CS229 Final Project: Multi-class motif discovery in keratinocyte differentiation

Daniel Kim

December 12, 2014

Introduction

Enhancer elements are short segments of regulatory DNA that play important roles in activating gene expression within dynamic biological processes like cell differentiation. These enhancer regions are often bound by activating proteins known as transcription factors (TFs), and the DNA sites bound by TFs often have a sequence pattern, known as a 'motif', that can be matched to individual TFs. With over 2 million putative enhancer regions in the human genome, understanding enhancer function is essential to understanding dynamic gene regulation. As such, discovering regulatory sequence motifs within enhancer regions is an area of continuing research. Previous methods have included the use of discovered motifs and looking for matches within enhancer regions, as well as machine learning techniques (logistic regression and SVMs) among others [2]. However, such methods have only compared positive examples to negative examples, in a binary-class format. Here, we extend machine learning techniques to discover sequence features in a multi- class format, which is useful in situations where enhancers have varying activity patterns and we are interested in the underlying sequence features that are leading to these multiple patterns. We use the skin differentiation process as our test case.

Skin differentiation, the transition from skin progenitor cell to differentiated keratinocyte cell, is a process disrupted in skin cancer (squamous cell carcinoma and basal cell carcinoma), psoriasis, chronic wounds, and nearly 100 inherited human skin disorders. Disruption of skin differentiation is most often caused by changes in gene regulation, as shown in studies on transcription factors and their roles in skin function and development. Additionally, accurate genetic models of primary keratinocytes (both *in vitro* and *in vivo*) have proven to be powerful tools in understanding regulation by transcription factors and non-coding RNAs. Furthermore, numerous genomic time series datasets have been generated with primary keratinocytes. The dynamic nature of epidermal homeostasis, biologically relevant model systems, and availability of genomic datasets makes keratinocyte differentiation an ideal process to understand enhancer elements and the underlying sequence differences.

Materials and Methods

Datasets We collected histone ChIP-seq (H3K4me1, H3K4me3, H3K27me3, H3K27ac, H3K9ac) and ATAC-seq for 3 time points in skin differentiation (days 0, 3, and 6). We pre-process the data so that we determine enhancers and their activity levels at each time point. Then, we convert the sequence in each enhancer region into counts of k-mers, which are short DNA sequences of length k (for k=2, sequences include AA, AC, AG, AT...TT). After pre-processing, the rows are examples of enhancers and the columns are the k-mer features (depending on k, there are 4^k features, since there are 4 bases). The examples are labeled with which enhancer activity pattern it belongs to.

Pre-processing: bifurcation analysis for labeling To get enhancer elements, we first utilize a hidden Markov Model to combine histone mark data and output chromatin 'states' across the genome (ChromHMM[1]). We also call peaks from ATAC-seq data, which marks regions of open chromatin. We then intersect the two set of regions obtained to get a consensus set of regions. We then utilize the ATAC-seq sequencing reads found in those regions and look for differential peaks utilizing DESeq. Thus, peaks from one time point to another can increase (1), decrease (-1), or stay the same (0). With three timepoints and two pairs of comparisons that can be made across time, this allows us to segment the peaks into 9 different peak patterns, which constitute our 9 classes for prediction. We then label each group in order (increase/increase is group 1, increase/same is group 2, increase/decrease is group 3, and so on, as shown in Table 1).

Table 1: Number of samples in each class

Group	Day 0 to Day 3	Day 3 to Day 6	Number of samples
1	Up	Up	704
2	Up	No change	5078
3	Up	Down	996
4	No change	Up	5592
5	No change	No change	41725
6	No change	Down	5265
7	Down	Up	1002
8	Down	No change	5158
9	Down	Down	738

Multinomial logistic regression We take the sequence data and convert into k-mers (lengths of 2-4) and track the counts of each kmer found in the sequence. For each length k, we have 4k features. We then run logistic regression with lasso penalty (R package: glmnet[3]), which runs multinomial logistic regression based on the equations below:

$$Pr(G = \ell|x) = \frac{e^{\beta_{0\ell}+x^T\beta_\ell}}{\sum_{k=1}^{K} e^{\beta_{0k}+x^T\beta_k}} \tag{1}$$

$$\max_{\{\beta_{0\ell},\beta_\ell\}_1^K \epsilon \mathbb{R}^{K(p+1)}} \left[\frac{1}{N}\sum_{i=1}^{N} \log p_{g_i}(x_i) - \lambda \sum_{\ell=1}^{K} P_\alpha(\beta_\ell)\right] \tag{2}$$

Succinctly, the model learns a logistic regression classifier for each class, and then takes the highest probability outcome as the predicted class. The lasso penalty is used to reduce weights on k-mer features that are relatively insignificant.

Multi-class SVM We use the spectrum kernel, which converts sequence into k-mers (lengths of 2-10). For each length k, we have 4k features. We then run multi-class SVM with L2 regularization and L1 loss with the one-vs-the-rest strategy (R package: KeBABS[4]), which runs the SVM based on the equation below:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi(w; x_i, y_i) \tag{3}$$

Succinctly, the SVM first uses the kernel to quickly compute the k-mer feature values without having to maintain the k-mer feature space, then it runs an SVM that takes a single class as a positive class and the rest as negative classes.

Results

Multinomial logistic regression Running multinomial logistic regression, we find that error rates (defined as all misclassified examples) are high and remain very high, despite increasing the training set size and the k-mer length (Figures 1-3). However, there is enough of a downward trend that it is possible that more training data could be useful in decreasing the error rate. On average, we get errors around 0.8-0.9. However, we do find that the weights learned in logistic regression for the different k-mer features are heterogenous across the nine classes, suggesting that there are sequence differences across the classes that were learned by multinomial regression (Figure 4).

Figure 1: Error analysis of multinomial logistic regression (kmer=2)

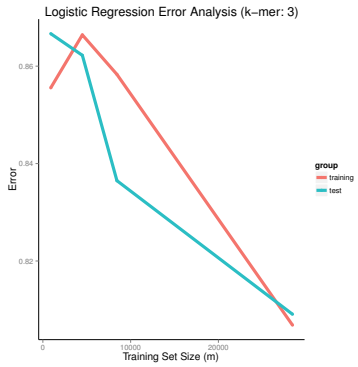

Figure 2: Error analysis of multinomial logistic regression (kmer=3)

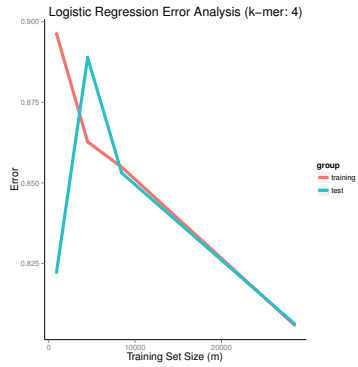

Figure 3: Error analysis of multinomial logistic regression (kmer=4)

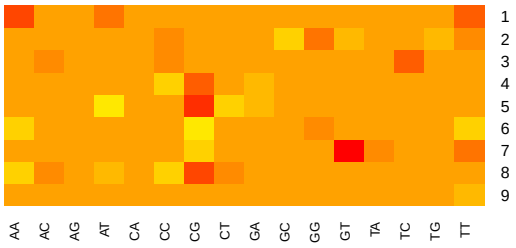

Figure 4: Heatmap of learned weights for the multi-class logistic regression using kmer=2.

Multi-class support vector machines Running multi-class SVMs, we find that the error rates are high and remain high, despite changing the k-mer length used and increasing the training set size (Figures 5-10). However, we again find that the weights learned by the SVM are heterogenous across the classes (Figure 11).

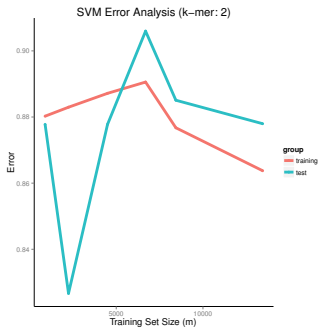

Figure 5: Error analysis of multi-class SVM (kmer=2)

Figure 6: Error analysis of multi-class SVM (kmer=3)

Figure 7: Error analysis of multi-class SVM (kmer=4)

Figure 8: Error analysis of multi-class SVM (kmer=5)

Figure 9: Error analysis of multi-class SVM (kmer=6)

Figure 10: Error analysis of multi-class SVM (kmer=7)

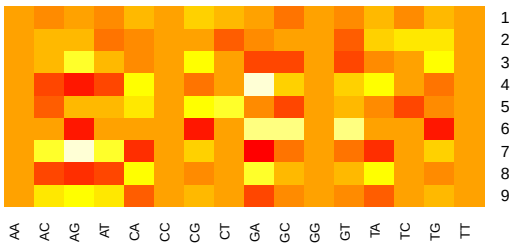

Figure 11: Heatmap of learned weights for the multi-class logistic regression using kmer=2.

Discussion

Error analysis suggests that more data could be useful for multinomial logistic regression but not for multi-class SVM. Notably, as the k-mer length increased for SVMs, we find significant over fitting - the training error drops to 0, while the test error stays around 0.80. However, there are only so many enhancers that are biologically active in keratinocyte differentiation, and so generating more examples of enhancers is not possible. We do note that there could be quality issues in the data, as the read depths of sequencing of the ATAC-seq datasets were low - increasing the sequencing levels could lead to better classes of data that lead to better classification and k-mer weights.

We do find that there are different weights on the k-mers for the different classes. This suggests that there are underlying sequence features that do lead to different enhancer activity patterns. However, the high classification error rate suggests that there are very few sequences in each group that actually have these sequence features (and thus get classified correctly in the test set). As such, further segmentation of the classes is likely necessary to actually have better predictions.

This problem of significant heterogeneity within the classes is demonstrated in Figure 12, where k-means clustering is performed on one class of enhancers. Using a k of 10, we see significant heterogeneity leading to at least 10 distinct clusters of sequence with differing numbers of each k-mer. Seeing this heterogeneity, further segmentation is necessary in each class to produce classification that is representative of the entire group of sequences in a class.

The high error rates seen in both multi-class learning techniques is somewhat expected, given the biological context. When considering the original segmentation of the classes (based on enhancer activity in three time points), it is highly likely that there are many different mechanisms at work in causing various enhancers to change activity patterns. In other words, there are many transcription factors and many combinations of transcription factors that are acting on groups of enhancers to change their behavior. As such, it is likely that many enhancers in a group may have very different motifs, despite acting in the same manner across time.

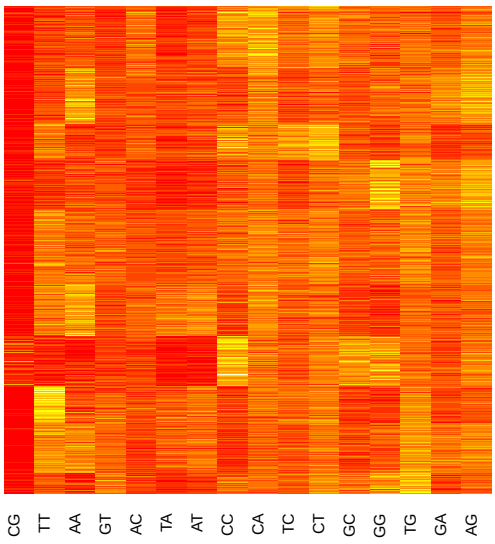

Figure 12: Enhancers from group 1 clustered into 10 groups using k-means clustering. There is notable sequence heterogeneity in the class.

Future Directions

Logistic regression was not performed on higher order k-mers due to computational constraints. Utilizing a kernel within logistic regression could be valuable and very interpretable. Additionally, a variety of other sequence-based kernels exist that consider mismatches and gaps. These are more biologically accurate kernels and may fit a better model. Noting the high error rate and the likelihood that biological heterogeneity plays a role in the error rate, segmenting the sequences further may also act as a filter for creating more homogenous classes. Finally, directly checking for motif instances within the regions may also provide important features.

Acknowledgments

Thank you to Andrew Ng and the TA's who put on a wonderful class!

References

[1] Ernst, J., & Kellis, M. (2012). ChromHMM: automating chromatin-state discovery and characterization. Nature Methods, 9(3), 215-6. doi:10.1038/nmeth.1906

[2] Heinz, S., Benner, C., Spann, N., Bertolino, E., Lin, Y. C., Laslo, P., ... Glass, C. K. (2010). Simple combinations of lineage-determining transcription factors prime cis- regulatory elements required for macrophage and B cell identities. Molecular Cell, 38(4), 576-89. doi:10.1016/j.molcel.2010.05.004

[3] Jerome Friedman, Trevor Hastie, Robert Tibshirani (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software, 33(1), 1-22. URL http://www.jstatsoft.org/v33/i01/

[4] J. Palme and U.Bodenhofer (2014). KeBABS - An R Package for Kernel Based Analysis of Biological Sequences.Unpublished. http://www.bioconductor.org/packages/ release/html/kebabs.html

[5] The Spectrum Kernel: A String Kernel for SVM Protein Classification. C. Leslie, E., Eskin, W. S. Noble. Proceedings of the Pacific Symposium on Biocomputing (PSB 2002), 2002.