공학석사 학위논문

Identifying stress-related genes and predicting stress types in a heterogeneous time-series data

이질적 시계열 유전자 데이터의 스트레스 연관 유전자 및 스트레스 예측 기법

2018년 8월

서울대학교 대학원
컴퓨터공학부
강동원

Identifying stress-related genes and predicting stress types in a heterogeneous time-series data

지도교수 김 선

이 논문을 공학석사학위논문으로 제출함
2018년 6월

서울대학교 대학원
컴퓨터공학부
강동원

강동원의 석사학위논문을 인준함
2018년 6월

위 원 장 Srinivasa Rao satti (Seal)
부 위 원 장 김 선 (Seal)
위 원 Bernhard Egger (Seal)

Abstract

Identifying stress-related genes and predicting stress types in a heterogeneous time-series data

Dongwon Kang
Computer Science and Engineering
The Graduate School
Seoul National University

As gene expressions which contains data of big dimension begin to be formed, the necessity of integrated analysis of time series gene expression data is emerged. However, analyzing gene expression data is a new time series analysis problem that is not addressed in existing computer science as there are not only much time series data with few time points though it has many features but also its heterogeneous time series analysis problem in which the measurement points and experiment conditions are different with data of disorganized form, such as raw text and expression data of mixed time series.

In this study, I introduce feature embedding method with such heterogeneous time series data in form of minimizing data loss, and introduce logical relevance layer which indicates stress-gene

correlation weight which is learned with cross-entropy and group effect. This layer also used in stress prediction model with logical filter layer on top of this model to get output in logical probability, and this layer is learned with CMCL (Confident Multiple Choice Learning) loss to prevent parameter overfitting.

This model revealed many Gene Ontology related to given stress with high stress-gene correlation weight. Also, to find out whether the genes which are only responding with specific stress are ranked higher, I compared gene rank for each stress of ordinary Fisher's method with my method, and I found many genes which has multiple GO term, which means correlated to multiple stimulus, are downranked in my method compared to combined limma p-value of each time series data using Fisher's method, which means this model gives high rank in genes which only respond to specific stress. Furthermore, this prediction model showed excellent performance compared to classical prediction methods like Random Forest and SVM.

Therefore, this result suggests new method for selecting gene only responding to specific stress type and predicting stress using time series data with small amount of time points and replication.

Keywords : Arabidopsis, Microarray, Time series, Machine Learning

Student Number : 2016-27463

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent advances in biological sciences, such as microarrays and RNA-seq, have enabled the simultaneous measurement of the expression levels of tens of thousands of genes in cells. Thanks to these advances in biological sciences, studies have been carried out to track changes in gene expression according to time series for specific stresses. These data used in studies are stored and released in databases such as GEO[1] and ArrayExpress[2].

Since gene expression data accumulates gradually, studying the analysis of gene expression data becomes possible. Identifying plant response to abiotic stress genes with microarray data had been studied based on various machine learning methods, such as Linear Regression, naive Bayes, PCA, KNN [3]. Also, predicting biological processes had been attempted based on simple neural network [4,5] and ensemble model [6]. Based on these studies, finding marker gene detection and predict traits using naive Bayes method had been devised [7]. However, given methods only perform on gene expression data for single sample, not on time-series data composed with multiple samples. Furthermore, there are no methods processing integration analysis on these data.

By analyzing time-series gene expression data with integrated stress, I demonstrated that:

- Identify differentially expressed biomarker genes and time-series microarray data with small amount of time points.
- Predict stress with time-series data with small amount of time points using biomarker genes.

1.1 Gene Expression Data

Gene expression is an information which is used in the synthesis of a function gene product, e.g. synthesis of proteins or regulating pathways. Such process is all common for all known life – plants, animals, bacteria, etc. In genetics, gene expression is the most basic level as its expression effects to the observable phenotype. Therefore, studying gene expression data is important to find out the way to controlling or changing phenotype, which is called genetic engineering.

1.1.1 Microarray Data

Microarray is a 2D array on a solid substrate to get expression level of thousands of genes, which is one of the methods extracting gene expression level like RNA-seq. Microarray data should be uploaded to public gene expression database in case of paper publishing using that data. Because of that, many public microarray data are generated and many microarray databases created providing microarray expression data with metadatas, such as GEO[1] and ArrayExpress[2].

1.1.2 Time-series Microarray Data

As technology improvement makes biochip cost low, genomic experiments with multiple time points and replications are rising, generating time-series microarray data. Such data are composed with multiple time points, which makes data even bigger than single microarray data, causing new problem of processing these data. However, as these are new type of format and processing microarray data multiple times is still expensive, quantity of these data is low and these data are not well formatted, such as in form of raw data without being aligned by time or packed together with different samples on different condition, which is in messy form to be used for experiment.

1.2 Motivation

As I addressed at background section, there are many algorithms analyzing gene expression data. However, they still don't meet all requirements in feature selecting and predicting traits with time-series microarray data.

1.2.1 Limitation of current biomarker detection methods

In case of prediction model based on SVM, it is only suitable for binary-class problems. and PCA itself is not suitable to predict traits as its axis differs from data input. and RF does sparse classification, so it's hard to interpret importance of each gene related to stress.

Naive Bayes is proposed as a method to not only solve such problems but also design prediction model [7]. But still no method exists to get stress-related genes and doing stress prediction with time-series gene expression data. We propose model identifying stress-related genes and predicting stress on time-series gene expression data. Such methods only get gene expression data with single time point as input data and marks genes with highly expressed one, makes it unable to find specific genes differently expressed by specific stimulus or stress. To deal with this issue, we propose model identifying stress-related genes and predicting stress on time-series gene expression data.

1.2.2 Difficulty of analyzing time-series data

However, analyzing time series data is a hard problem for the following reasons.

- Small amount of sample data

  Definition about time-series data is insufficient, as many time-series data is provided with microarray data that is mixed with other conditions or time-series inconsistently.

- High dimension of features

  As microarray experiment costs a lot, most microarray time series experiments have very short time points, and it makes hard to process time-series microarray data with deep learning techniques, like RNN. There had been attempts to analyze such time-series data Clustering genes [8] and extracting differentially expressed genes [9], But no method exists with stress prediction

and identifying key genes reacting to stress.

To solve these problems, I propose new time-series analyzing method in this study.

Chapter 2

Materials and Methods

In this section, I defined time series data and time series dataset, and described model which finds stress-related genes and predicts stress from given time series data. Time series data is hard to analyze due to it's large dimension and small amount of time points. I suggest new embedding method which separates up-down signal to prevent signal cancelling for data with small amount of time points. Generated feature vector is put into my model to find genes responsive to specific stress, which uses logical regression layer, cross-entropy and group effect to find these genes. Group effect gives penalty to genes which is responsive to multiple stress, assuming that gene does not have enough novelty as biomarker for specific stress. Assuming my logical regression layer means relativity between gene and stress, I used transposed form of that layer in stress prediction, and append logical filter layer to get probability of that stress. I used CMCL (Confident Multiple Choice Learning) loss to prevent its parameter overfitting. The total architecture of this model is described in Figure 1. The detailed description of the model is written in Section 2.2 and 2.3.

2.1 Time series data

2.1.1 Definition

Time series gene expression data (hereinafter referred to as time series data) is data consisting of m samples obtained by measuring the expression levels of gene group $G_k = \{g_{k1}, g_{k2}, \ldots, g_{kn}\}$ at time $T_k = \{t_{k1}, t_{k2}, \ldots, t_{km}\}$ for cells having a specific phenotype $F_k = \{stress: heat, genotype: wild, tissue: root\}$, and its gene expression data is represented by $\underset{(n \times m)}{D_k}$ matrix. For example, data on the expression levels of 20,000 genes at 0, 1, and 6 hours after treatment of cold stress in leaves (phenotype), indicate that the phenotype origin is the leaf and the expression level matrix is the number of genes (20,000) x measurement time (3 periods).

This study deals especially with `heterogeneous' time series dataset. The heterogeneous time-series data refers to time-series dataset in which the viewpoint, timepoint and the phenotype are heterogeneous, and the specific characteristics are as follows.

- Different time points for each time series data

  For example, time series 1 and time series 2 might be different from each other like $T_1 = \{0,1,3\}, T_2 = \{0,3,6,12\}$

- Different phenotype depending on the experimental condition

  Each experiment has different condition, like ecotype, genotype, temperature, osmotic, etc., and it makes very big dimension for time series data, which makes hard to solve exact feature with

small number of samples.

2.1.2 Dataset

As I explained in Section 1.1.2, there are no existing time-series microarray dataset, I packed time-series dataset into the form to be used for this experiment. The total dataset for this experiment is 138 time series samples from NCBI and Arrayexpress, and 108 samples were used as learning set and 30 samples were used as test set, consisted with heat 7 stress type and 20 cold stress type and 3 salt stress type as these stress types are redundant. Most stress type in this dataset was heat and cold, and I used them as test set to make amount of each stress type evenly in my learning set. Most of the time series contains only two time points (Figure 1). Every time point in each time series data contains at least 2 replications, which means count of samples with duplicated time point. Also, all time point is sorted in ascending order for ease of feature extraction. This database is available at http://epigenomics.snu.ac.kr/plant_stress_db/.

2.1.3 Feature embedding

As each time series data contains heterogeneous time point, it's hard to use microarray data directly to learning features. Therefore, I preprocess to extract features for each time series data. Using time series data with time point $\mathrm{T_k}$ and gene expression data $\mathrm{D_k}$, I generate feature vector $\mathrm{X_k} = \{x_{k1}, x_{k2}, \dots, x_{2kn}\}, \mathrm{X_k} \subset \{0,1\}$, which indicates up signal for odd element, and down signal for even element. That means

each gene is encoded into up and down signal, which means element size get multiplied by 2, but total gene expression matrix is reduced into single vector for each time series data. This makes data not only to more fixed data form but also prevent data loss by preventing feature loss due to summation of up(1) and down(-1) signal, which results in zero feature.

2.1.4 Limma and Foldchange

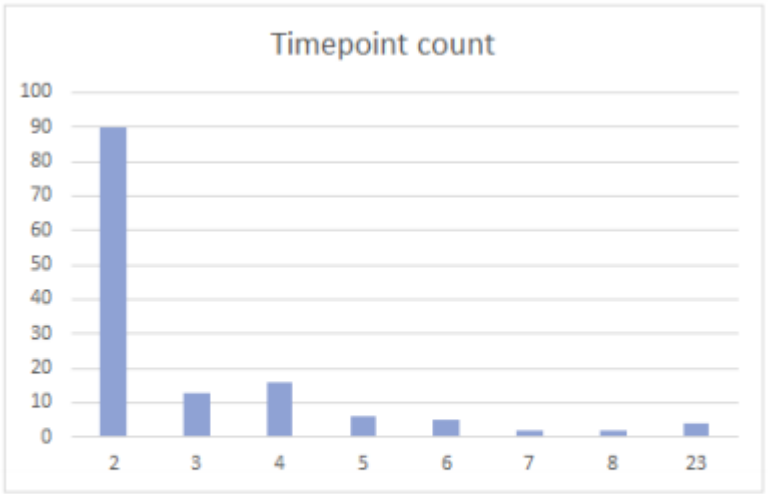
Best performing method will be selected for my feature selection algorithm out of these two feature embedding methods.

- Foldchange

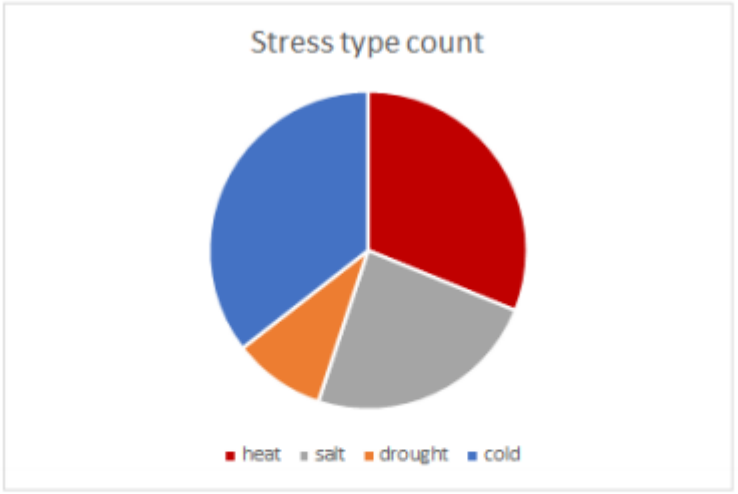
  After calculating average of each timepoint in case of replication of timepoint exists, I calculate gene expression difference between last and first time point. If difference of specific gene expression is over specific threshold, e.g. 0.8, the gene is marked as differentially expressed gene(DEG). I choose 0.8 as threshold at which at least 20 genes are marked as DEG within dataset.

- Limma

  As limma is good tool for identifying DEGs of time series data based on solid statistical theory, using this tool for feature embedding can also be a good idea. Whether a specific gene is differentially expressed or not is judged by limma's adj p-value with threshold 0.05, and t-value is used for is this gene expression up-regulated or down-regulated by stress.



2	3	4	5	6	7	8	23
90	13	16	6	5	2	2	4



heat	salt	drought	cold
43	33	13	49

Figure 1. Statistical summary of time-series dataset. Most dataset contains only two timepoints, and most stress are heat and cold stress.

2.2 Stress-related gene detection model

2.2.1 Logical correlation layer

The model in Figure 2 is composed with two parts; stress related gene learning model and stress prediction model, and logical correlation layer is shared between two models. Stress related gene learning model outputs loss function by comparing real value from generated value from result of grey box, of which each indicating output of stress-gene relation weight in Figure 3. Stress prediction model gives probability of stress in Figure 4. It's easier to learn weight by comparing each gene, rather than comparing stress as it not only causes sparse weight matrix but also is useless for stress-gene relativity indicator. Therefore, my first goal is assigning correlation weight with gene to specific stress by learning logical correlation layer consisted with single neural network W and sigmoid function, which is supposed to be relativity between specific gene and stress type. Given gene feature vector $\mathrm{X_k}$ and stress label vector $\mathrm{Y_k} = F_k^{stress}$, the gene stress class probability model can be written as:

$$\begin{aligned} \mathrm{X'_k} &= sigmoid(\mathrm{Y_k W}) \\ &= \frac{1}{1+\mathrm{e}^{-\mathrm{Y_k}W}} \end{aligned} \tag{2.1}$$

$$\mathrm{W} = \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{l1} & \cdots & w_{ln} \end{pmatrix} \tag{2.2}$$

Sigmoid function is often used to work as logical function as its output

is regulated between 0 to 1, therefore I use sigmoid function to make matrix W work as logical correlation matrix. $\mathrm{Y_k}$ is stress prior, which is encoded as one-hot vector with vector of L size indicating whole count of stress type. Therefore, by above equation (2.1), each feature generates vector of N size, e.g. $\mathrm{X'_k} = (\mathrm{x_{k1}}, x_{k2}, \dots, x_{kn}) = Y_k W = sigmoid(w_{l1}, w_{2l}, \dots, w_{nl})$. This vector contains activation of features between 0 to 1, which should show consistent pattern (up / down) through same type of stress l.

To learn this relation weight matrix, I use cross-entropy for penalty function between generated vector $\mathrm{X'_k}$ and real feature vector $\mathrm{X_k}$. It is well known that minimizing cross-entropy is good method for logical regression.

Thus, objective function for learning W is written:

$$\mathrm{loss_w} = \sum_{k=1}^{K} \Big((\mathrm{X_k} \log\big(sigmoid(Y_k W)\big) + (1 - X_k)\log(1 - sigmoid(Y_k W)\Big) \tag{2.3}$$

2.2.2 Group effect

With only cross-entropy loss, some gene weight would be tied to same value if there are same number of samples activated for specific feature. Also, there might be a gene which reacts to every stress, which is undesirable for biomarker. To solve these problems, the model gives group penalty to each feature weight if it's related to many stresses. For example, defining $\mathrm{n_{th}}$ gene weight for specific stress l as $\mathrm{g_{nl}} = \max(w_{1,2n}, w_{1,2n+1})$, suppose some gene's stress vector is $\mathrm{g_1} =$

[1,0,0,0] which is responding to single stress. Then, group effect cause by this gene is $\left(\sum(g_1)\right)^2 = 1$. And suppose other gene's stress vector is $g_2 = [1,1,0.5,0]$, which responds to multiple stress types. In that case, group effect of this gene is $\left(\sum(g_1)\right)^2 = 6.25$, which is bigger than previous gene. Such big loss caused by group effect will regulate feature weight which helps to getting genes only responding to specific stress. Total equation of group effect is written:

$$\text{loss}_{group} = \alpha \sum_{n=1}^{N} \left(\sum_{l=1}^{L} g_{nl}^2 \right)^2 \tag{2.4}$$

Pseudo-parameter α indicates how much a gene affected by group-effect, and I use total loss of mean of sigmoid-entropy and group loss for bayes learning. I processed model test with $\alpha = 0.06$, which I expect downregulation of weight value about 0.07 for two-stress related feature with original weight of 0.8.

2.3 Stress prediction model

2.3.1 Transposed logistic correlation layer

In stress-gene correlation learning model, I defined w_{lj} as relativity between stress l and gene j. So, it's proper use transposed logical correlation layer to predict stress from feature vector. In prediction model, definition of probability model as written:

$$A_{\mathrm{k}} = sigmoid(X_k W^T) \tag{2.5}$$

$$A_{\mathrm{kl}} = sigmoid\left(\sum_{i=1}^{N} x_{ki} w_{il}\right) \tag{2.6}$$

Matrix W is brought from stress-gene correlation layer as I mentioned above. Therefore, by using feature vector $\mathrm{X_k} = (x_{k1}, x_{k2}, \dots, x_{kn})$ of size N, I get sum of stress correlation weight $\mathrm{A_k} = (a_{k1}, a_{k2}, \dots, a_{kl})$ which I expect to be get higher if many stress-related features are activated.

2.3.2 Normalizing

Although I can assume stress-related feature activation through vector $\mathrm{A_k}$ in Section 2.3.1, its value is much different from each time-series sample, e.g. some sample shows activated feature about 8000 while some other sample shows few activated features about 100. Therefore, I did normalizing by sum of feature count as written:

$$A_{\mathrm{k}}^{norm} = \frac{A_k}{\sum_{i=1}^{N} x_{ki}} \tag{2.7}$$

By normalizing, I can get average stress-feature correlation weight, which is suitable form for logical filter as variance is quite big between embedded features of my time-series microarray dataset. Also, it prevents false positive (which are not affected by my 4 learned stress labels) as it indicates absolute average weight value, rather than

relative indicator like softmax which fits summation of vector into 1 without considering its absolute value.

2.3.3 Logistic filter

Now I do logical filtering to convert average weight into logical probability as written:

$$g_k(A_k^{norm}) = \frac{1}{1 + b_l \times \exp(A_k^{norm} - a_l)} \tag{2.8}$$

where a and b is general vector parameter of size L of logistic model g(x).

2.3.4 CMCL loss function

To learn parameter of this logistic filter layer, I first normalized result of logistic filter to make learning easy by regularizing mean of vector. After that, I used Confident Multiple Choice Learning(CMCL) loss function [10]. The reason of CMCL loss function is it minimize loss for positive label and maximize entropy for negative label, which prevents overfitting this model to "false negative" that occupies lots of loss. CMCL loss function for my model is written as:

$$loss_{CMCL}\left(Y_k, g(A_k^{norm})\right) = \sum_{k=1}^{K}\left(\left(1 - A_{kY_k}^{norm}\right)^2 - \beta \sum_{l \neq Y_k}^{L} \log(A_{kl}^{norm})\right) \tag{2.9}$$

Pseudo-parameter β prevents model parameter overfitting as it's bigger (mostly number of samples). In here I used $\beta = 0.01 \approx 1/112$.

2.4 Existing methods for performance comparison

In this section, I describe three methods used for performance comparison with my model, Fisher's method and Random Forest and SVM. Fisher's method is used for stress-gene correlation comparison, and Random Forest and SVM is used for stress prediction comparison. All models use input data as my embedded feature data.

2.4.1 Fisher's method

To compare gene ranking, I use traditional method, fisher's method, to integrate p-value of each gene for each stress types(heat, cold, drought, salt). Fisher's method is:

$$X_{2k}^2 \sim -2\sum_{i=1}^{k}\ln(p_i) \qquad (2.10)$$

After calculating Fisher p-value with limma, I ranked their score in descending order to rank most responsive gene first. After that, I compared its rank with one of my model to see this model finds responsive genes well and gives more score to gene which only

responds to specific stress.

2.4.2 Random Forest and SVM

With embedded feature dataset, I processed Random Forest and SVM to compare prediction accuracy with my stress prediction model, consisted with logistic correlation model and logistic filter learned with Cross-entropy with Group effect and CMCL loss. Random Forest and SVM is executed in Weka.

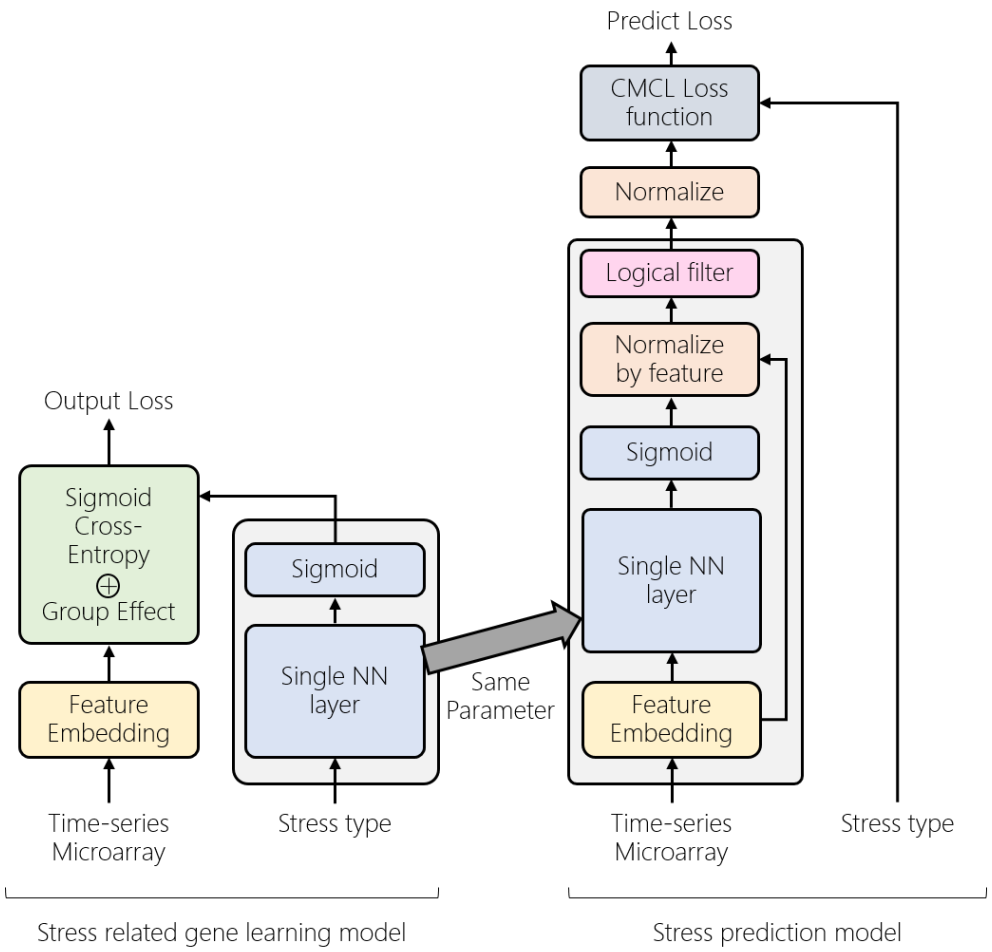


Figure 2. Total architecture of the model. Logical regression layer, which is consisted with sigmoid function and single neural network layer, shares parameter in two models. For ease of parameter learning, stress related gene learning model is first learned before learning stress prediction model.

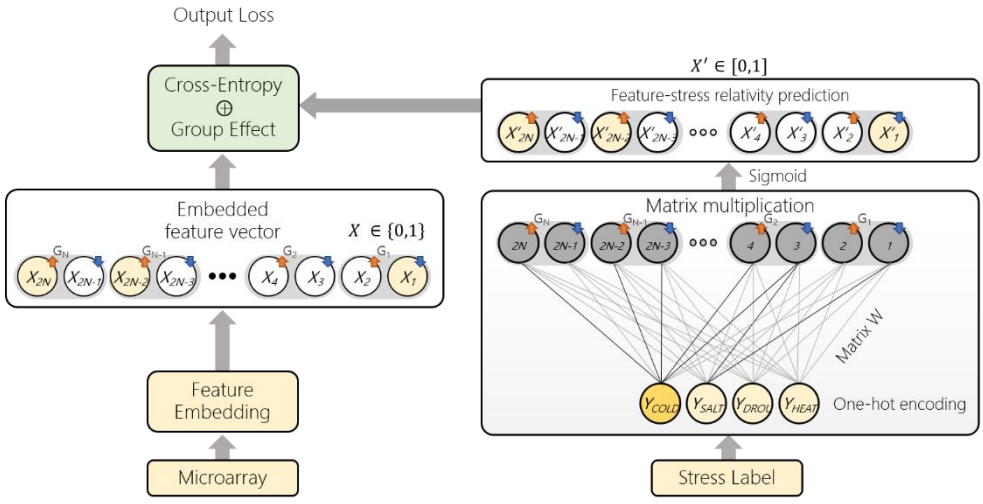


Figure 3. Stress-related gene detection model diagram.

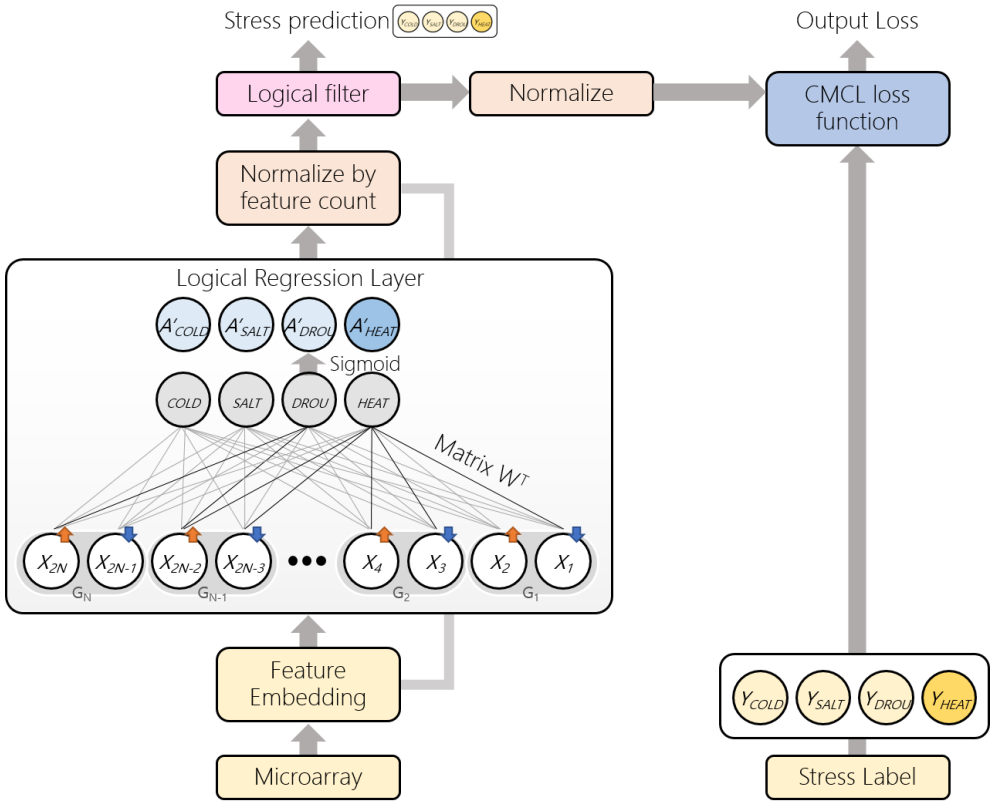


Figure 4. Stress prediction model diagram.

Chapter 3

Experiments and Results

In this section, I analyzed genes which are high-responsive to stress with Gene Ontology[11] to prove my model finds correct biomarkers. Also, I compare gene rank with my method and Fisher's method to check out my method is more sensitive to biomarkers responding only to specific stress with GOterm (Gene Ontology term). Finally, I compared this stress prediction model with other machine learning classification methods, Random Forest and SVM to compare model accuracy.

3.1 Analysis of high stress-responsive genes

I prepared GOterm database for Arabidopsis thaliana which is compatible with my dataset. Also, as the 'correlation weight' value is not p-value, I picked top 500 genes for each stress instead of using p-value cutting and processed GOterm analysis with Fisher's exact test.

Experiment result shows lots of GOterm which is highly related to specific stress in high rank. In Table 1, Gene Ontology like "GO:0009408+ response to heat", "GO:0009644+ response to high light intensity", "GO:0010286+ heat acclimation" are mapped in very high rank and p-value of these terms also shows high relation to heat

stress. Also, there were drought and salt stress related GOterm with very high p-value, which I suspect as sub-effect of heat stress. Also, “GO:0009414+ response to water deprivation”, “GO:0009651+ response to salt stress”, “GO:0006970+ response to osmotic stress”, etc. GOterms were found in salt stress samples, which are highly related to salt stress. Highly related GOterms were also found in drought samples, such as “GO:0009414+ response to water deprivation”, “GO:0080167+ response to karrikin”. Cold stress samples result GOterms which is highly correlated, “GO:0009409+ response to cold”, “GO:0009631+ cold acclimation”, “GO:0019761+ glucosinolate biosynthetic process”. Glucosinolate biosynthetic process is well known by decreasing expression profiles of BrMYB[12].

Stress	GOTerm	p-val	Rank of stress
Heat	GO:0009408+response to heat	0	1
Heat	GO:0009644+response to high light intensity	0	2
Heat	GO:0010286+heat acclimation	0	5
Heat	GO:0034605+cellular response to heat	0.0002	9
Heat	GO:0009651+response to salt stress	0.0012	12
Heat	GO:0009414+response to water deprivation	0.0016	14
Salt	GO:0009414+response to water deprivation	0.0039	1
Salt	GO:0009737+response to abscisic acid	0.0039	2
Salt	GO:0009651+response to salt stress	0	3
Salt	GO:0006970+response to osmotic stress	0	4
Salt	GO:0006979+response to oxidative stress	0	11
Salt	GO:0009415+response to water	0	20
Salt	GO:0006749+glutathione metabolic process	0	24
Drought	GO:0009414+response to water deprivation	0	1
Drought	GO:0009651+response to salt stress	0	5
Drought	GO:0009611+response to wounding	0	9
Drought	GO:0080167+response to karrikin	0	12
Drought	GO:0006970+response to osmotic stress	0.0002	20
Cold	GO:0009409+response to cold	0	1
Cold	GO:0009631+cold acclimation	0	2
Cold	GO:0019761+glucosinolate biosynthetic process	0	6

Table 1. Gene Ontology analysis result for Arabidopsis time-series dataset with top 500 genes for each stress. Most stress are successfully found with very high p-value.

3.2 Gene rank comparison with Fisher's method

To find out my method identifies genes which is only responsive to specific stress, I compared gene rank which contains at least two GO terms between traditional fisher method to integrate p-value and my model. To prevent useless genes are included in my comparison, I excluded genes which are ranked under 2000, which is too low rank to be considered as important biomarker. Also, I checked gene rank twice for same gene with different stress type as such genes are correlated to multiple stress.

My model respectively showed more lower rank for those genes (45 genes are downranked out of 60 genes compared with comparative method) in Table 2, showing my group effect functions significantly. Further investigation with my database visualization tool for these time series samples were done, and it was absolute that those samples only react to specific stress type, which indicates my method finds genes only responding to specific stress type with higher level compared to other method (Table 3).

Genename	GOTerm	Rank of our model	Rank of fisher method
AT2G47180	heat,cold	heat(243), cold(500)	heat(39), cold(164)
AT5G37770	heat,cold	heat(2007), cold(3414)	heat(1878), cold(2510)
AT5G57560	heat,cold	heat(1357), cold(1428)	heat(235), cold(627)
AT5G58070	heat,cold	heat(693), cold(111)	heat(258), cold(167)
AT5G59820	heat,cold	heat(1069), cold(512)	heat(234), cold(128)
AT2G47180	heat,salt	heat(243), salt(842)	heat(39), salt(722)
AT3G09350	heat,salt	heat(61), salt(1341)	heat(35), salt(1712)
AT1G01060	cold,salt	salt(1762), cold(1342)	salt(1578), cold(298)
AT2G17840	cold,salt	salt(120), cold(247)	salt(279), cold(34)
AT2G19450	cold,salt	salt(1201), cold(86)	salt(700), cold(162)
AT2G38470	cold,salt	salt(234), cold(4958)	salt(142), cold(3504)
AT2G42540	cold,salt	salt(257), cold(79)	salt(538), cold(23)
AT2G46830	cold,salt	salt(506), cold(267)	salt(338), cold(31)
AT2G47180	cold,salt	salt(842), cold(500)	salt(722), cold(1642)
AT3G23830	cold,salt	salt(2516), cold(3530)	salt(1590), cold(2493)
AT3G48360	cold,salt	salt(1007), cold(1968)	salt(111), cold(447)
AT5G23860	cold,salt	salt(1280), cold(320)	salt(2527), cold(449)
AT5G52300	cold,salt	salt(43), cold(2982)	salt(38), cold(1327)
AT5G52310	cold,salt	salt(10), cold(333)	salt(6), cold(4)
AT5G58670	cold,salt	salt(291), cold(2148)	salt(634), cold(1284)
AT4G02380	cold,drought	drought(1013), cold(416)	drought(136), cold(278)

Table 2. Rank difference between my biomarker prediction model(NB) and Fisher's method. Genes with bold font are downranked in my model.

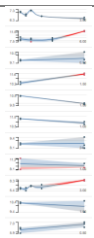
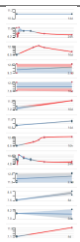
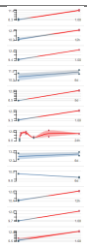
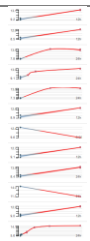
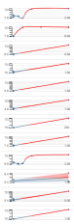
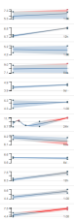
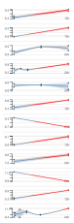
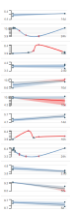
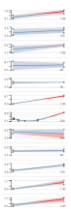
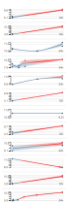
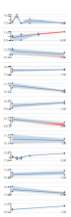
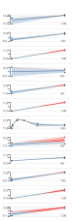
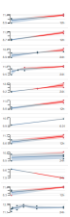
GENE	HEAT	DROUGHT	SALT	COLD
AT5G52310 (SALT, COLD)				
AT2G47180 (HEAT, SALT)				
AT2G46830 (SALT, COLD)				
AT4G02380 (DROUGHT, COLD)				

Table 3. Time-series gene expression visualization with some time series data and genes which are downranked in my model from Table 2. All genes are reacting actively to other stress that indicates these genes are responding to other stimulus, giving less novelty as biomarker.

3.3 Stress type prediction

I further analyzed which feature embedding method gives more suitable feature for current model (limma or Foldchange) and whether my prediction model showing valid prediction compared with traditional machine learning techniques, e.g. SVM, Random Tree, Random Forest.

In Table 4, feature embedding with foldchange shows much better accuracy compared to limma, so Foldchange feature embedding is better in small timepoints like my dataset. Also, my model correctly estimated the stress for 48 data, which shows high predict accuracy compared to other methods. In further investigating with these three samples which my model predicted wrong, those two time series data of which are predicted wrong stress type are actually reacted to both stresses (Figure 5), meaning my prediction is not completely wrong.

Methods	Accuracy
LRL+FC	0.963
RF+FC	0.961
SMO+FC	0.945
LRL+limma	0.821
RF+limma	0.853
SMO+limma	0.813

Table 4. Prediction accuracy with feature embedding types (FC; foldchange, limma p-value) and models (NB: my model). Feature embedding using Foldchange extracts feature more better than limma method in small timepoints. Also, my model accuracy shows high accuracy compared to other models (RF, SMO).

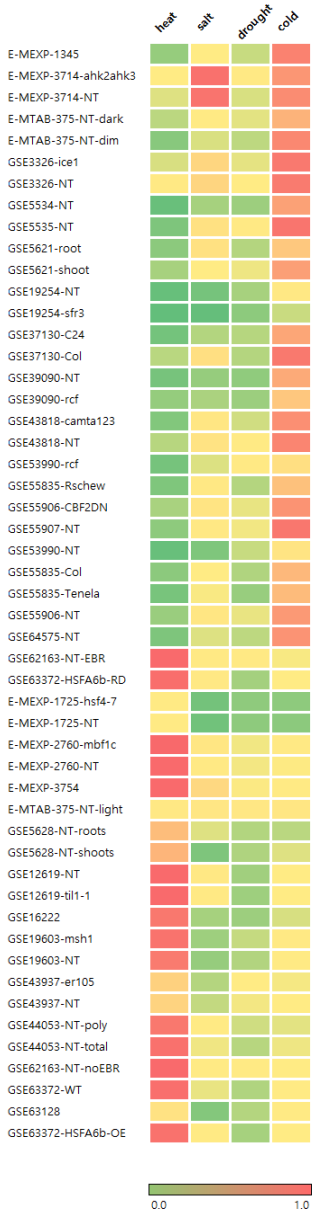


Figure 5. Stress type prediction result of my model. Cold stress samples until GSE64575-NT, and others are heat stress samples. Two time series data, E-MEXP-3714-ahk2ahk3 and E-MEXP-3714-NT, are mispredicted with salt stress, but it's still showing high probability with cold stress, which is true label.

Chapter 4

Discussions

In this study, I present a method to identify effectively responsive genes to specific stress and identifying stress type using time series data with Arabidopsis time series dataset. Since time series gene expression data isn't well organized problem in computer science, I defined time series data and made dataset from Arabidopsis gene expression data.

I embedded feature with foldchange rather than limma which is well-known method for marking DEGs using statistical methods as foldchange works better. I suspect the reason is there are very few timepoints (most of them are 2 points), in which statistical method is almost useless. Also, I processed feature embedding separating up-down signal to prevent signal cancelling.

Using this embedded feature data, I made special model to calculate relativity between stress and genes using logical regression layer with sigmoid cross-entropy loss and group effects to identify gene only responsive to specific stress. Furthermore, I made model predicting stress from time series data using transposed logical regression layer and CMCL loss function.

This study showed I identified differentially expressing gene which reacts to specific stress by showing complex GOterm registered genes are ranked down in my method. It suggests highly ranked genes in my

method may play important role to specific stress, or may be important biomarker for checking out specific stress. Also, my method predicts stress with high probability, and even better than other machine learning methods with same embedding features.

Furthermore, this model stress type prediction result suggests how much sample is affected by stress from absolute value of predicted probability. For example, if reacted genes in a time series data are only reacted to feature which learned as weakly responsive to stress, then output probability to stress would be low even if it's stress type is correctly predicted. But still I need more investigation to see my assumption is correct.

This method can be extended to solving high-dimensional problems with few samples. Also, I could think up better method to extract more effective feature as current method, foldchange, losses a lot of information through dimension reduction.

References

[1] Edgar R., D.M., E., L.A.: Gene expression omnibus: Ncbi gene expression and hybridization array data repository. Nucleic acids research 30(1), 207-220 (2002)

[2] Kolesnikov N., e.a. Hastings E.: Arrayexpress update-simplifying data submissions. Nucleic acids research 43(1), 1113-1116 (2015)

[3] Lan, H., Carson, R., Provart, N.J., Bonner, A.J.: Combining classifiers to predict gene function in arabidopsis thaliana using large-scale gene expression measurements. BMC Bioinformatics 8(1), 358 (2007). doi:10.1186/1471-2105-8-358

[4] Ko, D., Xu, W., Windle, B.: Gene function classification using nci-60 cell line gene expression profiles. Computational Biology and Chemistry 29(6), 412-419 (2005). doi:10.1016/j.compbiolchem.2005.09.003

[5] Tong, D.-L.: Hybridising genetic algorithm-neural network (gann) in marker genes detection 2, 1082-1087 (2009). doi:10.1109/ICMLC.2009.5212372

[6] Ko, D., Windle, B.: Enriching for correct prediction of biological processes using a combination of diverse classifiers. BMC Bioinformatics 12(1), 189 (2011). doi:10.1186/1471-2105-12-189

[7] Wu, M.Y., Dai, D.Q., Shi, Y., Yan, H., Zhang, X.F.: Biomarker

identification and cancer classification based on microarray data using laplace naive bayes model with mean shrinkage. IEEE/ACM Transactions on Computational Biology and Bioinformatics 9(6), 1649-1662 (2012). doi:10.1109/TCBB.2012.105

[8] Ernst, J., Bar-Joseph, Z.: Stem: a tool for the analysis of short time series gene expression data. BMC Bioinformatics 7(1), 191 (2006). doi:10.1186/1471-2105-7-191

[9] Ritchie ME., W.D.H.Y.L.C.S.W. Phipson B., GK, S.: limma powers differential expression analyses for rna-sequencing and microarray studies. Nucleic Acids Research 43(7), 47 (2015)

[10] Lee K., P.K.S.J. Hwang C.: Confident multiple choice learning. arXiv 1706(03475) (2017)

[11] Ashburner et al. : Gene ontology: tool for the unification of biology. Nat Genet 25(1), 25-9 (2000)

[12] Seo, Mi-Suk et al. : Expression Profiles of BrMYB Transcription Factors Related to Glucosinolate Biosynthesis and Stress Response in Eight Subspecies of Brassica Rapa. FEBS Open Bio 7.11: 1646-1659. (2017)

초 록

이질적 시계열 유전자 데이터의 스트레스 연관 유전자 및 스트레스 예측 기법

강동원

컴퓨터공학부

서울대학교 대학원

큰 차원의 데이터를 포함하는 유전자 발현 데이터가 형성됨에 따라, 시계열 유전자 발현 데이터의 통합 분석의 필요성이 대두되고 있다. 그러나 유전자 발현 자료를 분석하는 것은 기존의 컴퓨터 과학에서 다루지 못했던 새로운 시계열 분석 문제이다. 이는 시계열 유전자 데이터가 수만여개의 유전자 및 복합적인 변수를 가지고 있는 높은 차원의 데이터를 가지고 있는 문제이지만, 이에 비해 샘플 수가 적고 측정된 시점의 개수 또한 적은 문제를 가지고 있기 때문이다.

이 연구에서는 이러한 이질적 시계열 데이터에 적합한 형태의 데이터를 가공하고, 이러한 가공한 데이터에 대한 논리적 연관성 레이어를 Cross-entropy 및 그룹 효과를 사용하여 학습시켜 특정 스트레스 유형에만 반응하는 유전자를 조사하였다. 또한, 해당 논리

연관성 레이어와 CMCL(Confidence Multiple Choice Learning) 학습 함수를 이용하여 시계열 유전자 발현 데이터에 대해 해당 개체가 받았을 것으로 추정되는 스트레스 유형을 예측하는 모델을 만들었다.

본 모델이 스트레스와 연관성 있는 유전자를 찾아내는지 확인하기 위하여 GOterm analysis를 수행하였고, 그 결과 스트레스와 연관있는 Gene ontology들이 확인되었다. 그리고 단일 스트레스에 대해 반응하는 유전자가 유의미하게 높게 확인되는지 확인하기 위하여 fisher's method을 사용하여 확인된 유전자 순위가 본 메서드에 비해서 어떻게 나타나는지 확인하였다. 그 결과 여러 GO term을 가지는 유전자, 즉 다수의 스트레스에 반응하는 유전자들의 순위가 낮아졌음을 보여 주었고, 이는 본 메서드가 단일 스트레스에 반응하는 유전자를 변별력 있게 찾아냄을 보여준다. 또한, 본 모델이 스트레스를 얼마나 정확히 예측하는지를 비교하기 위해 기존의 머신 러닝 기법인 랜덤 트리(Random Forest), SVM과의 비교를 하였고 에 비해 우리의 예측 모델이 더 나은 성능을 보였다. 또한 단순한 foldchange 방법은 적은 양의 시점을 가진 시계열 데이터에서는 limma p-value에 비해 모델 입력에 대한 유전자 특징을 선택하는 데 더 나은 기준임을 확인할 수 있었다.

이러한 결과들로 보았을 때, 본 연구는 적은 양의 시점을 가지는 시계열 데이터를 이용하여 특정 스트레스에 반응하는 유전자를 선택하고 스트레스를 예측하는 새로운 방법을 제시한다.

Keywords : Arabidopsis, Microarray, Time series, Machine Learning

Student Number : 2016-27463

감사의 글

먼저 해당 연구를 처음 진행할 데이터를 제공해주신 홍렬선배께 감사하다는 말씀을 전해드리고 싶으며, 그 이외 머신러닝 모델 작성 및 논문 작성에 있어서 기술적인 도움 및 언질을 주신 김선 교수님 및 여러 연구실 선배 분들에게 감사의 말씀을 표합니다. 뿐만 아니라 학위논문을 무사히 작성하기까지 응원해준 부모님과 여러 친구들에게도 감사하다는 말씀을 드리고 싶습니다.