Gordan Lab TF E2F1 DNA Binding Site Specificity Project Update

Sunwoo Yim

June 2015

1 Introduction

I currently work at the Duke Center for Genomic and Computational Biology, where I am part of the Gordan lab. As part of this lab, my goal is to better understand how transcription factors (TF), specifically TF E2F1, interact with different DNA binding sites. To do this, I use data that the lab has given me in order to apply a type of machine learning called support vector machines. So, with the data, I do a support vector regression with a program called LIBSVM in order to come up with a model that best predicts how well TF E2F1 binds to new and untested DNA sequences. My goal in this project is to come up with this kind of model that best predicts the testing data set with minimum error.

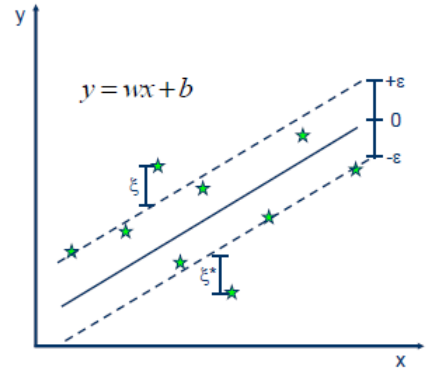
2 Project Description

This project aims to find the most accurate model to predict TF E2F1's binding specificity with different DNA sequences by:
1. Selecting the appropriate data subset from the train data that will be used to train the model.
2. Converting the selected data into the LIBSVM format that the program can use.
3. Creating a grid search that will find the optimal parameters to train the model with in order to find the most accurate prediction model.
5. Use this optimal model to find the accuracy with which it predicts the testing set.

3 Literature Review

Transcription factors are proteins that control the rate of transcription in the body by binding to DNA sequence binding sites that are located adjacent to the genes that the transcription factors regulate. TFs can either express the genes they regulate or prevent certain genes from being transcribed, thus enabling the same DNA sequences to perform different functions in the body. These transcription factors have a greater binding specificity to certain DNA sequences over others, and although TFs have been largely identified, there is still a lack of data as to which sequences a TF binds to[2]. In addition, it has been found that although a TF tends to bind more with sequences that have a specific core motif, the flanking sequences on either side of the core site has been discovered to greatly influence how well a TF binds to the sequence[6]. So, when testing how well a TF binds to a sequence, it is important to consider the surrounding sequences, and this is what gcPBMs do. gcPBMS are genomic-context protein binding microarrays, which is a method of comparing the relative specificities of a transcription factor binding to different tested sequences [6]. In a protein binding microarray, a custom-designed microarray is filled with different variations of DNA sequences, each variation in a separate slot, in which most of the sequences share the same core sequence but contain different flanking regions[5]. Then, the tested transcription factor is added and allowed to bind to the sequences. A fluorescent label is then added which attaches to the sequences that the transcription factor has bounded to, and then a laser is used to measure the relative signal intensities given off by the fluorescence quantitatively. A higher signal intensity corresponds to more instances of the TF binding to the sequence variant and thus a higher specificity to that sequence. The gcPBMs are different from universal PBMs in that they consider the genomic context of the DNA sequences rather than the sequence individually and by itself, allowing a more accurate determination of the ability of a TF to bind to the tested sequences.

The gcPBMs quantitatively measure the specificity of a TF to different sequences, and the value is given in an Expectation or E score. This score is given for each tested sequence and is relative to the other sequences rather than absolute as a result of the way PBMs measure relative fluorescence intensities[6]. These values with their corresponding sequences can then be put into a support vector machine (SVM), which uses statistical learning theory, or VC theory, to apply a learning algorithm to find a model that fits in with the training data given to it[1]. In particular, linear support vector regression (SVR) finds a model for the data by finding the line that encompasses the maximum number of vectors in the input data within a margin of tolerance that is defined as epsilon. It does this by penalizing any errors where the input vectors do not fit in the margin with a cost function, which can be varied to obtain a better model. This can be visualized as below[7]:

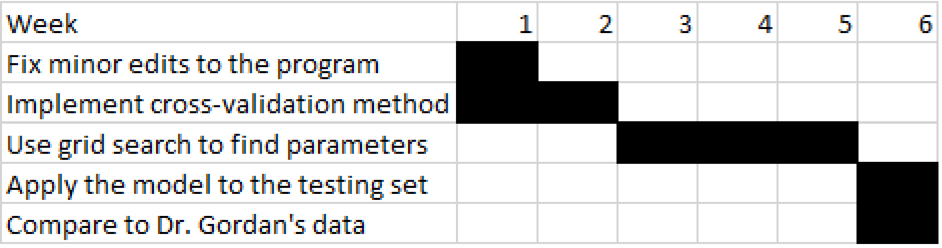


So, this kind of model can be found with a library called LIBSVM, which provides the tool necessary to perform linear SVR[3]. By varying the epsilon and cost parameters, better models can also be found for each data set. The accuracies of each model given a specific parameter setting can be found with a cross-validation procedure[4]. In k-fold cross-validation, the training set is divided into k equal sized subsets, and then a model is found using all but 1 of the k subsets of data. This model is then tested on the remaining subset in order to determine the accuracy of the model, and this process is repeated k times with each of the k subsets being tested with its corresponding model. After k-fold cross-validation is done, a single accuracy score can be determined from the k accuracies produced from the k iterations. In this way, the accuracy of each parameter setting can be determined. In order to find the optimal parameter setting, a grid search can then be used in which various pairs of epsilon and cost are tried and the pair with the best cross-validation accuracy score is selected[4]. Finally, the model with this parameter pair is used to predict the testing set, and thus the accuracy of the model to predict new data is determined.

4 Conclusion

Currently, I have finished selecting the data, converting it to the format LIBSVM can use, and creating the grid search method. Now, I need to use the grid search to find the optimal parameter setting pair to produce the highest accuracy in cross-validation. To do this, I must implement a cross-validation method and a way to produce a single accuracy score from this cross-validation. Then, I need to try different parameter settings and narrow down the best parameter setting possible. Finally, I will use this new model with the optimal parameter setting to predict the testing data set and find the accuracy of this final model.

5 Timeline



Week	1	2	3	4	5	6
Fix minor edits to the program	■					
Implement cross-validation method	■	■				
Use grid search to find parameters			■	■	■	
Apply the model to the testing set						■
Compare to Dr. Gordan's data						■

References

[1] Alex J. Smola et al. *A Tutorial on Support Vector Regression.* Statistics and Computing, 2004.

[2] Arttu Jolma et al. *DNA-Binding Specificities of Human Transcription Factors.* Cell, 2013.

[3] Chih-Chung Chang et al. Libsvm – a library for support vector machines.

[4] Chih-Wei Hsu et al. *A Practical Guide to Support Vector Classification.* National Taiwan University, 2003.

[5] Michael F. Berger et al. *Universal protein binding microarrays for the comprehensive characterization of the DNA binding specificities of transcription factors.* Nature protocols, 2009.

[6] Raluca Gordan et al. *Genomic Regions Flanking E-Box Binding Sites Influence DNA Binding Specificity of bHLH Transcription Factors through DNA Shape.* Cell, 2013.

[7] Saed Sayad. Support vector machine - regression (svr).