Individual Project: Microarray Based Tumor Classification

By Aishwarya Deengar

(PROGRAMMER)

Introduction

The Marisa et. al. study uses microarray data to provide a molecular classification based on expression of mRNA [1]. The goal of this study is to provide a new method of classification of colon cancer based on transcriptome analysis. This improves disease stratification by allowing the use of common DNA markers and clinical variables not previously used. Genome-wide mRNA expression analysis was performed and analysed using a large number of multicentre and extensively characterised samples. The main aim is to normalise the data obtained to ensure that the differences in intensities are due to differential gene expression and not due to printing, hybridization, or scanning artifacts.

Methods

To convert the probe level data to expression values, the “affy” package (1.64.0) in R was used. It reads the array (using “ReadAffy()”) and normalises it (using “rma()”). This also includes background correction which is probe specific. The array has 1164 rows and columns which is converted to a ExpressionSet object prior to normalisation. Next “affyPLM” package (1.62.0) is called to compute Relative Log Expression (RLE) and Normalized Unscaled Standard Errors (NUSE) scores of the microarray samples. The function “fitPLM()” converts the AffyBatch into a PLMset and then normalises the data. RLE is a quality assessment tool which is used to compute the RLE values of each probeset by comparing each of the expression values with the median expression value for that probeset across all arrays. Once done, the distribution is examined by plotting a histogram. The output matrix has samples in the columns and the median in the first row. Similarly, NUSE is computed and a histogram is plotted.

Prior to computing the Principal Component Analysis, the Batch Effect Correction is performed using the sva package (3.34.0) in R. It identifies and removes artifacts. Functions like ComBat directly remove known batch effects using empirical Bayesian framework. Post this, the expression data is written to a CSV file, where probesets are rows and samples are columns.

PCA is a type of linear transformation on a given data set that has values for a certain number of variables (coordinates) for a certain amount of spaces. This linear transformation fits this dataset to a new coordinate system in such a way that the most significant variance is found on the first coordinate, and each subsequent coordinate is orthogonal to the last and has a lesser variance. In this way, one can transform a set of “a” correlated variables over “b” samples to a set of “p” uncorrelated principal components over the same samples.

Besides the above mentioned packages AnnotationDbi (1.48.0) and hgu133plus2.db (3.2.3) is used. AnnotationDbi provides a user-friendly interface for querying SQLite-based annotation data while Hgu133plus2.db is Affymetrix Human Genome U133 Plus 2.0 Array annotation data.

Result

RLE and NUSE are key steps for quality assessment to identify aberrant samples, and the results are visualized in the plots below.

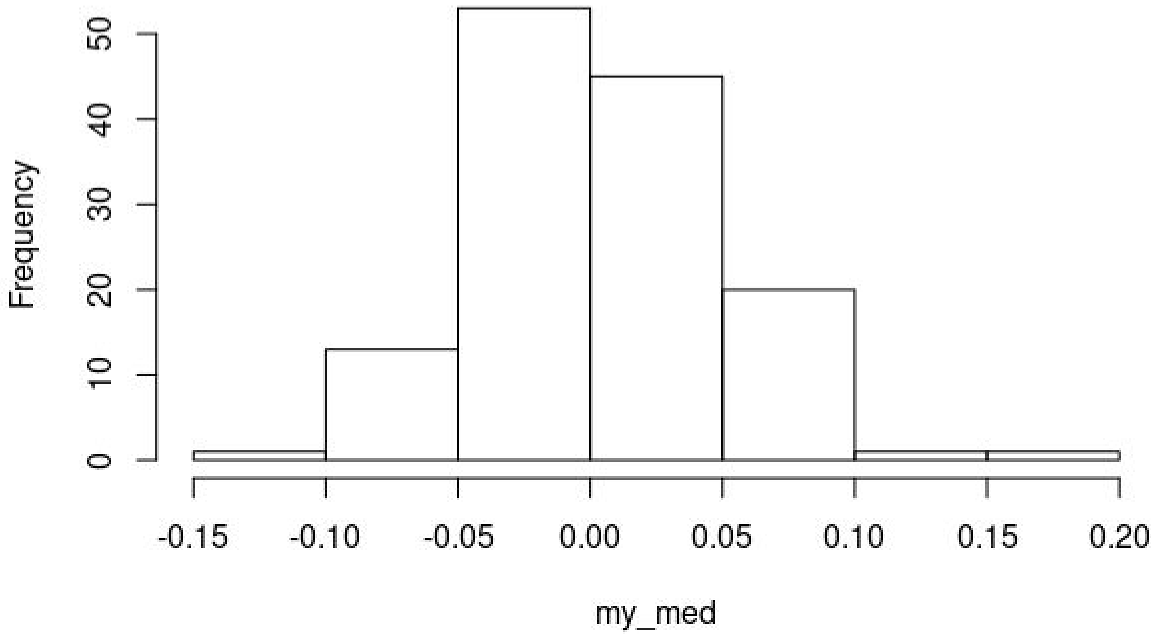

Fig 1: Histogram of RLE medians

RLE medians of about 90 samples are concentrated between -0.05 to 0.05. Assuming that most genes are not changing in expression across arrays, RLE values will be near 0, so this signifies good quality. But many samples remain outside this range with lesser quality. In the case of NUSE, the standard error estimates obtained for each gene on each array from fitPLM are taken and standardized across arrays so that the median standard error for the gene is 1 across all arrays. This process accounts for differences in variability between genes, such that medians close to 1 indicate high quality. In Fig 2, some samples are not very close to 1, for example, the long tail after 1.04. But, the majority of the data is within acceptable limits.

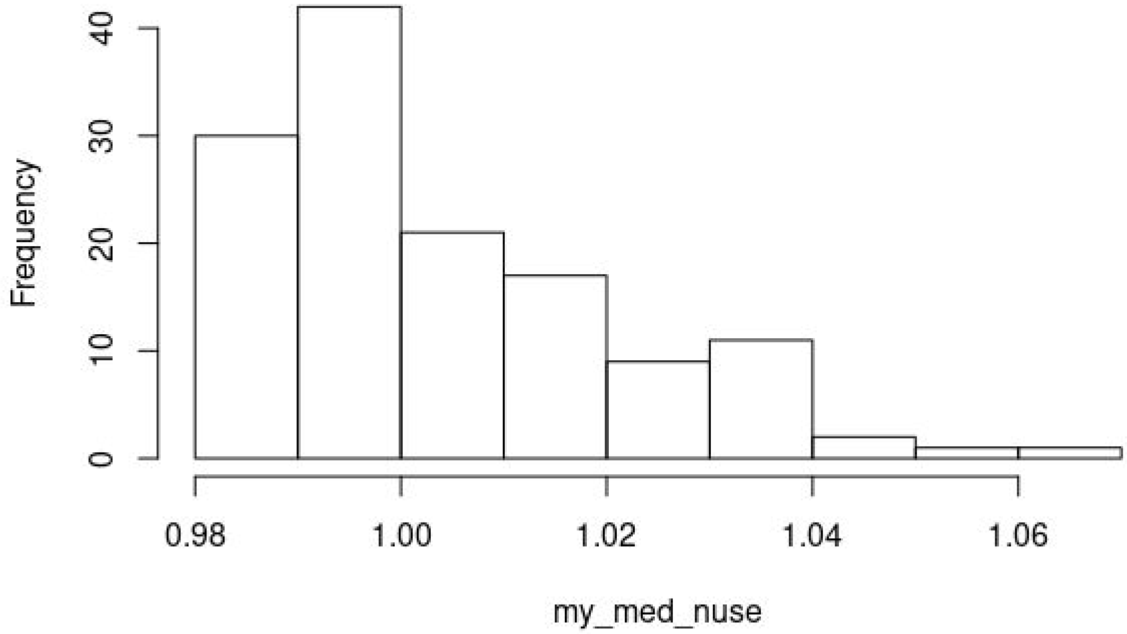

Fig 2: Histogram of NUSE medians

The gene expression data was then transformed with PCA to visualize the variance and find possible outliers. However, the sample distribution in Fig3 is not concentrated enough to form clusters. So, other parameters are tried in PCA plots, namely PC2 VS. PC3 (Fig 4), and PC1 VS. PC3 (Fig 5)

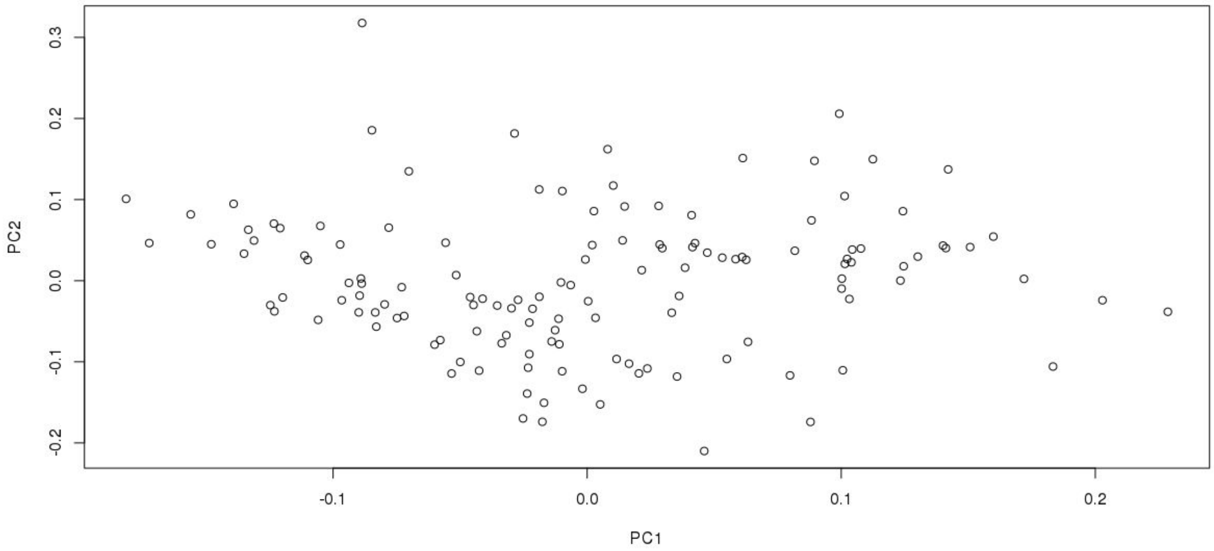

Fig 3: PC1 VS. PC2

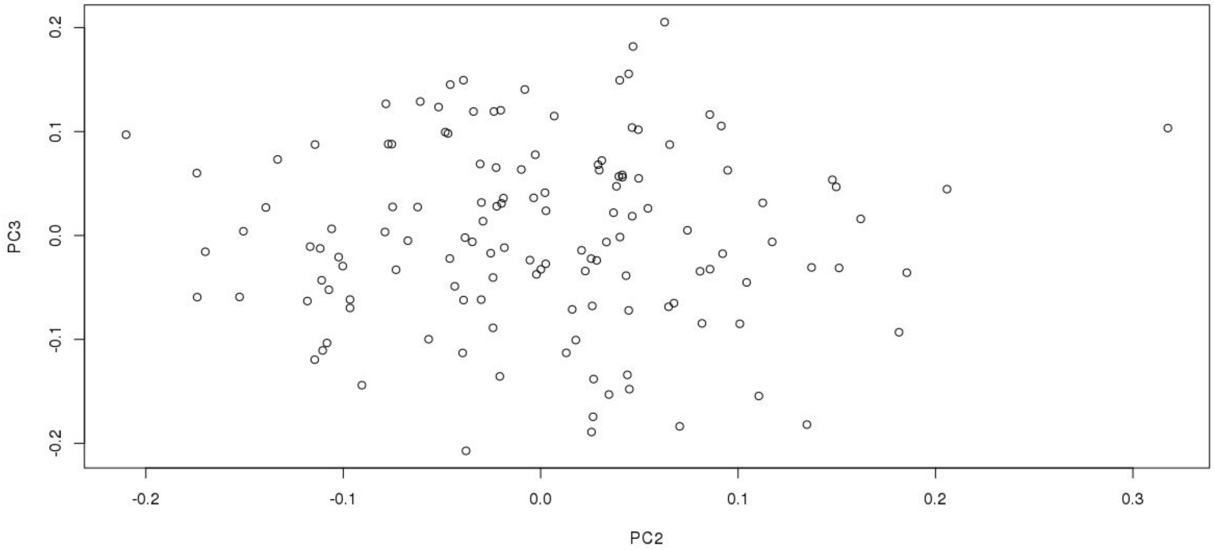

Fig 4: PC2 VS. PC3

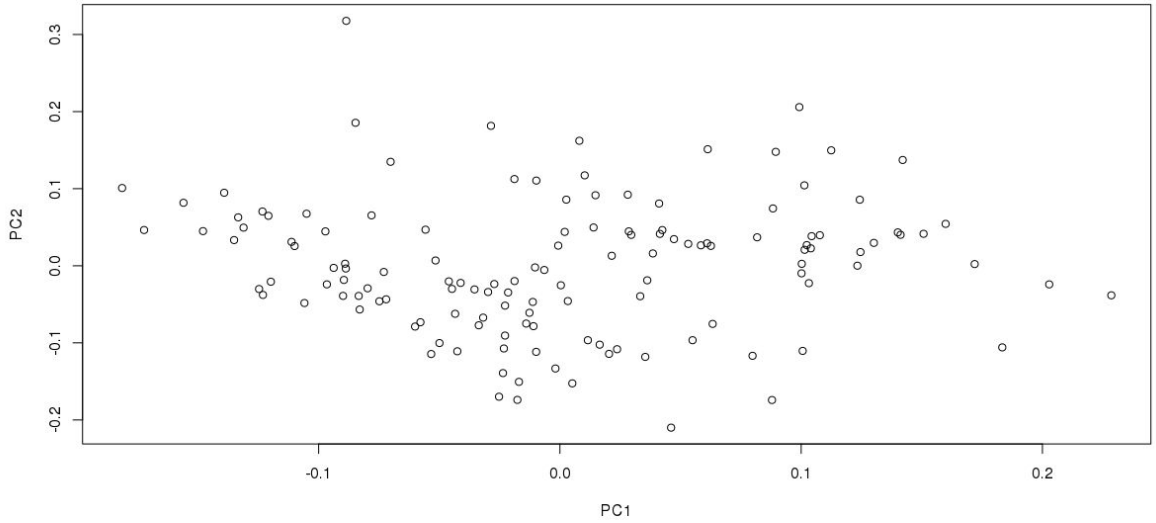

Fig 5: PC1 VS. PC3

However, the results obtained are not sufficient to describe the dataset so those rows which do not have values less than 2/1 or 2 are found to return a larger set of data. 'prcomp()' is again used with the same parameters as before and PC1 VS. PC2 is plotted as seen in the figures below.

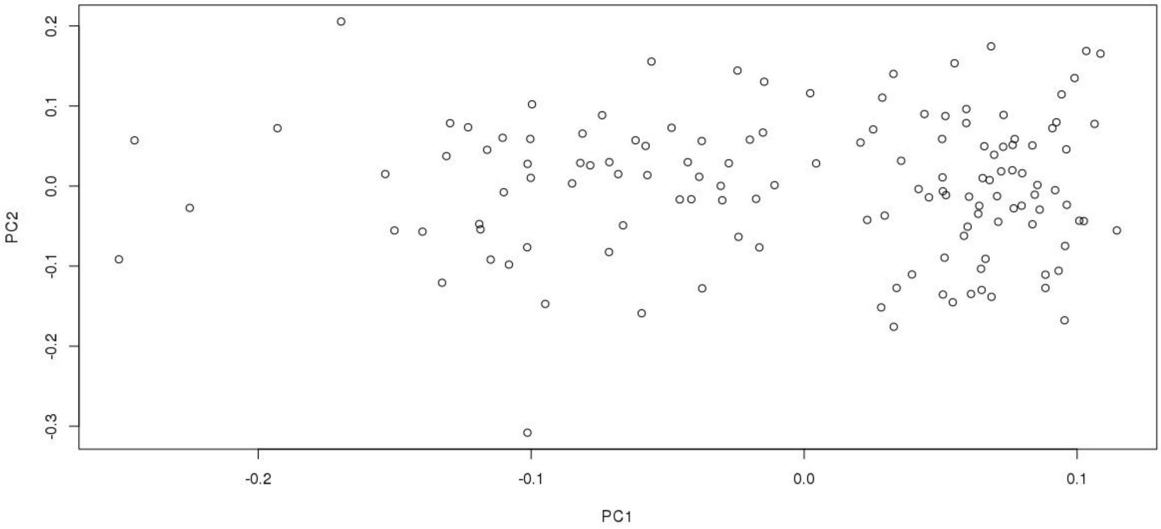

Fig 6: PC1 vs PC2

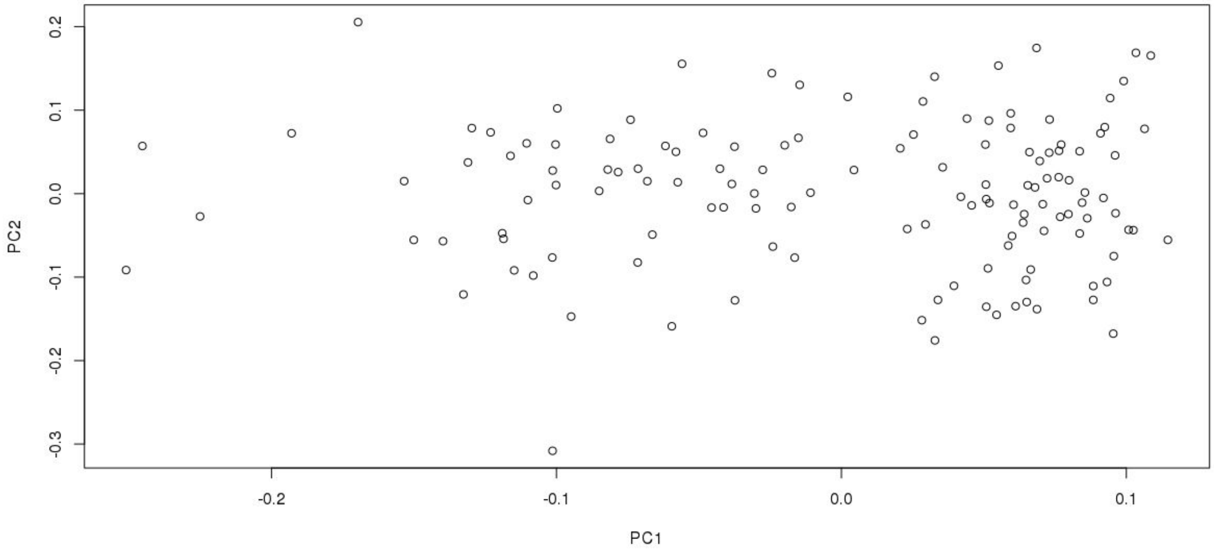

Fig 7: PC1 vs PC3

In conclusion, PC1 vs. PC2 contains more readable information as it is more clearly clustered.

Discussion:

Data obtained is normalised following which quality assessment is performed by computing RLE and NUSE medians. This is done to normalize data using quantile normalization, and correct background with RMA background correction. Prior to principal component analysis batch effect correction is performed to remove artifacts. If we perform PCA on unfiltered data, the result is not readable enough, in other words, no clear clusters can be found in the plot. If we set threshold or filter low value data, we are not certain whether these processes could influence the result. Following the analysis of the plots obtained, it was seen that PC1 vs PC2 is more clearly clustered.

(ANALYST)

Introduction

From the Genome-wide mRNA expression analysis the associations between the molecular subtypes and its resultant pathology, DNA alterations and the prognosis was studied. The obtained RMA normalized and Combat adjusted matrix was further filtered for noise since a low signal to noise ratio results in poor multivariate analysis. Hierarchical Clustering is a powerful analytical unsupervised method for grouping sets of similar objects based on some criterion, usually a series of features whose similarity is defined by some distance function. In the original paper Consensus Clustering is used for the analysis but since it is computationally intensive hierarchical clustering is performed here.

Methods:

The matrix obtained after normalisation is filtered to remove noise. For this purpose the following factors are taken into consideration:

1. 20% of the samples for each gene have an expression value greater than log2(15)
2. Variance for each gene is significantly different from the median variance ($p < 0.01$)
3. Coefficient of variance for the gene is > 0.186

Filter 1 tracks how many samples are above log2(15). It returns the count divided by the total number of samples to indicate genes where 20% of samples pass the threshold. Filter 2 begins by calculating the standard deviation, and returning the median. The chi-square test is then employed to find genes with significantly different variance from this median. This involves calculating the degrees of freedom, the test statistic, and the upper critical value given an alpha of 0.01. It returns whether the test statistic is greater than the critical value. Finally, filter 3 calculates the coefficient of variation (the ratio of standard deviation to mean) and checks if it is greater than 0.186.

Hierarchical Clustering is performed on the filtered matrix with the primary objective of grouping the samples based on their gene expressions. First, dist() function is used to generate a dissimilarity matrix. The transverse of the matrix must be used to ensure that the samples are being clustered. The value returned is passed through hclust() function to generate a dendogram or tree which represents the clustering. Finally, cutree() takes the output of hclust() and a number of groups, in this case k=2, and returns a vector assigning each of the samples to one of the groups. This is used to create two separate matrices for each cluster.

Next, gplots package (3.0.3) is employed for heatmap construction. The subtype is determined for each sample and a vector of color values corresponding to them is constructed, aiding the interpretation. Welsch t test is performed using t.test() function which is useful in comparing the gene's mean expression and determining if the differences are significant. The t-statistics and p-values are selected from the results and aggregated into a data table, which includes a column for adjusted p-values. This is found using the function p.adjust(), which takes the unadjusted p values and a specified method, in this case "fdr". The purpose of adjusting them is to reduce the false discovery rate.

Finally, to find the genes of greatest significance, the data table of t-test results is filtered in a similar manner to before. The genes with an adjusted p-value below 0.05 are kept and are organized in ascending order of adjusted p-value.

Results

Post filtering 54,675 probesets are reduced to 1531 using which the hierarchical clustering identifies similarities between 134 samples and divides them into two clusters of 57 and 77. Welch's t-test is subsequently used to find differential gene expression for all 1531 genes between clusters 1 and 2. The number of genes that pass the adjusted p-value threshold of 0.05 is 1236.

In the original paper subtypes were defined using the top five upregulated and downregulated genes. Thus, the genes were sorted by their significance (lowest p value) which corresponds to the comparisons with a positive t-statistic, indicating gene expression was higher in cluster 1. Alternatively, those with the lowest adjusted p-values but negative t-statistics should have higher gene expression in cluster 2.

Table 1: Table of most differentially expressed genes for each cluster, in order of significance.

Cluster 1		Cluster 2	
Probeset ID	Adjusted p-value	Probeset ID	Adjusted p-value
204457_s	6.78e-46	203240	3.99e-28
209868_s	2.94e-44	220622	8.99e-25
223122_s	5.44e-44	210107	1.70e-24

225242_s	1.18e-43	238750	6.11e-24
202291_s	2.78e-43	204673	9.79e-24

Next, heatmap was constructed with red representing the C3 subtypes and blue as the C4 subtype. Each row represents a different gene or probeset, and each axis also has an associated dendrogram.

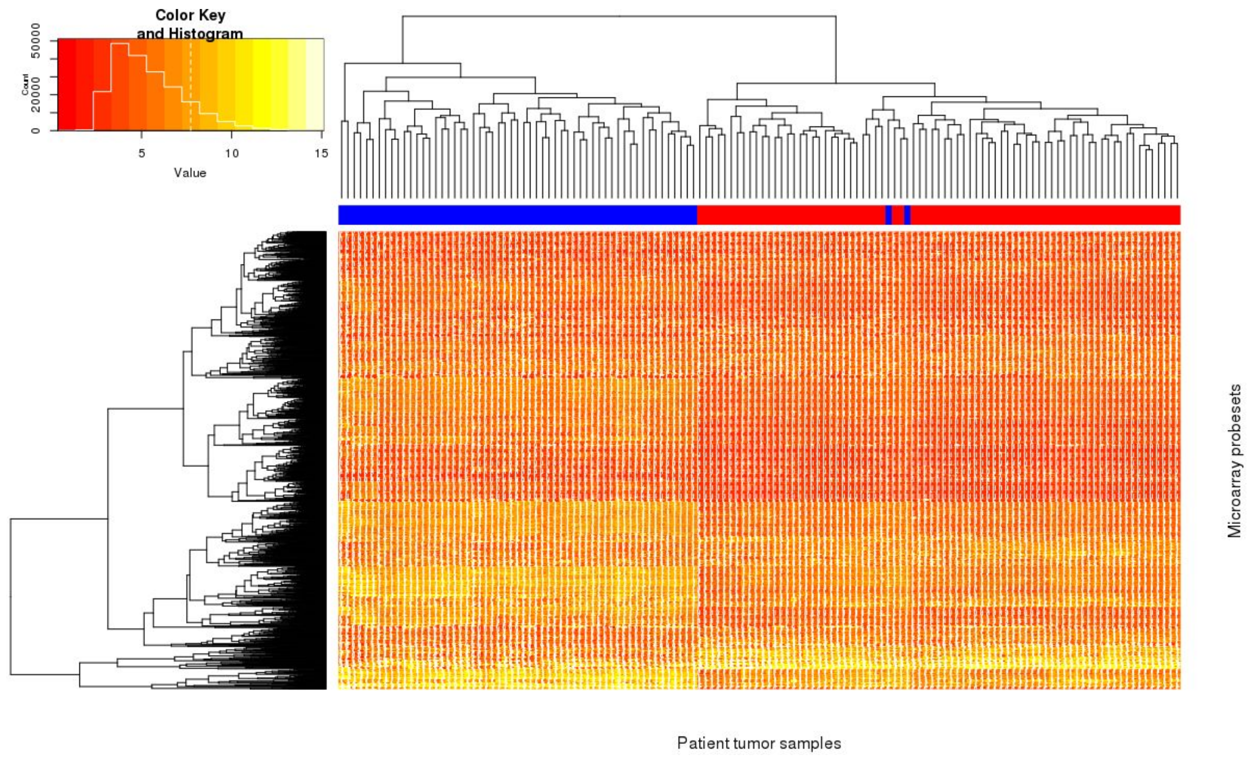

Fig 1: Heatmap of gene expression across C3 (red) and C4 (blue) subtypes for colon cancer samples

Discussion:

Genes filtered on the basis of high variance compared to all other genes might be indicative of genes related to the cancer phenotype of interest rather than housekeeping genes for example. The coefficient of variation filter on the other hand reflects the variation across samples for a single gene. Greater variation may suggest a gene that would be relevant when finding differences between samples during hierarchical clustering. One observation from the table of probe sets in each cluster is that adjusted p-values were much lower for cluster 1 than cluster 2. This suggests that the most highly differentially expressed genes occurred in this cluster, but this would be subject to change with the addition of more clusters or subtypes.

As for the heatmap seen in Fig 1, the density of data means at best, general patterns in gene expression between the two clusters can be commented on. It appears that for a majority of the genes assessed, the red C3 samples had higher expression levels than their blue C4 counterparts. Because this heatmap was constructed separately from the clustering and t-tests, we cannot determine if the red and blue subtypes correspond to the two clusters, although a relationship is likely.

References:

1. Marisa et al. Gene Expression Classification of Colon Cancer into Molecular Subtypes: Characterization, Validation, and Prognostic Value. PLoS Medicine, May 2013.