# Recapitulation of Leukemia Cell Gene Clusters using Transcription Factor Binding Sites as Indicators of Gene Expression

Submitted in partial fulfillment of the Senior Seminar Requirements for the Computer Science Major at Dickinson College.

By
Philip Hubert
hubertp@dickinson.edu

Advised by
John MacCormick, Jeffrey Forrester, Michael Roberts
jmac@dickinson.edu forresje@dickinson.edu robertsm@dickinson.edu

5/7/2012

## Abstract

# Recapitulation of Leukemia Cell Gene Clusters using Transcription Factor Binding Sites as Indicators of Gene Expression

by
Philip Hubert

This study aimed to recapitulate gene clusters derived from real gene expression data from a previously conducted microarray analysis at Dickinson College. This was done by finding matches for specific transcription factors in the DNA sequences of 748 genes' promoter region and clustering the genes based upon the number of matches using multiple clustering algorithms in the Cluster 3.0 software as well as the GenePattern clustering modules. First, the number of occurrences of each of the transcription factors in the promoter region was used as a predictor of the gene’s expression level. Then multiple sets of transcription factor matches were used to generate new clusters using predominately k means, Self-Organized Maps (SOMs), and some hierarchical clustering. For each set of gene clusters that was produced, a pairwise comparison of all of the genes was done to determine if they were in the same cluster for both the original expression data cluster set and the new cluster set. Using this analysis, a percentage of maintained cluster relationships was then calculated as a measure of the clustering model's success.

# Chapter 1

# INTRODUCTION AND BACKGROUND

## 1.1. Project Explanation

The task of curing cancer has proven itself to be one that requires a wide array of approaches as well as a large amount of new research to better understand the genetic mechanisms that cause cancer. Student researchers, including Natalie Stanley and Phoebe Oldach, working under Professors Jeffrey Forrester and Michael Roberts have previously generated data showing the amount of various proteins produced from the genes that encode those proteins for a particular type of leukemia cell. These leukemia genes were then organized into groups called gene clusters based upon the amount of protein that each gene produced over a period of 24 hours. The amount of protein produced from a gene is known as the gene expression level.

It is known that other environmental factors play into a gene's expression level but in my research I worked to create an accurate method for predicting leukemia gene expression levels using only their relevant DNA sequences, specifically the strings of molecules that contain code for proteins (see Background for DNA information). This goal would be achieved by generating groups/clusters of genes based upon a variety of scoring algorithms and comparing them with the clusters derived from the original gene expression data that is trusted to reflect how these genes actually operate.

When this research began it was thought to be novel, but recent investigation has shown that research done by Beer et al. and Elemento et al. (2004, 2007) has already explored this topic and determined that the methods being analyzed here would not produce suitable results. Nonetheless, this research should be presented in order to support their statements that the methods explored are not able to handle the genetic complexities inherent in predicting gene expression from genetic sequences. Additionally, there is still a limited amount of work done on mammal genes as they have proved even harder to predict than original tests done on the genes of *S. cerevisiae* and *C. elegans* (Beer & Tavazoie, 2004; Hill, Hunter, Tsung, Tucker-Kellog, et al., 2000). Clearly the research has wide implications but it will appeal particularly to experts in bioinformatics, biology, and possibly mathematics as it requires a particular level of genetic and bioinformatics knowledge to be fully understood.

## 1.2. Background

### *1.2.1. DNA*

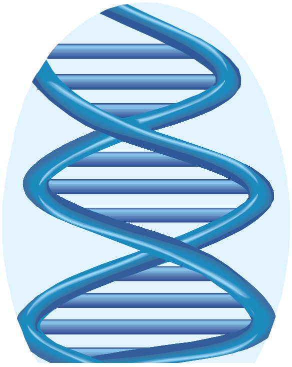


Figure 1.1: DNA is composed of two intertwining strands in a double helix formation.

Deoxyribonucleic acid or DNA as it is more commonly known is a double helix strand that contains all the genetic information necessary to create and regulate organic life. The term "double helix" refers to the structure of DNA and as can be seen in Figure 1.1, means it is made up of two long strands that spiral around each other, never touching but connected by chemical bonds in between them. These chemical bonds exist between individual nucleotides, which are the building blocks of DNA and RNA (ribonucleic acid). The nucleotides for DNA are Adenine, Guanine, Cytosine, and Thymine. Adenine and Thymine are complementary pairs that join together, as are Cytosine and Guanine. These nucleotides join together in a specific sequence (not just the complementary pairs) to form a strand of DNA and this strand will join with another strand of DNA that is its complement (e.g. Cytosine on the first strand will join with Guanine on the second) in order to create a full double helix DNA.

### *1.2.2. Transcription and Promoter Regions*

Within these DNA sequences there are regions that contain the code for the creation of proteins. The sequence of nucleotides (also known as base pairs) in a protein coding region are "read" by a structure called RNA polymerase which creates a complementary strand of nucleotides known as mRNA (messenger RNA) that will be used later to create the actual proteins. The RNA polymerase binds to a particular site on the DNA strand known as the Transcription Start Site (TSS) and moves downstream from the TSS making mRNA. This process is known as transcription and in order for the RNA polymerase to be able to engage in this process it must have assistance finding the TSS as well as the binding to the TSS.

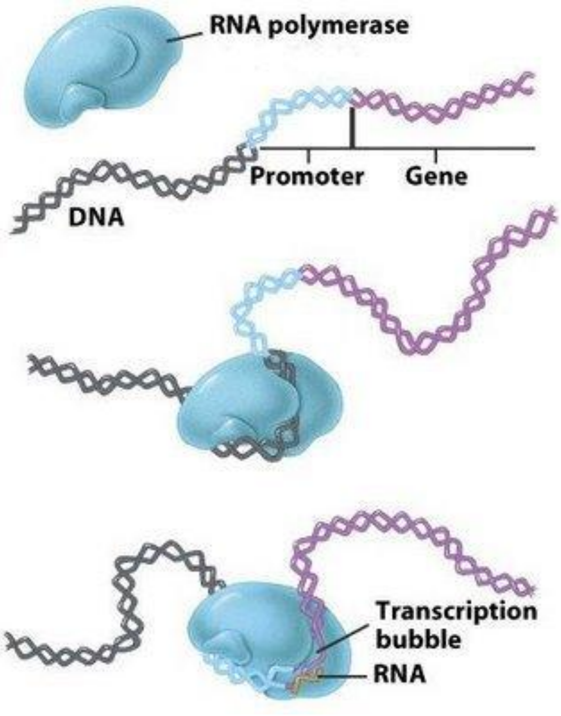


Figure 1.2: RNA Polymerase binding to a promoter region and engaging in protein transcription.

The promoter region is the area lying upstream of the TSS (upstream is opposite the direction of transcription, downstream and upstream are relative terms) and plays a key role in the process of attracting the RNA polymerase and binding it to the TSS. The promoter region contains sequences of nucleotides known as binding sites that provide locations for other molecules known as transcription factors to bind to. These transcription factors are what help the RNA polymerase bind and remain stably bound to the strand as it is transcribing the DNA. Without the transcription factors, the RNA polymerase can easily fall off because it is only loosely bound to the DNA or it may not be able to bind at all.

In order for proteins to be created, the RNA polymerase must be able to perform transcription and create mRNA that will be later read to create proteins through other processes. Thus the transcription factors and the promoter regions they bind to are known to play a large role in the levels of gene expression because the more mRNA produced in transcription, the more proteins that are likely to be produced. Gene expression can then be seen as the quantity of a particular protein structure that is created meaning more proteins created means a higher level of gene expression.

While the complex interactions that occur in promoter regions during transcription are not fully understood it is known that they have a large influence on gene expression levels (Eisen, Spellman, Brown, & Botstein, 1998; Bussemaker, Li, & Siggia, 2001). Recent research has shown that the orientation of the transcription factor (upstream or downstream), certain patterns of transcription factors, as well as the distance of a transcription factor from the TSS are all useful in predicting the level of gene expression. (Elemento, Slonim, & Takazoie, 2007)

### *1.2.3. Gene Sequence Acquisition and Storage*

The University of California in Santa Cruz (UCSC) maintains an online genome database called the Genome Browser (Fujita, Rhead, Warrell, Zweig, Hinrichs, 2011), which contains the DNA sequences of the complete human genome as well as other species and other genetic related data. A file format known as fasta (.fasta) has been previously created for the storage of both nucleotide (DNA) and amino acid (protein) sequences. The sequences are placed in a basic text file that contains a header line notated by '<' or ';' at the beginning of the header and then a character representation of the sequence follows with a fixed number of sequence elements on each line, typically 50 or 80.  Using the fasta file format, the DNA sequences for 748 relevant genes (as determined by the previous gene expression work) were stored locally for use.  Only 2,000 nucleotides were stored for each gene, 1,500 upstream and 500 downstream as it is known that this region is responsible for the large majority of gene expression regulation.

### *1.2.4 Clustering Techniques*

There are a large number of common clustering techniques and an even larger number of possible scoring metrics that could be used in those clustering techniques.  The idea behind clustering is to make it easier to look at a few hundred or thousand genetic sequences and organize them into groups (clusters) based upon some sort of sequence similarity or other feature similarities.  Clustering techniques for sequence alignment vary based upon whether you are looking for a global alignment (the entire sequence) or a local alignment (based upon an input) and how they are scored, e.g. using a distance algorithm to compute the level of difference between two sequences or regions within sequences.

For example, if there is interest in occurrences of a particular transcription factor in various gene sequences, then a local alignment would be done to line those sequences up based on the known sequence of that transcription factor.  Alternatively, if the interest lies in the relationships and history between different genes and their sequences, a global alignment may be done to make the entirety of each sequence match up with the rest as best as possible by shifting them up or down relative to each other. Clustering can also be done using a set of gene features, the values for those features in each gene, and then clustering using those values.

Given the same input, different algorithms, such as hierarchical or BLAST (Johnson, Zaretskaya, Raytselis, Merezhuk, 2008), will typically yield different results so a variety of algorithms need to be tested in order to find the one that will best model the gene expression data that we know to be true. This research used k-means, Self-Organized Maps (SOMs) and a few clusters were generated using a hierarchical clustering algorithm.

The k-means algorithm is a well-known clustering algorithm because of its simplicity as well as its ability to be reasonably effective for most clustering problems (Hartigan, Wong, 1979). In general, the k-means is an iterative algorithm that operates in two steps. The algorithm takes as input a set of data entries E with a corresponding set of values V and a starting cluster assignment. It begins by calculating the mean values of all the entries in each cluster and assigning those values as the cluster center. Then it reassigns the data entries to the cluster whose center its values are closest to (See Figure 1.3). This algorithm continues until a set number of iterations have been performed or a particular finish parameter has been met such as meeting a particular sum of all entries distances from their cluster centers. In the Cluster 3.0 implementation it also takes as input the number of clusters to be generated, the number of iterations to be run, as well as an option to set the distance metric.

Self-Organizing Maps are a similar algorithm to k-means in that it works to minimize the distance between items in a single cluster and maximize the distance between clusters. What separates it from k-means is that it creates an artificial neural network that maps N features into a 2-dimensional map (See Figure 1.4) that has nodes placed to create edges that draw boundaries between cluster areas rather than defining a cluster center. This is a useful tactic in that it allows for an image to be drawn of the cluster space to be analyzed visually as well as also providing an efficient means for processing large amounts of outputs from the algorithm. The implementation used was from the GenePattern Server (Reich et al, 2006). Additionally, though this research has not explored it, a 2-step technique has been presented in which an SOM is generated and another clustering algorithm run with the SOM as input in order to achieve time-efficient and successful results (Vesanto, Alhoniemi, 2000).

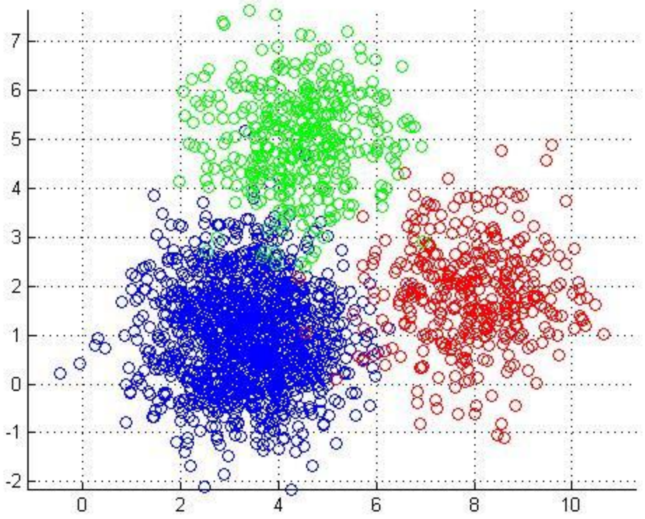


Figure 1.3: Visualization of K-means clusters (Mathworks, 2012).

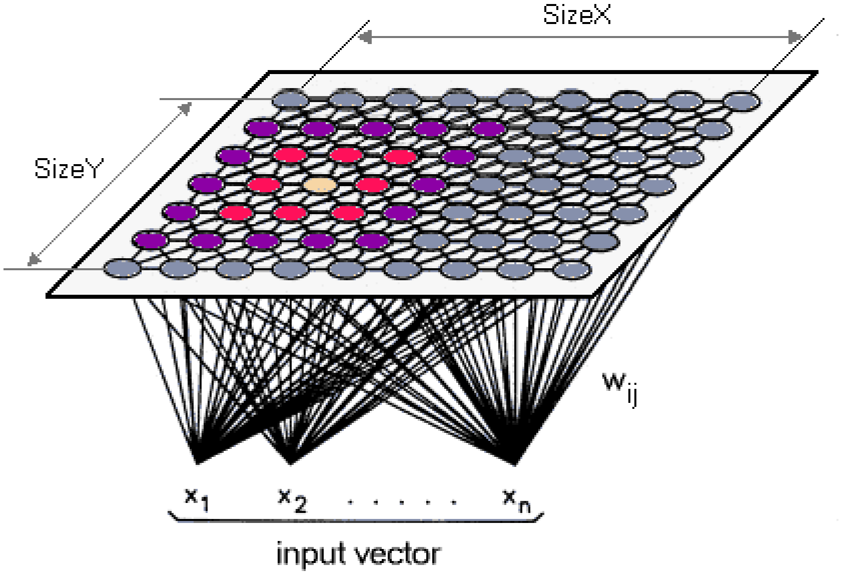


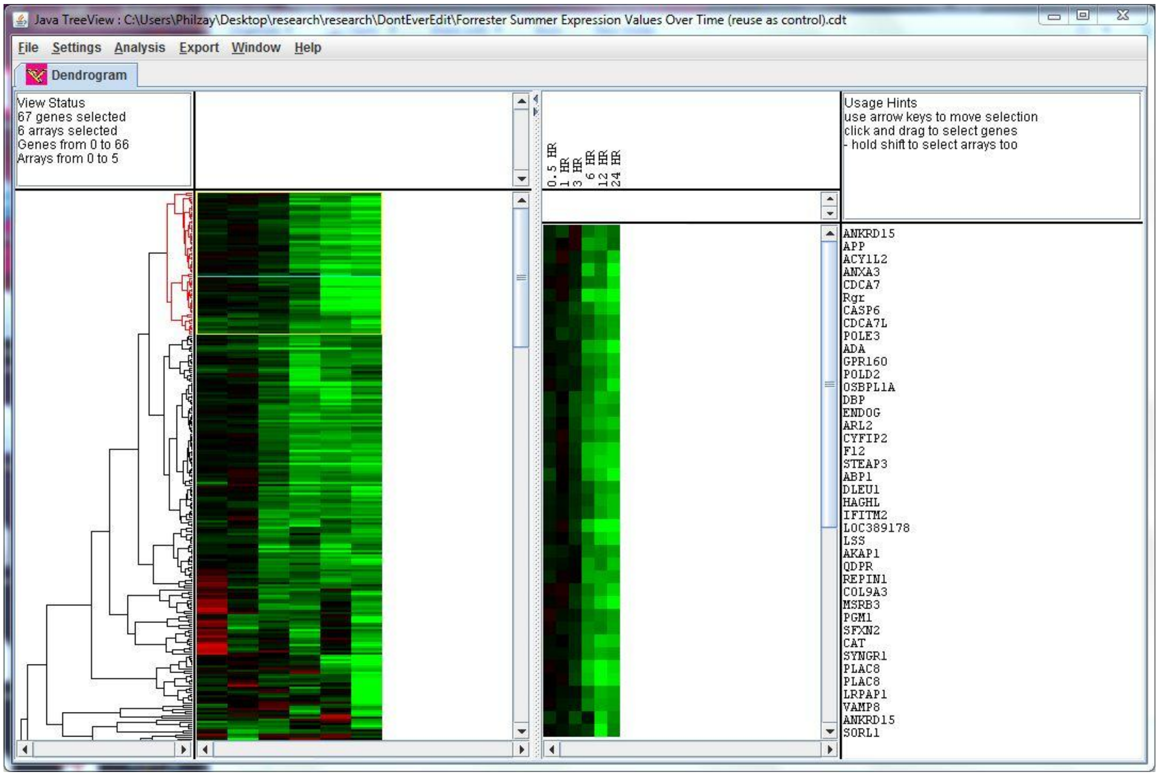
Figure 1.4: Visualization of a Self-Organizing Map mapping N features into a 2-dimensional mapping space (SDL Component Suite, 2008).

The hierarchical clustering implementation in Cluster 3.0 software was used (de Hoon, Imoto, Nolan, Miyano, 2004). Hierarchical clustering uses a distance algorithm, such as the Euclidean distance, on gene data to create a gene tree where the length of a branch is representative of the distance between genes. It can take either a top-down approach where all genes start in the same cluster and then are recursively divided up based on the distance function used. Alternatively, it can take a bottom-up approach where each gene starts out in its own cluster and pairs of clusters are then recursively merged together until there is one cluster at the top of the hierarchy.

This hierarchy can then be fed into a tree-viewing program such as Java TreeView (See Figure 1.4) that graphically displays the cluster hierarchy with a heat map of the feature values. It also identifies the clusters from the hierarchical tree by finding the hierarchical level with the lowest sum of distances between each of the gene's scores in each of the clusters (Saldanha, 2004). This comparison requires the two trees to both have the same set of features so it was not feasible to compare because the original data used expression levels over time and the models tested looked at transcription factor counts and location. This technique was only used to cluster the original gene expression data and a few tests using scoring techniques that were promising in other algorithms because it is more suitable for visual representation.

### 1.3. Relevant Research

The work done by Forrester, Roberts, and their student researchers this summer is the most closely related research as its results are being directly used in my research.  In their research, they conducted a microarray analysis of the gene’s expression levels over a period of 24 hours and clustered them using the hierarchical clustering method in Cluster 3.0. This produced a set of 23 clusters containing 4 to 159 genes with most clusters containing 15 to 20 genes.



Java TreeView : C:\Users\Philzay\Desktop\research\research\DontEverEdit\Forrester Summer Expression Values Over Time (reuse as control).cdt

View Status
67 genes selected
6 arrays selected
Genes from 0 to 66
Arrays from 0 to 5

Usage Hints
use arrow keys to move selection
click and drag to select genes
- hold shift to select arrays too

```
ANKRD15
APP
ACY1L2
ANXA3
CDCA7
Rgr
CASP6
CDCA7L
POLE3
ADA
GPR160
POLD2
OSBPL1A
DBP
ENDOG
ARL2
CYFIP2
F12
STEAP3
ABP1
DLEU1
HAGHL
IFITM2
LOC389178
LSS
AKAP1
QDPR
REPIN1
COL9A3
MSRB3
PGM1
SFXN2
CAT
SYNGR1
PLAC8
PLAC8
LRPAP1
VAMP8
ANKRD15
SORL1
```

Figure 1.4: Example of a hierarchical clustering being viewed in Java TreeView. It contains a dendrogram of the clustered genes with their associate feature values represented as a heat map. The red highlighted portion in the first window has its heat map magnified in the second window and a list of all the genes within it in the third window.

Research into the strength of DNA sequences as predictors of gene expression has only been investigated in the last 15 or so years. Initial work was done looking at gene expression levels and the presence of particular regulatory elements and later expanded to add conditions under which a particular TF motif would act as a regulatory element (and thus have an impact on the gene expression). Most experiments were done on simple organism such as *S. cerevisiae* and *C. elegans* (Beer & Tavazoie, 2004; Hill, Hunter, Tsung, Tucker-Kellog, et al., 2000) which have less complicated regulatory mechanisms than humans. This led to the development of a more general framework that focused solely on the gene sequences and their correlation to gene expression rather than using probability matrices for TF motifs in particular genes.

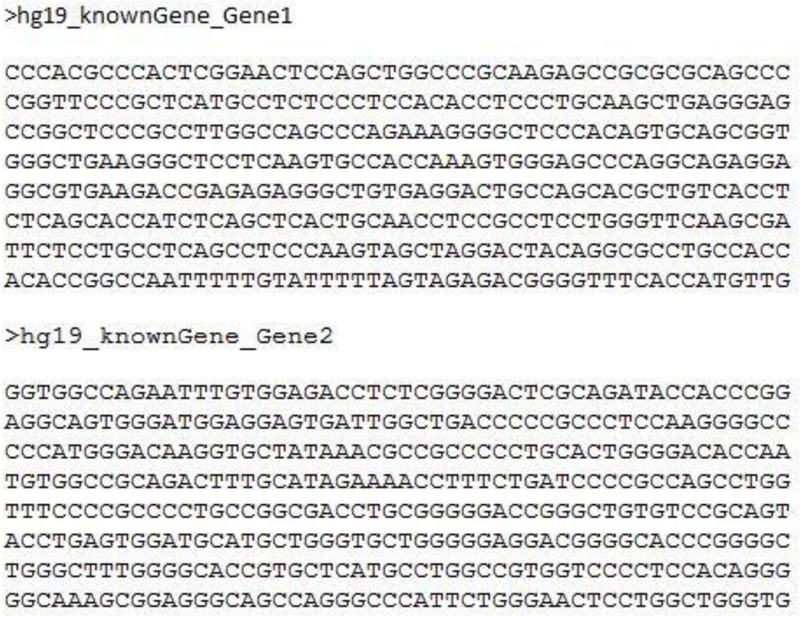
Others have compared promoter region sequences with different cells and gene expression data from experimentally generated data or using online gene expression databases such as Transfac (Wasserman, Sandelin, 2004; Wingender, Chen, Hehl, Karas, 2000). Some useful clustering techniques intended for the discovery of structures in a gene's promoter region have also been suggested using machine learning (Curk, Petrovic, Shaulsky, & Zupan, 2009; Fitzgerald, Shlyakhtenko, Mir, & Vinson, 2004). These articles also discussed the difficulties of determining the combinatorial effects of transcription factors and other promoter region regulatory structures.

# Chapter 2

# Project Implementation

## 2.1. Methods

Using the standard Java 5.0 libraries, DNA sequence input is received in a .fasta file, processed, and then stored in an object containing a gene name and the one or more string representations of that gene's promoter region sequence. An example fasta file can be seen in Figure 2.1.  This sequence information was downloaded from the UCSC Genome Browser with the promoter region defined as 2000 nucleotides, 1500 base pairs upstream from the TSS and 500 base pairs downstream.  The sequences were then searched for known TF binding site DNA sequences and passed to the Cluster 3.0 clustering software for either k-means or hierarchical clustering. Then the resulting cluster set was compared with the cluster resulting from running the original gene expression values through the same clustering algorithm with the same parameters in order to determine the model’s accuracy.  In all cases, the modeled gene clusters were compared with the clusters generated by passing the actual gene expression data through to determine the accuracy of the model.  Each model will seek to improve upon the previous model by using insight gained from analysis of the two sets of gene clusters produced by that previous model.



```
>hg19_knownGene_Gene1

CCCACGCCCACTCGGAACTCCAGCTGGCCCGCAAGAGCCGCGCGCAGCCC
CGGTTCCCGCTCATGCCTCTCCCTCCACACCTCCCTGCAAGCTGAGGGAG
CCGGCTCCCGCCTTGGCCAGCCCAGAAAGGGGCTCCCACAGTGCAGCGGT
GGGCTGAAGGGCTCCTCAAGTGCCACCAAAGTGGGAGCCCAGGCAGAGGA
GGCGTGAAGACCGAGAGAGGGCTGTGAGGACTGCCAGCACGCTGTCACCT
CTCAGCACCATCTCAGCTCACTGCAACCTCCGCCTCCTGGGTTCAAGCGA
TTCTCCTGCCTCAGCCTCCCAAGTAGCTAGGACTACAGGCGCCTGCCACC
ACACCGGCCAATTTTTGTATTTTTAGTAGAGACGGGGTTTCACCATGTTG

>hg19_knownGene_Gene2

GGTGGCCAGAATTTGTGGAGACCTCTCGGGGACTCGCAGATACCACCCGG
AGGCAGTGGGATGGAGGAGTGATTGGCTGACCCCCGCCCTCCAAGGGGCC
CCCATGGGACAAGGTGCTATAAACGCCGCCCCCTGCACTGGGGACACCAA
TGTGGCCGCAGACTTTGCATAGAAAACCTTTCTGATCCCCGCCAGCCTGG
TTTCCCCGCCCCTGCCGGCGACCTGCGGGGGACCGGGCTGTGTCCGCAGT
ACCTGAGTGGATGCATGCTGGGTGCTGGGGGAGGACGGGGCACCCGGGGC
TGGGCTTTGGGGCACCGTGCTCATGCCTGGCCGTGGTCCCCTCCACAGGG
GGCAAAGCGGAGGGCAGCCAGGGCCCATTCTGGGAACTCCTGGCTGGGTG
```

Fig 2.1: Example of a .fasta file containing DNA sequences for two genes' promoter regions.

### *2.1.1. Number of TF Matches as Scoring Method*

The first scoring procedure counted the number of occurrences for each of 54 different transcription factor binding sites. A matrix of nucleotide frequencies for the transcription factors was retrieved from the JASPAR database and a list of all the sequences used to create the transcription factors was created (Vlieghe, Sandelin, De Bleser, Vleminckx, et al., 2006; Abbreviations, 1970).  Out of a total of 65 possible transcription factors, 11 did not have any matches in any of the genes and were removed from the set.  For each gene, an initial range of 500 nucleotides upstream of the TSS was defined as the search region to find matches of the transcription factor and the number of instances of each transcription factor was stored.

A basic scoring function was created using the number of transcription factor occurrences as the only feature used to generate the gene clusters. The list of genes with the number of instances of each transcription factor was then given as input into the Cluster 3.0 k-means clustering algorithms. The set of gene clusters that was produced by the hierarchical clustering was then compared with the original gene clusters to determine the accuracy of the model.

The creation of an improved scoring function was attempted by integrating the transcription factor binding site match's location in the promoter region into the simple match count. This approach was never fully utilized however as it was difficult with such a large amount of data to determine the significance and effect of a motif's location and position on the gene expression/cluster.

### *2.1.2. Analyzing Model Success*

A variety of methods to determine the success level of each gene clustering were used. First, a confusion matrix was utilized with the rows representing the gene clusters generated from the model and the columns represented the original gene clusters. The confusion matrix was necessary and useful to determine which of the original clusters each of the modeled clusters are similar (ideally identical) to. The value at a particular intersection in the confusion matrix is simply the total number of genes that were in the union of the two gene clusters. If the two gene clusters are similar, then every row and column will have one large number and the rest will be relatively small or zero indicating that each cluster matches well with only one of the clusters from the other set. Said another way, there should be a one to one relationship between the pairs of similar gene clusters. This is necessary as unless the clusters can be found to be comparable there is not a useful way to further analyze them as they're clearly not accurate.

Next, percentage accuracy for the model was determined. Ideally, for each of the matched pairs of gene clusters, the percentage of the genes in the original clusters that are also found in the modeled clusters would be stored. As there is no implicit mapping between the two sets of clusters there is no way to determine whether a particular gene was placed in to the “correct” cluster.

Therefore, a different approach utilizing the basic relationship between genes in the cluster sets as either being in the same cluster in a set or being in different clusters was used. This pairwise gene comparison scoring algorithm iterated through all possible gene pairs, determined if the two genes were in the same cluster in both the original cluster set and modeled cluster set, and then computed the overall score. This score was equal to S/D where S is the total number of pairs for which the pair of genes was in the same cluster in both sets and D is the total number of pairs for which the pair of genes were in the same cluster in one set and different in another. This score essentially gives a ratio for the number of gene cluster relationships that are maintained between the two cluster sets. The total number of gene pairs that were found to be in different clusters in both cluster sets were not included because we are interested in genes that are meant to be clustered together and determining the number of genes that aren’t supposed to be clustered together is not a useful measure.

In order to determine the significance of the resulting pairwise comparison, 100 different cluster sets where all 748 genes were randomly assigned to a cluster and then compared with the original summer results. The average performance of these random cluster sets was used as the baseline value for comparison.

# Chapter 3

# Project Results and Future Work

## 3.1. Results

The confusion matrix method did not show strong cluster correlations between the two cluster sets so the pairwise comparison was the only method used for comparison. Interestingly, the results showed that for particular TFs, even searching for just one TF gave an increase over the baseline values of 2.7% for k-means and 2.6% for SOMs. This performance boost did not increase significantly when additional TF matches were searched for and actually drops off after 10 TFs. When all 54 TF motifs were searched for, the accuracy was 3.17%. The results for all possible subsets of four TFs achieved only a maximum of 7.4% retained gene relations using the k-means clustering algorithm. When compared to the results from the SOM testing given the same input, the SOM produced clusters that had an average 2% less retained gene relations with the highest only reaching 6.8%.

These results are not high but the metric's scale itself is not linear so these numbers need to be interpreted with that understanding. For each additional gene that is placed into the "correct" cluster, there is also then a decrease in the number of genes that it is not in the "correct" cluster with. This ratio of positive to negative effects on the overall accuracy score starts out low when the modeled clusters show low correlation to the original clusters and increases quickly as the number of genes "correctly" assigned increases. Since it is also known that k-means and other common agglomerative clustering algorithms aren't well suited to solve this problem two things can be concluded from these results. First, these results support the previous statement and second, as SOMs have been said to be best suited for pre-clustering, it is not surprising that k-means consistently produced better cluster sets than the SOMs.

The ten most prevalent transcription factors and their combinations with two other transcription factors did not achieve high results either, reaching only 7.8%. Prevalent transcription factors included: NKX3-1, RELA, NF-kappaB, SRF, EGR1, MIZF, HLF, NR2F1, NFIL3, TAL1-TCF3, RORA_1, INSM1, NHLH1, ELK4, MEF2A, NFE2L2. The best pairwise comparison results of the k-means clustering algorithms using different numbers of transcription factors can be seen in Figure 3.1.

Lastly, it should be noted that the hierarchical clustering method proved difficult to analyze as the file output from Cluster 3.0 was designed to lend itself to efficiently produce an image for a graphical interface. In the case of the k-means and SOM clustering file outputs, the files made it clear which cluster each gene was assigned to whereas the hierarchical clustering file output would have required a manual analysis.

### 3.2. Future Work

As was noted before, the research that has been presented here has already been found to be an ill-suited approach to solving this problem as some *a priori* information is necessary for accurate gene expression level prediction, including the distribution probability of the transcription factors in each original gene cluster. Attempting to determine the relationships between individual transcription factors and that relationship's resulting effect on a gene's expression level is difficult due to the large amount of data and the sheer complexity of the features involved. Thus more work could be done to create a Bayesian network including this and other transcription factor motif match information and following the procedures laid out by Elemento et al (2007). Their research suggests that their approach could be useful on this data set and new parameters can be researched and integrated into their model to further improve its accuracy. This could include more environmental factors, larger sequence search areas and motif match scoring.

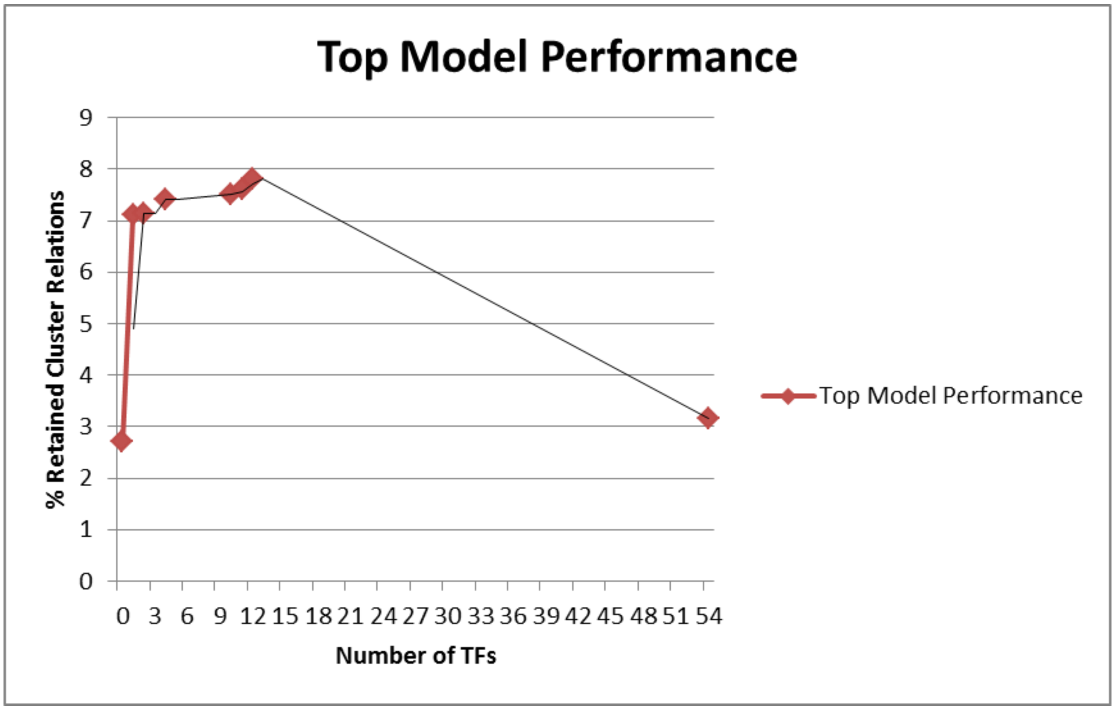


Figure 3.1: Top model performance based upon the number of transcription factors searched for using the k-means algorithm and the pairwise comparison analysis.

Additionally, because this research worked with subsets of variable size from the total set of 54 relevant different transcription factors, there were many more possible transcription factor subsets than could be tested within this study's timeline. Research into the relationships between the significant transcription factors found will also prove useful in determining the common functionalities of the genes in particular clusters. It is known that NF-kappaB is composed of RELA and NFKB1, yet NFKB1 did not appear in the set of significant transcription factors thus further research into this is necessary to elucidate any possible significance of this information. It may be that a distance or orientation metric is necessary to determine how NFKB1 binding sites have an effect on gene expression level, if it does at all.

Some environmental factors that will prove particularly difficult to model include the ability of some transcription factors to bind to other transcription factors rather than the gene DNA sequence. Though it is useful to visualize DNA sequences as linear, in reality they are known to loop and coil depending upon both the DNA sequence itself and external factors. This leads to the possibility that a binding site thousands of base pairs away from the TSS can have a significant impact on the gene expression level, further increasing the difficulties of creating an accurate model.

One option that may serve to both validate the existence of transcription factor binding sites discovered in this research as well as to discover new binding sites would be to perform ChIP-seq experiments (Liu, Potts, Huss, 2010). Simply put, ChIP-seq is a test that is able to experimentally determine the specific binding sites by testing for particular changes in the DNA that occur when a transcription factor binds to a gene's DNA. The resulting data could then be used as input into the Bayesian network described by Elemento et al. (2007).

This future work only includes a few possible directions for this research to go in but there are many more options available for improving the efficacy of this gene clustering model. Additional knowledge of the genetic mechanisms of transcription as well as experimentally determined binding sights from ChIP-seq experiments will give insight as to what extensions or modifications to the existing approaches will prove most useful.

<u>**Chapter 4**</u>

## REFERENCES

Abbreviations and symbols for nucleic acids, polynucleotides, and their constituents. (1970). *Biochemistry*, *9*(20), 4022-4027.

Beer, M., Tavazoie, S. (2004). Predicting Gene Expression From Sequence. *Cell*, *117*(2), 185-198.

Bussemaker, H., Li, H., & Siggia, E. (2001). Regulatory element detection using correlation with expression. *Nature Genetics*, *27,* 167-174.

Curk, T., Petrovic, U., Shaulsky, G., & B. Zupan. (2009). B. Rule-based clustering for gene promoter structure discovery. *Methods Inf Med*, *48*(3), 229-235.

de Hoon, M.J., Imoto S., Nolan J., Miyano S. (2004). Open source clustering software. *Bioinformatics*, *20*(9), 1453-4.

Eisen, M.B., Spellman, P.T., Brown, P.O., & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences, 95*(25)*,* 14863-16868.

Elemento, O., Slonim, N., & Tavazoie, S. (2007). A Universal Framework for Regulatory Element Discovery across All Genomes and Data Types. *Molecular Cell*, *28*(2), 337-350.

Fitzgerald, P.C., Shlyakhtenko, A., Mir, A.A., & Vinson, C. (2004). Clustering of DNA sequences in human promoters. *Genome Res*., *14*(8), 1562-1574.

Fujita, P., Rhead, B., Warrell, B. J, Zweig, A., Hinrichs, A., et al. (2011). The UCSC Genome Browser database: update 2011. *Nucleic Acids Research*, 1-7.

Hartigan, J.A., Wong, M.A. (1979). *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *28*(1), 100-108.

Hill, A.A., Hunter, C.P., Tsung, B. T., Tucker-Kellog, G., et al. (2000). Genomic Analysis of Gene Expression in *C. elegans*. *Science*, *290*(5492), 809-812.

Johnson, M., Zaretskaya, I., Raytselis, Y., Merezhuk, Y., et al. (2008). NCBI BLAST: a better web interface. *Nucleic Acids Research*, *36*(2), W5-W9.

Liu, Edison T., Pott, Sebastian, Huss, Mikael. (2010). Q&A: ChIP-seq technologies and the study of gene regulation. *BMC Biology*, *8*(56).

Mathworks. (2012). kmeans clustering – File Exchange – MatLab Central. Retrieved May 3, 2012, http://www.mathworks.com/matlabcentral/fileexchange/24616.

Reich, M., Liefeld, T., Gould. J., Lerner, J., et al. (2006). GenePattern 2.0. *Nature Genetics*, *38*(5), 500-501.

Saldanha, A. (2004). JavaTreeview – extensible visualization of microarray data. *Bioinformatics*, *20*(17), 3246-3248.

SDL Component Suite. (2008). Kohonen Network - Background Information. Retrieved May 3, 2012, http://www.lohninger.com/helpcsuite/kohonen_network_-_background_information.htm.

Vesanto, J., Alhoniemi, E. (2000). Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, *11*(3) 586-600

Vlieghe, D., Sandelin, A., De Bleser, P.J., Vleminckx, K., et al. (2006). A new generation of JASPAR, the open-access repository for transcription factor binding site profiles. *Nucleic Acids Res.*, *34*, D95-7.

Wasserman, W., Sandelin, A., (2004).Applied bioinformatics for the identification of regulatory elements. *Nature Reviews Genetics*, *5*, 276-287.

Wingender, E., Chen, X., Hehl, R., Karas, H., et al. (2000). TRANSFAC: an integrated system for gene expression regulation. *Nucleic Acids Research*, *28*(1), 316-319.