Annotating Gene Functions by Spectral Clustering for Combining Gene Expressions and Sequences

Limin Li\textsuperscript{1} motoki shiga\textsuperscript{2}
limin@hkusua.hku.hk \hspace{2cm} shiga@kuicr.kyoto-u.ac.jp
wai-ki ching\textsuperscript{1} hiroshi mamitsuka\textsuperscript{2}
wching@hkusua.hku.hk \hspace{2cm} mami@kuicr.kyoto-u.ac.jp

\textsuperscript{1}Advanced Modeling and Applied Computing Laboratory, Department of Mathematics, The University of Hong Kong, Pokfulam Road, Hong Kong
\textsuperscript{2}Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji 611-0011, Japan

Keywords: gene function annotation, spectral clustering, microarray, sequence

1 Introduction

Annotating gene functions is a fundamental issue in the post-genomic era. A typical procedure for this issue is first clustering genes and then assigning functions of unknown genes by using known genes in the same cluster. A lot of genomic information are available for this issue, but two major types of data which can be measured for any genes are microarray expressions and sequences, both of which however have their own flaws. Thus a natural and promising approach for gene annotation is to combine these two data sources.

We developed an efficient gene annotation method with three steps containing spectral clustering over the integrated clustering cost for each data source. We examined the performance of our proposed method from viewpoints of clustering and annotations. All experimental results indicate our performance advantage over possible clustering/classification-based approaches of gene function annotation, using expressions and/or sequences.

2 Method

Inputs of our method are gene expression profiles and genome sequences. For genome sequences, we checked similarities between all pairs of sequences by the Smith-Waterman algorithm. And then we converted sequences to a network by generating edges for pairs of genes with high similarity scores.

Our method has two steps: i) clustering genes by a spectral method, ii) annotation of gene functions using generated clusters. Clustering is performed by [4], where expression dataset and a network of sequence similarities are inputs. The weight for two data sources can be optimized by using a clustering cost [4]. An input parameter is the number of clusters $K$ only.

In step (ii) for annotating gene functions, we check the overlap between a cluster generated by step (i) and a function term of Gene Ontology (GO) [1]. Our overlap measure between a generated cluster $C$ and a known function $F$, which we call $F$-value, is defined as $F$-value$(C,F) = \frac{\text{#genes in } C \text{ and } F}{\text{#genes in } C}$. Finally the function of a cluster $C$ is assigned function $F^*$ whose $F$-value is largest over all known functions. function $F^*$ can be assigned to unknown genes which are in cluster $C$. Then function $F^*$ can be assigned to unknown genes which are in cluster $C$.

We note that our F-value is a better measure for gene assignment than $P$-value given by GO-term finder [2], because $P$-value depends on the size of known functions too much, which causes incorrect gene assignment.
3 Results and Discussion

We analyzed yeast genome datasets by our method and all possible competitive methods. Expression datasets were taken from Gene Expression Omnibus (GEO) and we picked up datasets where the number of experiments is over than 300. And sequence similarity scores were taken from Saccharomyces Genome Database (SGD) [3]. We extracted genes in the maximum connected component only in the network.

We first examined the performance of Step (i) by Normalized Mutual Information (NMI) between generated clusters and standard functions. As standard clusters we use major fifteen terms of Gene Ontology (GO) [1]. In this experiment, we analysed 487 genes categorized in one or more the major GO terms in this experiment. Table 1 shows performance results of our method with $K = 10$. Experiment names in Table 1 correspond to datasets used in [5]. In Table 1, “Exp.” and “Sequence” are the NMI obtained when we used expressions only and or sequences only, respectively. “Bot” shows the NMI obtained by using both data sources. This results shows that our method could optimally combine gene expressions and sequences to outperform the case of using a single data source.

We then examined the performance of Step (ii). In this experiment, we used 582 genes labeled by any GO terms including fifteen major terms. We implemented a 100×5-fold cross validation, in which we divided all genes into five blocks of roughly equal size, and at each trial, four out of five were used for training and the remaining one was for test. We repeated this division 100 times and the results were averaged over the 100 runs. Table 2 shows Area Under the ROC Curve (AUC) by our method and that by SVM for the Spellman’s expression dataset and sequence dataset. Table 2 shows that our method by using $F$-value is better than our framework using $P$-value and SVM. And our method using both gene expressions and sequences outperform another competitive method.

Acknowledgments

This work was supported by MEXT KAKENHI 20700269, BIRD of JST, Japan, and RGC Grant 7017/07P, HKU CRCG Grants, Hung Hing Ying Physical Sciences Research Fund and HKU Strategic Research Theme Fund on Computational Physics and Numerical Methods.

References