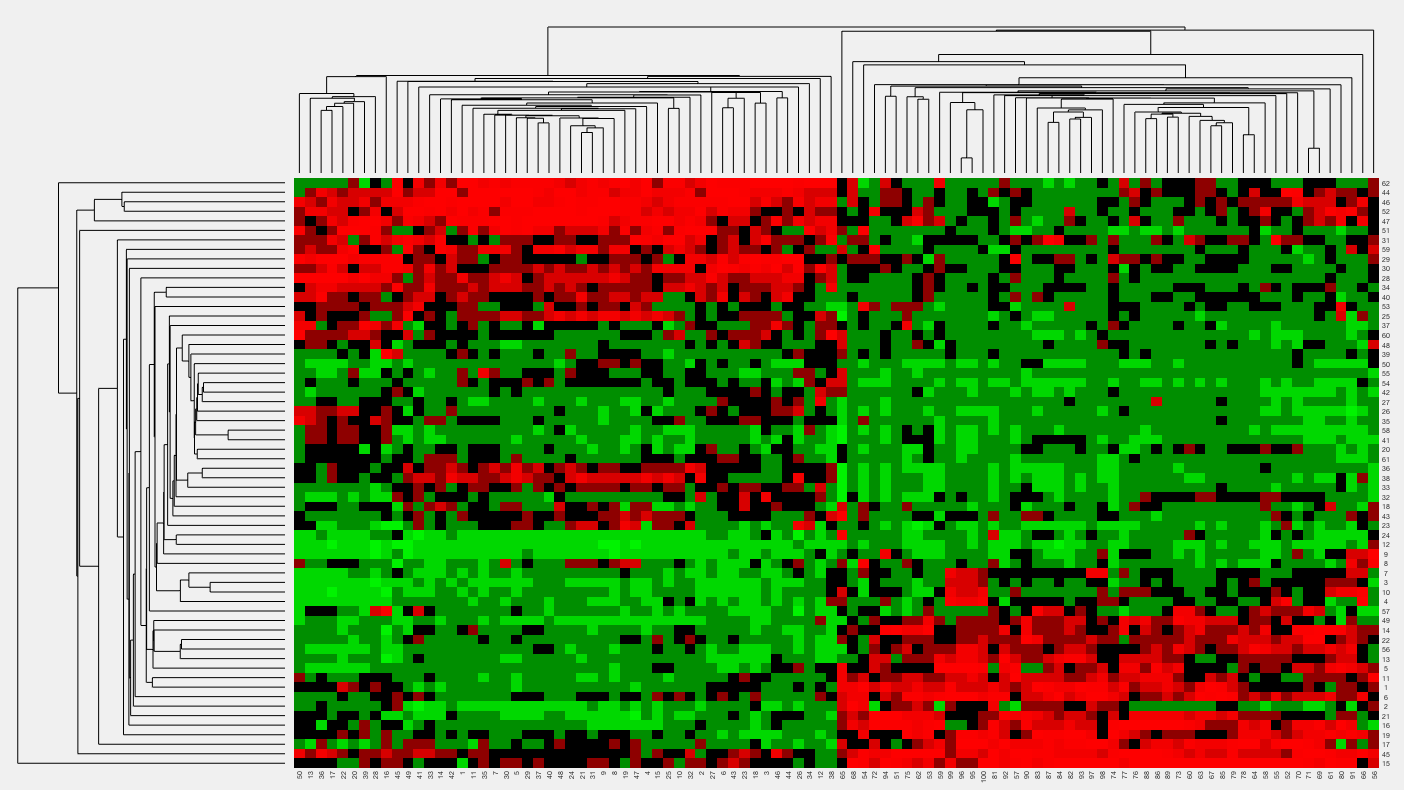
Distance: Euclidean, Linkage: Complete



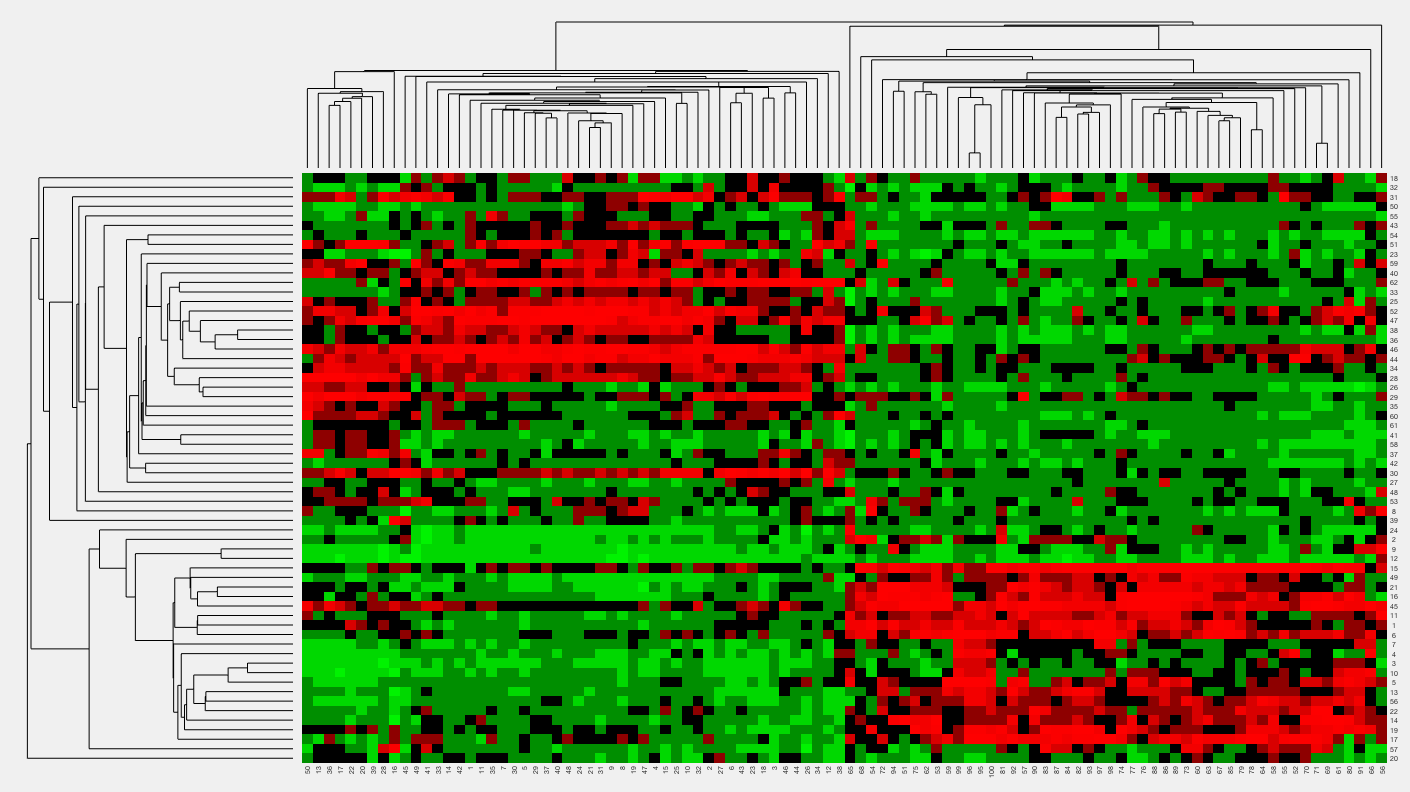
Distance: Pearson, Linkage: Complete



Distance: Euclidean, Linkage: Single

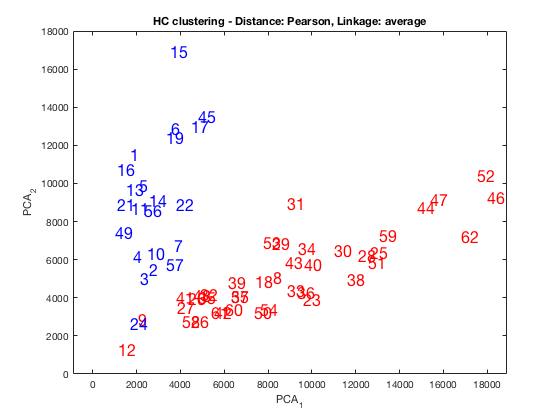
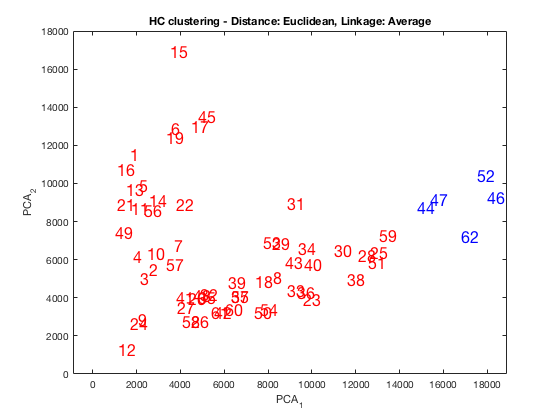


Distance: Pearson, Linkage: Single

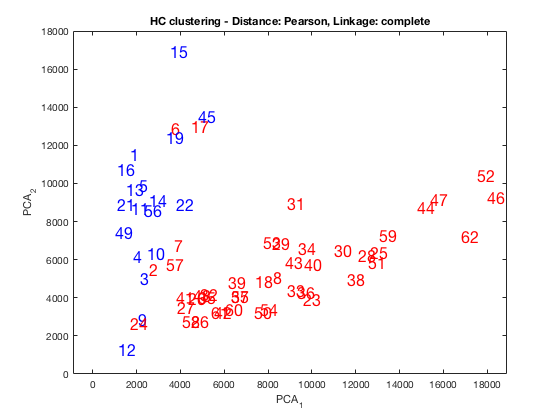
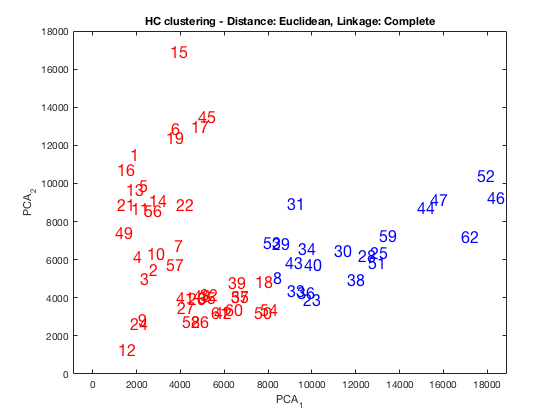


Below I display all of the scatter plots:

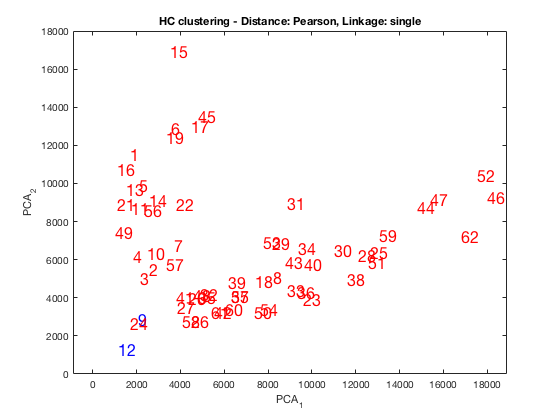
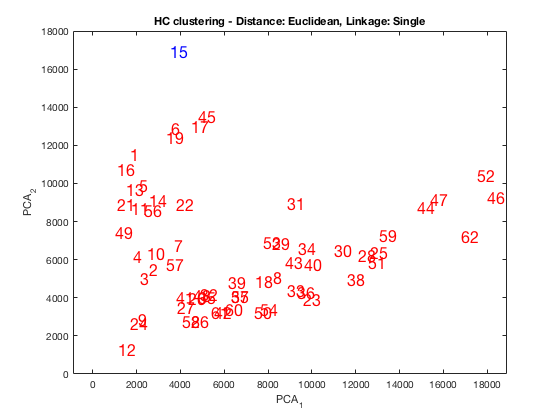
Distance: Euclidean, Linkage: Average / Distance Pearson, Linkage: Average



Distance: Euclidean, Linkage: Complete / Distance Pearson, Linkage: Complete



Distance: Euclidean, Linkage: Single / Distance Pearson, Linkage: Single



Take note that 22 points are healthy tissue and 40 are tumorous tissue. That means that about 1/3 of the points should be in one group and the rest of the points should be in the other.

We can make several observations from the scatter plots above. First of all, Single linkage preforms terribly. Only one or two points are in the blue group; we know that the smaller group should contain 1/3 of the points. If we look back at the error rates in the table above (section 2.1), we can see that the error rates for the single linkage were above 30%. This means that almost all of the points in the entire smaller cluster are being misclassified.

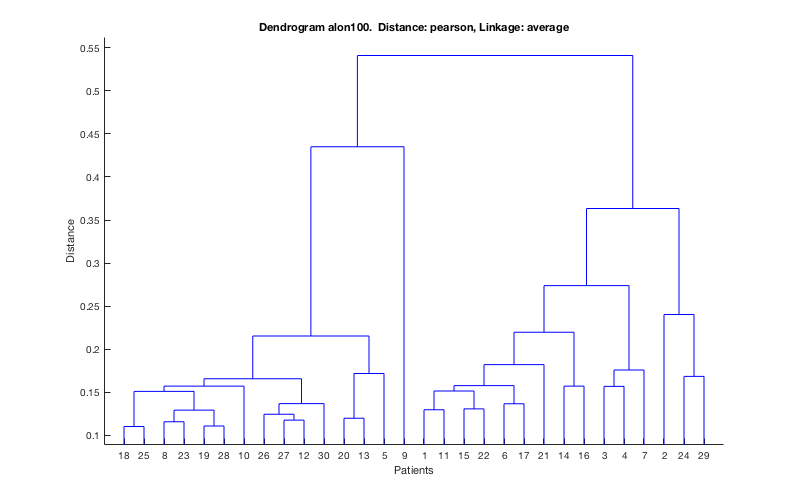
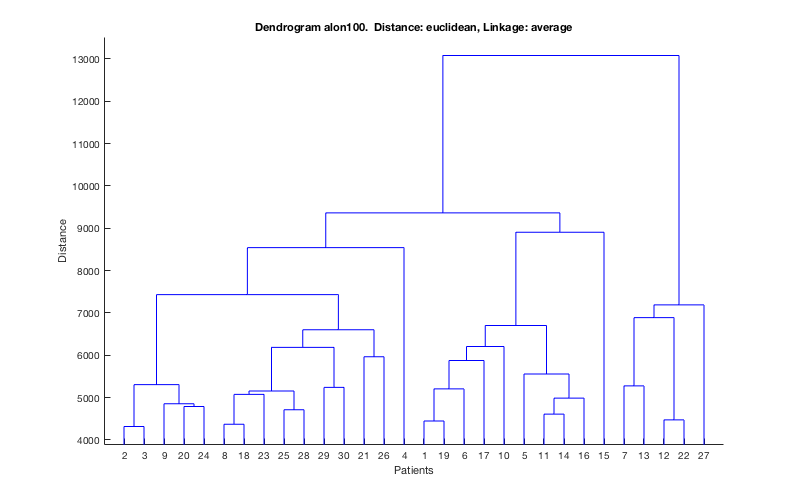
Another aspect to note is the poor performance of the Euclidean distance with average linkage. I’m not sure why this one performed so poorly, but it got the worst error rate out of everyone (43.55%).

One curiosity is the plot with Euclidean Distance and Complete Linkage. This clustering seemed to cluster the farther out members of the tumorous group together. It also clustered the normal tissue with some of the tumorous tissue. From the scatter plot, one could reason that this looks like a somewhat reasonable cluster (even though we know the real labels).

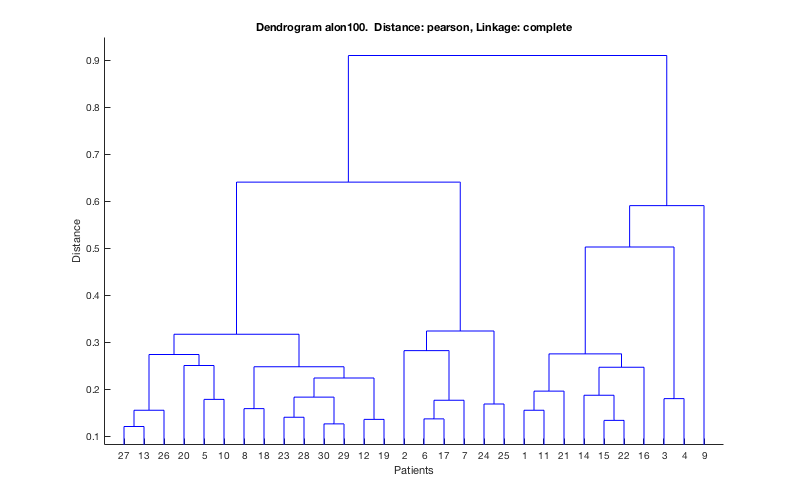
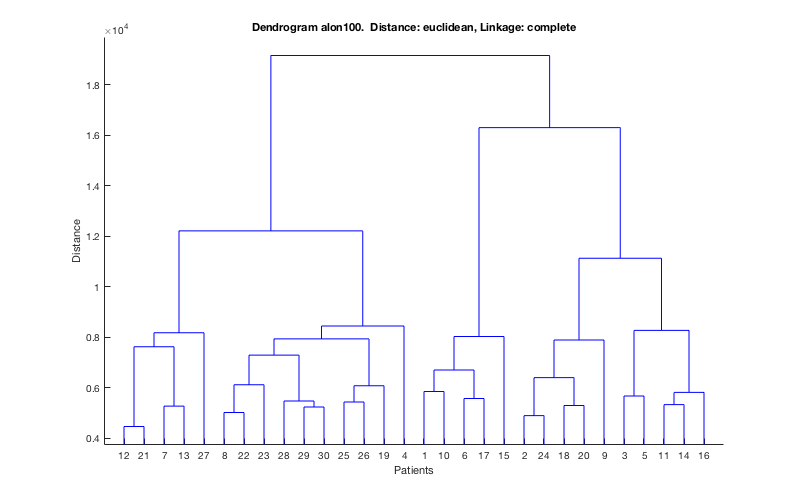
The two clustering examples that performed well (and look good) are the ones with Pearson distance and complete and average linkage. From the plots, most of the points from 1-22 are blue (which is the normal group), and most of the points from 23-62 are red (the tumorous group). The one with complete linkage has some pretty bad misclusterings (e.g. 6, 17), but it is good overall.

Below I show the dendrograms for completeness:

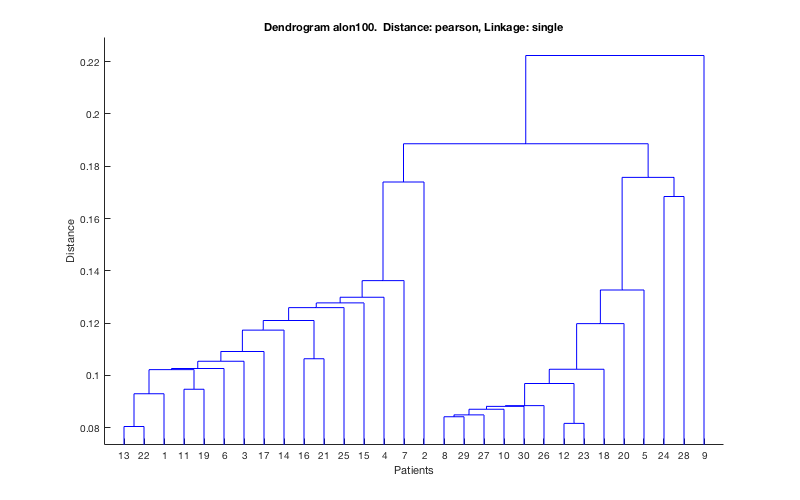
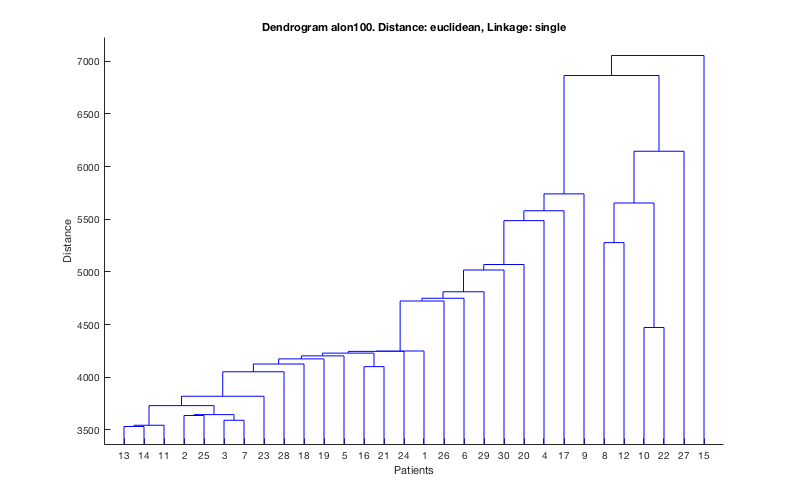
Distance: Euclidean, Linkage: Average / Distance: Pearson, Linkage: Average



Distance: Euclidean, Linkage: Complete / Distance: Pearson, Linkage: Complete



Distance: Euclidean, Linkage: Single / Distance: Pearson, Linkage: Single



The Average and Complete linkages bring larger clusters together (The complete linkage case more so). The Single linkage has the unattractive aspect of adding one element to a larger cluster. A very bad aspect is that in the final cut, one element is joined with a group of everything else. This leads to a large imbalance in items in a cluster and gives a bad outcome in our example.

1. **Perform KMC and compare the supervised results.**

I performed K means clustering with Euclidean and Pearson distances. The error rate table is shown below:

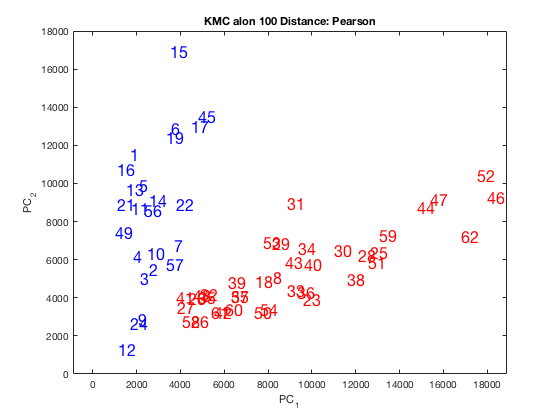
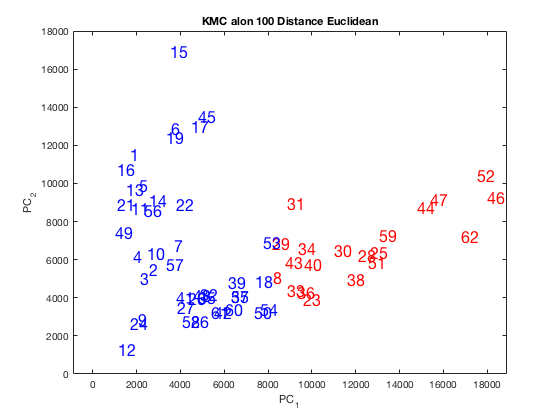
|  |  |  |
| --- | --- | --- |
|  | Euclidean | Pearson |
| Error Rate | **32.26%** | **12.90%** |

You can see that the Pearson distance outperforms the Euclidean distance again, this time in K means clustering.

1. **Display the tree view and colored scatter plots.**

I’m not sure how to make a tree view with K means clustering, but I made two scatter plots:

Distance: Euclidean Distance: Pearson



From the two scatter plots, the Pearson outperforms the Euclidean distance metric. In the Euclidean clustering, many of the points above 22 are classified as blue (along with the 1-22 points). This means that the normal tissue that was somewhat ‘close’ to the tumorous tissue was clustered with that tumorous tissue. A nearly identical event happened with the Euclidean distance with complete linkage in the scatter plots for hierarchical clustering. The Pearson distance performed quite well, and quite similarly to the hierarchical clustering examples with Pearson distance, and complete and average linkage.

1. SOM: compare with supervised results.

I performed the SOM with the newsom function in Matlab. I did not perform the other SOM method because it gave me errors. I used 100 epochs for my output. I did more epochs (even 1000), but they did not give me better results. By error rate compared to the real labels is in the table below.

|  |  |
| --- | --- |
|  | SOM |
| Error Rate | **30.65%** |

SOM did not perform as well as some of the others, but it was not the worst.

1. **Compare HC, KMC, and SOM. Make your case.**

The clusterings that performed the best were the hierarchical clustering with Pearson distance and Complete and average linkage, and the K means clustering with Pearson distance. SOM, HC with Single linkage, HC with Euclidean Distance and average or complete linkage, and KMC with Euclidean distance performed poorly in comparison.

One pattern to note that that KMC and HC performed very well with the Pearson distance metric (if we ignore the HC single linkage case).

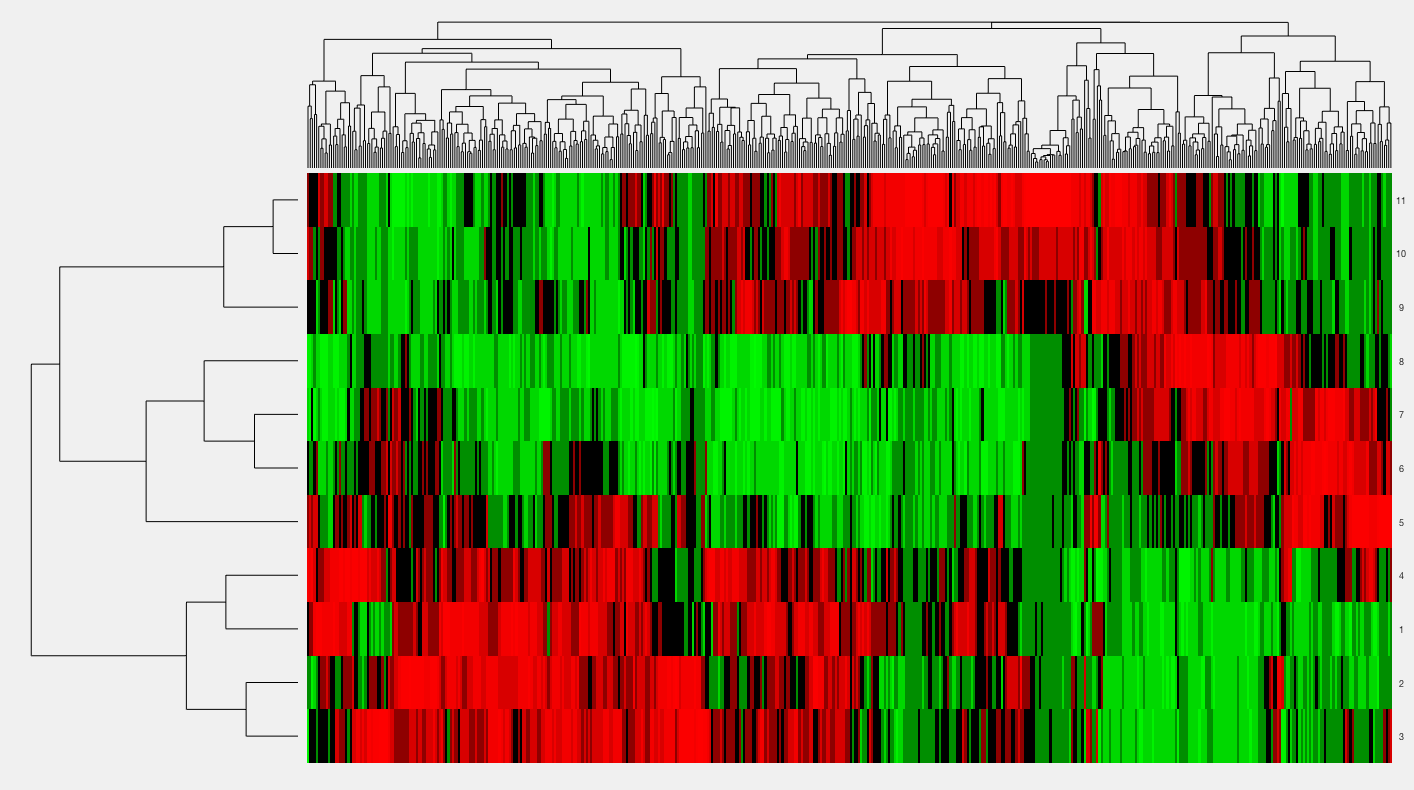
**Add MOAR**

**Part3: Gene-Based Analysis**

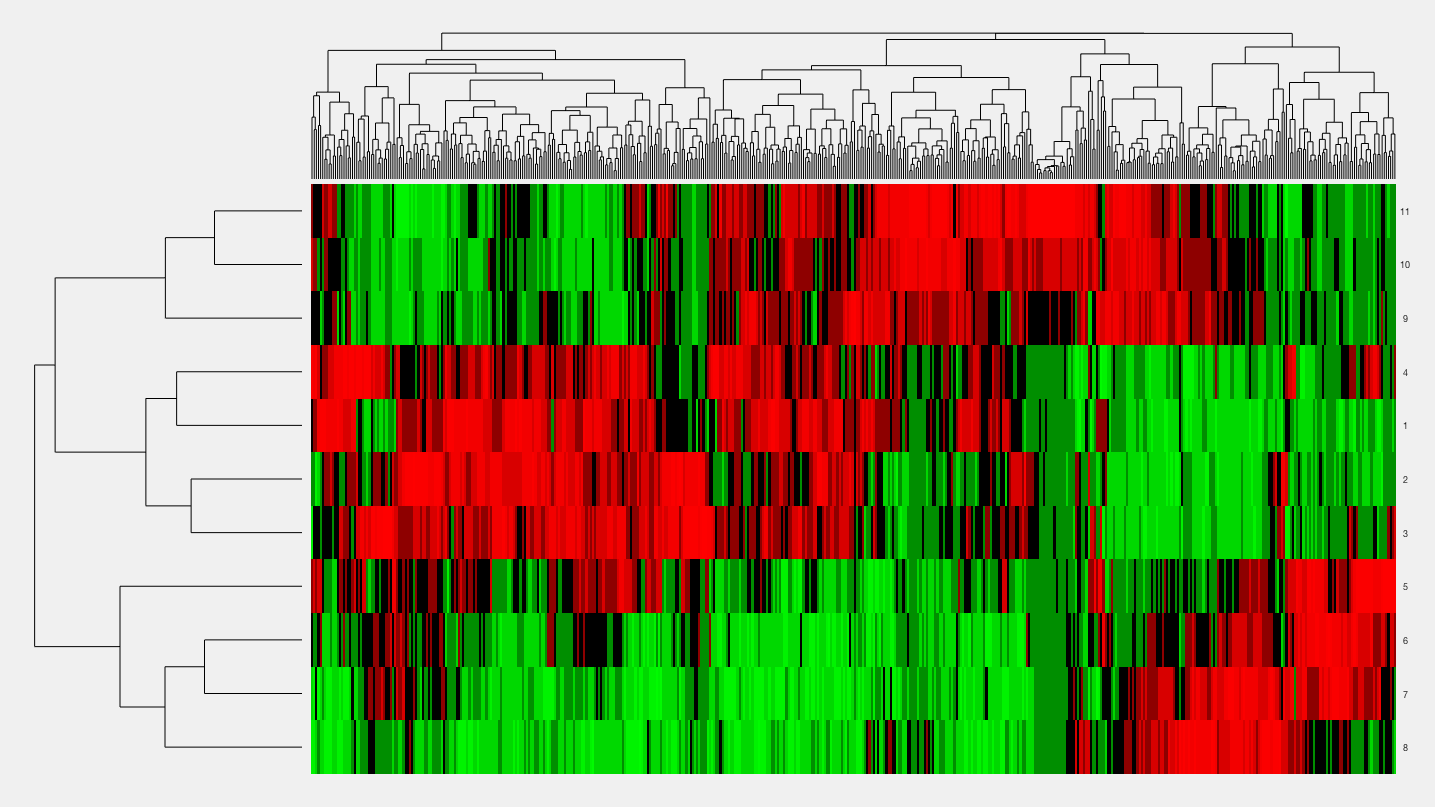
1. **Visualization Only: provide clustergrams of the entire original data set. You are free to choose any linkage or distance.**

I will choose Pearson distance and complete linkage. It gave good results in the previous section so I will choose it for this one. I will also choose Euclidean complete for a comparison.

**Pearson Complete:**

****

**Euclidean Complete:**

****