[CS229 Final Report]

Knowledge Based Reconstruction of the Transcriptional Regulatory Network in Yeast

Junhee Seok and Seok Chang Ryu

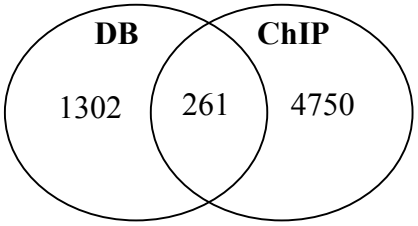
I. Introduction

After genome sequence is revealed, it becomes the significant issue to discover the function of genes. The transcriptional regulatory network (TRN), which describes interactions between genes, is the key point to reveal the function of genes. The TRN can be constructed from the experimental data such as gene expression microarray data. Many computational and statistical methods using various resources have been studied to construct the TRN. But none of them is quite successful because biology system is very complex and its experimental results are affected largely by environment and noise. For example, supervised learning approaches using SVM by Qian et. al.[Qian2003] is robust, but it does not consider network features by multiple TFs. Bayesian network approach by Friedman et. al.[Friedman2000] finds network structure, but it is based on joint probabilities which are hard to be obtained from small number of training sets.
In this project, we would like to provide network learning model using SVM and compare it with the previous SVM approaches.

II. Data Sources

Collecting Data

We used 643 microarray expression data sets for yeast which were collected and pre-processed by Stuart et. al.[Stuart2003]. We collected 3430 confident regulatory pairs from Proteome Database[Proteome]. Since this database is based on literatures and curated by human, it is highly confident. For biological validation, we used the most recent ChIP-chip experiment data by Harbison et. al.[Harbison2004]. We collected 5821 binding pairs with P-values less than 0.001.



Processing Data

We excluded genes which have missing data points more than 30%. We only considered 5940 genes among total 6646 genes. For 5940 genes, there are missing data points of 1.3%. Since it is statistically low enough, the missing points are set as zeros.

Considered Transcriptional Factors

Since the database has more TFs which are not considered ChIP-chip experiments, direct comparison between the database and ChIP-chip experiments is not fair. Therefore, we only considered 187 common TFs of the both sides.

III. Classification using SVM

Training Sets

We chose 1563 confident pairs, which have common TFs with ChIP-chip data, from the database as positive training sets. Negative training sets are chosen randomly excluding

pairs of the database. Because of the imbalance problem, it is common to choose 10 times more negative sets than positive sets[Qian2003]. But considering computing time, we chose 3 times more negative sets.

Features and Kernel

Qian et. al. concatenated data points of the TF and the target gene and made a feature of double data points. To model co-expressed patterns and reduce the dimension of a feature, we made a feature vector by multiplying data points of the TF and the target gene of the same experiments. Then, we used a simple linear kernel. Let E_{gi} be the expression level of the gene g at the experiment i. Then the feature vector and kernel is like below.

$$\phi(p_{tg}) = \left[E_{t1}E_{g1},\ \ldots,\ E_{tn}E_{gn}\right] \quad (1)$$

$$K(p_{tg}, p_{sh}) = \phi(p_{tg})^T \phi(p_{sh}) \quad (2)$$

IV. Learning Transcriptional Factors using SVM

Review of Previous Approaches

As shown before, the pure supervised classification does not work well. One of the possible reasons is that it cannot model regulations by multiple TFs. To model multiple TFs and complex networks, Bayesian network learning with posterior probabilities is commonly used[Friedman2000]. But the Bayesian approach is not successful because there are not enough data sets to get posterior probabilities confidently. We need to combine robustness of SVM and network learning of the Bayesian networks.

Modeling Multiple TFs: Features and Kernel

Let a gene g be regulated by TFs $t_1,\ \ldots,\ t_N \in T_g$. To model regulations by AND and OR operations, we need to consider all co-expression patterns,

$$E_{gi}\prod_{t_k \in S_k} E_{t_k i},\ S_k \in \{\text{all subsets of } T_g\} \quad (3)$$

It is hard to get the exactly same expression with the above, but we can get the similar expression easily like below.

$$\left(\sum_{k=1}^{N} E_{t_k i}E_{gi}\right)^N \quad (4)$$

Let $\phi_i(g)$ be a feature at the experiment i for the gene g which is regulated by TFs $t_1,\ \ldots,\ t_N$. Then,

$$\phi_i(g) = \left[E_{t_1 i}E_{gi},\ \ldots,\ E_{t_N i}E_{gi}\right] \quad (5)$$

By taking a kernel

$$K_i(g_1, g_2) = \left(\phi_i(g_1)^T \phi_i(g_2)\right)^N \quad (6)$$

we can get expressions like the equation (4). Finally,

$$K(g_1, g_2) = \sum_i K_i(g_1, g_2) \quad (7)$$

If two genes have different number of bound TFs, we can match the dimension by repeating average of other TFs.

Learning Classification Boundary using SVM

Using the above kernel and training sets of III, we can learn classification boundary. The prediction function with learnt parameters can calculate a confidence score for a test sample.

Learning TFs using SVM

TFs bound to a gene can be learnt using a greedy algorithm. Let $P(g,T)$ be the prediction function for a gene g with a set of TFs T.

1. Initialize $Score = 0,\ T_g = \varnothing$.
2. Get $MaxScore = \max_{\text{for all possible TFs}} P\left(g, T_g \cup \{t_i\}\right),\ t_i \notin T_g$ and TF t^* which maximize the score.
3. If $Score < \gamma \cdot MaxScore$, update $T_g := T_g \cup \{t^*\},\ Score = MaxScore$, and repeat the step 2. Otherwise, the algorithm is finished.

It is the similar approach with works by Friedman et. al., but there are two different points. One is that we are using confident scores instead of posterior probabilities, and the other one is the reduction factor γ. It compensates effects of multiple TFs which regulate genes independently.

V. Results

The following tables shows testing results for the pure SVM and SVM network learning.

	Pure SVM	NL SVM(N=3)
Training Sets Positive/Negative	1563/1563x3	710/710x3
Test Sets	730497	5940
Positive Results	53099	8851
Coverage for 5011 ChIP-chip data	870(17.36%)	182(3.63%)
True-positive	1.64%	2.06%

In the pure SVM, 1563 pairs are chosen from the database as positive training sets and 4689 pairs are chosen randomly as negative sets. 730497 candidate pairs between 123 common TFs and 5940 genes except self-regulation are tested. As results 53099 pairs are determined as true regulation pairs. Among them, 870 pairs are included in ChIP-chip data. Its true positive rate is 1.64%. SVM network learning is used for up to 3 TFs. Positive training sets are chosen from the database. If a gene has more than 3 TFs, 3 mostly correlated TFs with the gene in the sense of gene expression are chosen. Negative training sets are chosen randomly with similar distribution as the positive sets. 5940 genes are tested. As results, 3663 genes have one TF, 1633 genes have two TFs and 639 genes have three TFs. Among total 8851 pairs, 182 pairs appear in ChIP-chip data. Its true positive rate is 2.06%.

For both methods, the true positive rate is too low. It is mainly because ChIP-chip data and the database don't show a good correlation in the view of the gene expression profiles. As shown in Figure 1, score distributions between ChIp-chip and Random pairs are similar. Another reason is the gene expression profile used as features does not indicate clear difference between positive and negative training sets. As shown in Figure 2 and 3, correlation coefficients of positive sets and negative sets have similar distribution. Even though their average is slightly different, it is not enough difference for genome-wide prediction. One possible reason is that the gene expression data is old. They were measured in 1999 and 2000 with old method, which has lots of noise.

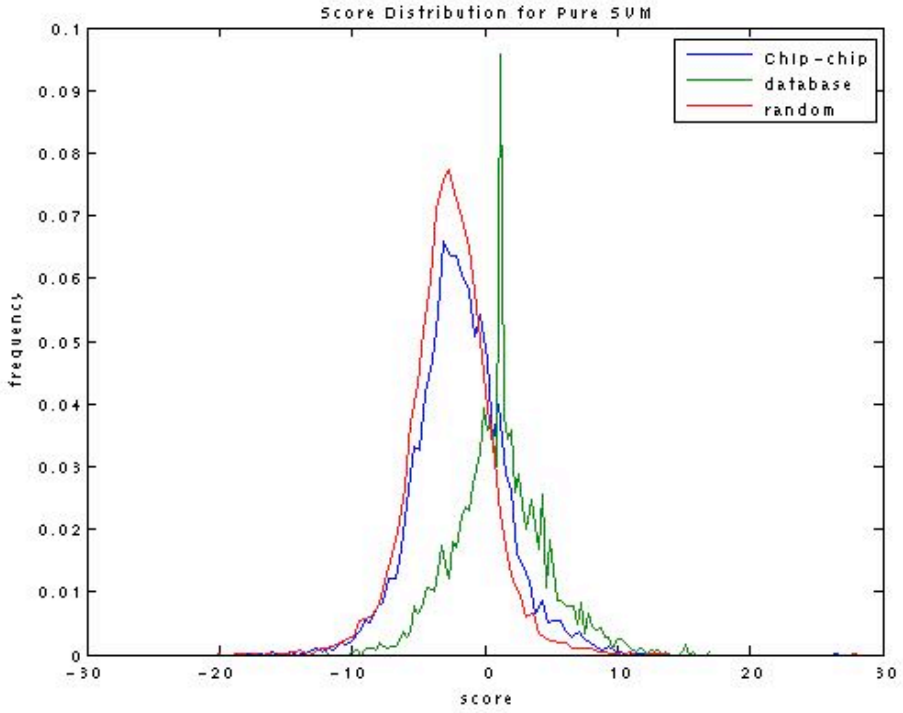


Figure 1. Score distributions for ChIP-chip, Database and Random pairs in the pure SVM

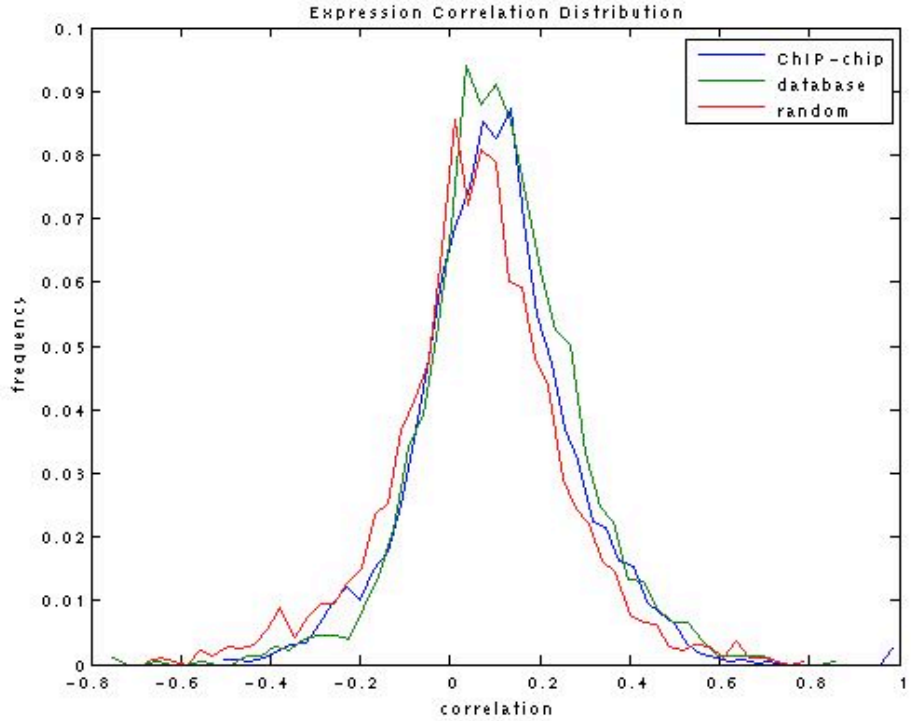


Figure 2. Correlation coefficient distribution of gene expression for ChIp-chip, Database and Random pairs

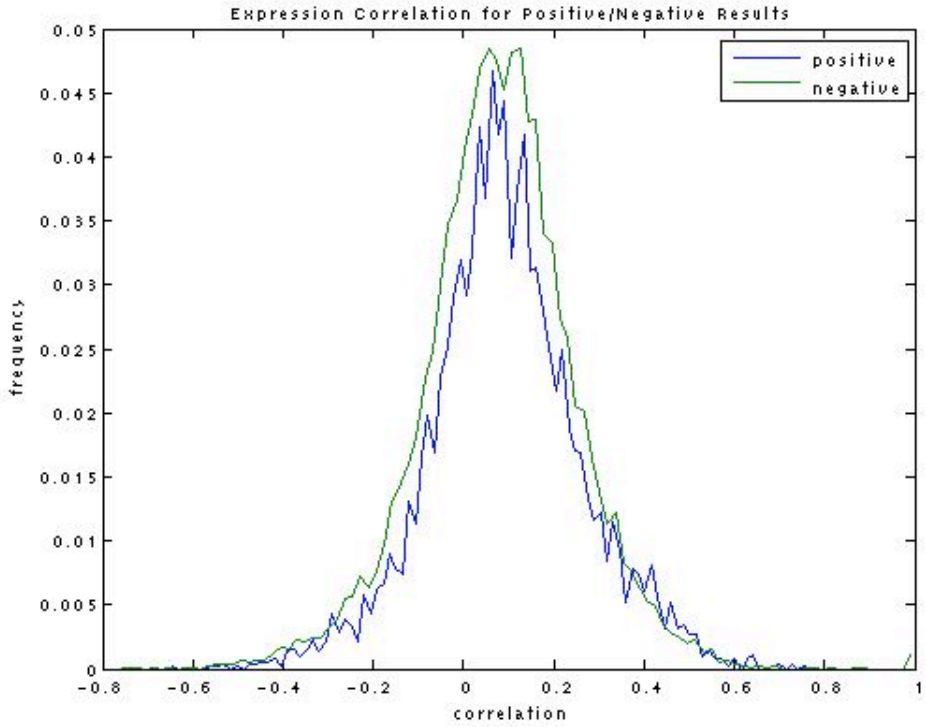


Figure 3. Correlation coefficient distribution of gene expression for positive and negative results from the pure SVM methods

VI. Conclusion

As shown before, three data sources, the knowledge base, gene expression data and ChIP-chip binding data are less correlated. For those uncorrelated data sets, the supervise learning does not show good performance. For future works, we can test other machine learning methods for gene expression data and compare with ChIP-chip data. It will verify the performance of machine learning methods.

Reference

[Friedman2000] N. Friedman, et. al., "Using Bayesian Networks to Analyze Expression Data", Journal of Computational Biology, 2000

[Harbison2004] C. T. Harbison, et. al., "Transcriptional Regulatory Code of a Eukaryotic Genome", Nature, 2004

[Proteome] https://www.proteome.com/proteome/Retriever/index.html

[Qian2003] J. Qian, et. al., "Prediction of Regulatory Networks: Genome-wide Identification of Transcription Factor Targets from Gene Expression Data", Bioinformatic, 2003

[Stauart2003] J. M. Stuart, et. al., "A Gene-Coexpression Network for Global Discovery of Conserved Genetic Modules", Science, 2003