International Journal of Hybrid Information Technology
Vol. 10, No.2 (2017), pp. 69-78
http://dx.doi.org/10.14257/ijhit.2017.10.2.07

Infer Gene Regulatory Network Based on the Novel Classifiers Fusion

Wei Zhang, Bin Yang* and Jiaguo Lv

School of Information Science and Engineering, Zaozhuang University, Zaozhuang,China 277160
**Corresponding author: batsi@126.com*

Abstract

Reconstruction of gene regulatory network (GRN) from gene expression data is still a big challenge. Inference of gene regulatory network is considered as a binary classification problem. In this paper, we develop a new supervised learning approach based on several classifiers fusion (SLCF) for inference of gene regulatory network. According to the characteristics of classified data, SLCF uses three classification methods: direct classification, minimal distance selection and flexible neural tree, respectively. The data from E.coli network is used to test our method and results reveal that SLCF performs better than classical unsupervised and supervised learning methods.

Keywords*: Gene regulatory network, flexible neural network, minimal distance, firefly algorithm*

1. Introduction

Transcriptional regulation is a basis of many crucial molecular processes such as oscillator, differentiation and homeostasis, and the correct inference of gene regulatory networks (GRN) is a helpful and essential task to understand the intricacies of the complex biological regulations and gain insights into biological processes of interest in systems biology for researchers [1-3]. With the availability of large dimensional microarray data and lots of true regulation relationships of biology processes which have been verified by biology experiments, relationships among thousands of genes could be inferred simultaneously [4-8].

However gene regulatory network is a complex and nonlinear dynamics system, inference of gene regulatory network is still a big challenge. Many statistics, data mining and computational intelligence methods have been proposed to identify regulations among genes. The methods are usually divided into two parts. The first part is unsupervised learning method, which contain Boolean network [9], Bayesian network [10], Petri network, differential Equation [11] and Information theory model [12]. The second one is supervised learning method, in which gene expression data and a list of known regulation relationships are required. Inference of gene regulatory network is considered as a binary classification problem [13-14]. For each target gene, the regulatory factors which regulate target gene are set as positive samples, while the regulatory factors which could not regulate target gene are set as negative samples. The tradition classification methods have been successfully applied to the inference of gene regulatory network, especially Support Vector Machines (SVM). SIRENE [15] based on SVM was proposed for reconstruction of gene regulatory networks from a compendium of gene expression data of *Escherichia coli* genes and a set of known regulations. Gillani *et al* developed a tool (CompareSVM) based on SVM to compare different kernel methods for inference of GRN [16].

ISSN: 1738-9968 IJHIT

In the supervised learning algorithm, a set of known regulations which are verified by biology experiments from public datasets such as RegulonDB, TRRD and KEGG, are required as the training data. But the gene regulatory network is a sparse network, and only a tiny fraction of the candidate regulators are expected to be true regulators for target gene, so this will cause that the number of positive samples is less than the number of negative samples. In some cases, the number of positive samples is zero. This unbalanced problem of training set could affect the performance of traditional classifiers and raise the time complexity.

In this paper, a new supervised learning method based on several classifiers fusion (SLCF) is proposed to infer gene regulatory network. In SLCF, firstly the train set is analyzed with the known regulation relationships. Secondly according to the number of positive and negative samples, SLCF divides three cases to identify regulations among genes. If the number of positive samples is zero, the test sample is directly classified as negative class. If the ratio of positive samples is relatively small, the simple classification method based on minimal distance selection is used. In other cases, the flexible neural tree is proposed to resolve the binary classification problem. The gene express data from E.*coli* network is used to test the performance of SLCF.

The paper is organized as follows: Section 2 gives the materials and methods. Section 3 presents some experiments for construction of gene regulatory networks. Some concluding remarks are presented in Section 4.

2. The Materials and Methods

2.1. Flexible Neural Tree

Flexible neural tree (FNT) model was proposed by Chen, which has been widely applied to solve classification problems such as intrusion detection, breast caner identification [17].

2.1.1. Flexible Neural Instructor

The used function set F and terminal instruction set T for creating a FNT model are described as follows:

$$S = F \bigcup T = \{+_2, +_3, \ldots +_N\} \bigcup \{x_1, x_2, \ldots, x_n\}. \quad (1)$$

Where $+_i$ denotes non-leaf node's instruction taking *i* arguments and x_i is leaf node's instruction taking no arguments. The output of a non-leaf node $+_i$ is calculated as a flexible neural operator (Figure 1), which could be calculated as follows.

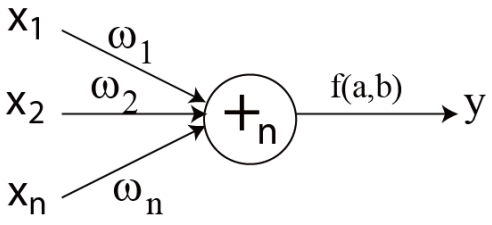

Figure 1. A Flexible Neuron Operator

$$net_i = \sum_{j=1}^{i} w_j x_j,$$
$$out_i = f(a_i, b_i, net_i) = e^{-(\frac{net_i - a_i}{b_i})^2}. \quad (2)$$

Where w_j is weight, x_j is the input to node $+_i$, $f(\cdot)$ is the flexible activation function, two adjustable parameters a_i and b_i are randomly created as flexible activation function parameters.

In the FNT, every node is selected randomly from the predefined instruction/operator sets *S*. If a leaf node is selected, this branch is terminated. If a non-leaf node $+_i$ is selected, *i* children are created in the next layer (do not exceed pre-defined maximum depth of FNT). A typical flexible neural tree model is shown as Figure 2.

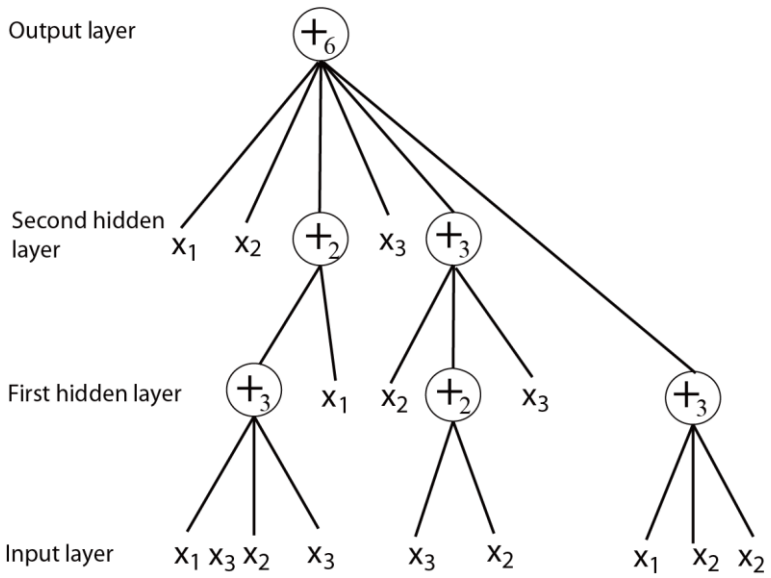

Figure 2. An Example of Flexible Neural Tree Model

2.1.2. Structure Optimization

Finding an optimal or near-optimal neural tree is formulated as an evolutionary search process. In this paper, we use three neural tree variation operators: mutation, crossover and selection. The detailed process of three operators is introduced in Ref [18].

2.1.3. Parameters Optimization

Firefly algorithm (FA) is an efficient optimization algorithm which was proposed by Xin-She Yang in 2009 [19]. It is very simple, has few parameters and easy to apply and implement, so this paper uses firefly algorithm to optimize the parameters of FNT model.

Firefly algorithm is the random optimization method of simulating luminescence behavior of firefly in the nature. The firefly could search the partners and move to the position of better firefly according to brightness property. A firefly represents a potential solution. In order to solve optimization problem, initialize a firefly vector $[x_1, x_2, \ldots, x_n]$ (n is the number of fireflies). As attractiveness is directly

proportional to the brightness property of the fireflies, so always the less bright firefly will be attracted by the brightest firefly.

The brightness of firefly *i* is computed as

$$B_i = B_{i0} * e^{-\gamma r_{ij}} \tag{3}$$

Where B_{i0} represents maximum brightness of firefly *i* by the fitness function as $B_{i0} = f(x_i)$. γ is coefficient of light absorption, and r_{ij} is the distance factor between the two corresponding fireflies *i* and *j*.

The movement of the less bright firefly toward the brighter firefly is computed by

$$x_i(t+1) = x_i(t) + \beta_i(x_j(t) - x_i(t)) + \alpha\varepsilon_i \tag{4}$$

Where α is step size randomly created in the range [0, 1], and ε_i is Gaussian distribution random number.

2.2. Minimal Distance Selection

In the classifier problem, the unbalance data problem could lead to low accuracy of tradition classifiers. Thus we use minimal distance to measure the similarity between train and test datasets. Suppose that the train data $(x_1\ c_1; x_2\ c_2; \ldots; x_m\ c_j)$ which are divided into *j* classes. Test data is $(y_1, y_2, \ldots y_n)$. For each test data y_i ($i = 1, 2, \ldots, n$), compute $\mathrm{Min}\{F(x_1, y_i), F(x_2, y_i), \ldots, F(x_m, y_i)\}$. If $F(x_k, y_i)$ is the minimal, the test data y_i is classified as the same as the train data x_k.

Where $F(\cdot)$ is the distance function and here we choose Euclidean distance and Pearson correlation coefficient.

Suppose that X and Y are two *n*-dimension vectors. Euclidean distance is the easiest to understand and calculate.

$$ED(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{5}$$

Pearson correlation coefficient is used to reflect the linear correlation degree of two variables.

$$PCC(X,Y) = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} \tag{6}$$

Where $Cov(X,Y)$ is the covariance of X and Y, σ_X is the standard deviation of X.

The final distance value is computed as follows.

$$MD(X,Y) = \frac{ED(X,Y)}{1 + |PCC(X,Y)|} \tag{7}$$

2.3. Ensemble Method

Suppose that the number of the positive samples is $N_{positive}$, the number of the negative samples is $N_{negative}$, and the number of overall samples is N_{all}. The flowchart of SLCF is depicted in Figure 3.

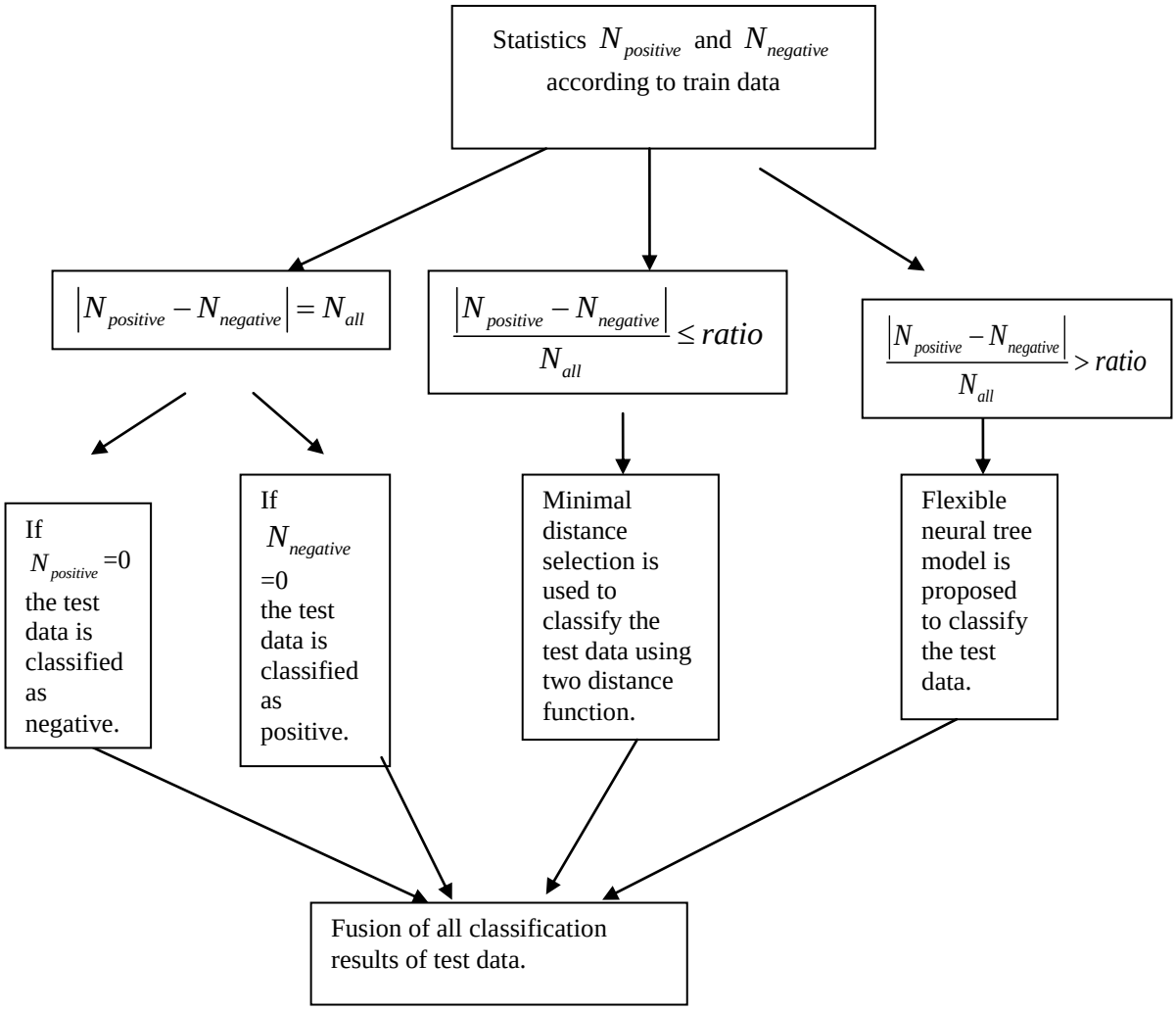

Figure 3. The Flowchart of SLCF

3. Experiments

In this part, the expression data generated from sub network from *E.coli* network using three different experimental conditions (knockout, knockdown and multifactorial) are used to test our method [16]. This network contains 150 genes and 202 true regulations. To evaluate the performance of our method, we compare it with CLR (context likelihood to relatedness) [20], SVM [15] and FNT. The parameters in these methods are set by default.

Five criterions (sensitivity or true positive rate (TPR), false positive rate (FPR), positive predictive (PPV), accuracy (ACC) and F-score) are used to test the performance of the method. Firstly, we define four variables, *i.e.*, TP, FP, TN and FN are the number of true positives, false positives, true negatives and false negatives, respectively. Five criterions are defined as followed.

$$\begin{aligned} TPR &= TP/(TP+FN), \\ FPR &= FP/(FP+TN), \\ PPV &= TP/(TP+FP), \\ ACC &= (TP+TN)/(TP+FP+TN+FN), \\ F-score &= 2PPV*TPR/(PPV+TPR) \end{aligned} \tag{8}$$

Through several runs, the results are listed in Table 1. From the results, we can see that supervised learning methods (SVM, FNT and SLCF) perform better than unsupervised learning methods (CLR and GENIE) except that CLR has the highest sensitivity (TPR) with multifactorial data. SLCF has the highest F-score, which means that the inferred network achieves the optimal balance in terms of sensitivity and positive predictive rate (more true regulations and less false positive regulations).

In addition, to assess the effectiveness of our proposed method, the ROC curves obtained by SVM, FNT and SLCF on E.coli network with different experimental conditions are shown in Figure 3, Figure 4 and Figure 5 respectively. The results show that SLCF performs better than other popular supervised learning methods (SVM and FNT).

Table 1. Comparison of Two Methods on E.Coli Network with Different Experimental Conditions

		TPR	FPR	PPV	ACC	F-score
Knockout data	CLR	0.4356	0.3478	0.0114	0.6444	0.0222
	GENIE	0.4010	0.2515	0.0145	0.7387	0.0279
	SVM	0.4554	**0.0076**	0.3552	0.9786	0.3991
	FNT	0.5099	0.0085	0.3552	0.9783	0.4187
	SLCF	**0.5594**	0.0092	**0.3587**	**0.9783**	**0.4371**
Knockdown data	CLR	0.4406	0.3602	0.0111	0.6323	0.0217
	GENIE	0.4009	0.2925	0.0125	0.6984	0.0242
	SVM	0.5198	**0.0073**	0.3962	0.9796	0.4497
	FNT	0.5792	0.0089	0.3738	0.9784	0.4544
	SLCF	**0.6881**	0.0089	**0.4162**	**0.9795**	**0.5187**
Multifactorial data	CLR	**0.8168**	0.3355	0.0219	0.6600	0.0427
	GENIE	0.3366	0.2931	0.1046	0.6973	0.0203
	SVM	0.5445	**0.0076**	0.3971	0.9795	0.4593
	FNT	0.6139	0.0079	0.4175	0.9798	0.4970
	SLCF	0.6931	0.0087	**0.4242**	**0.9798**	**0.5263**

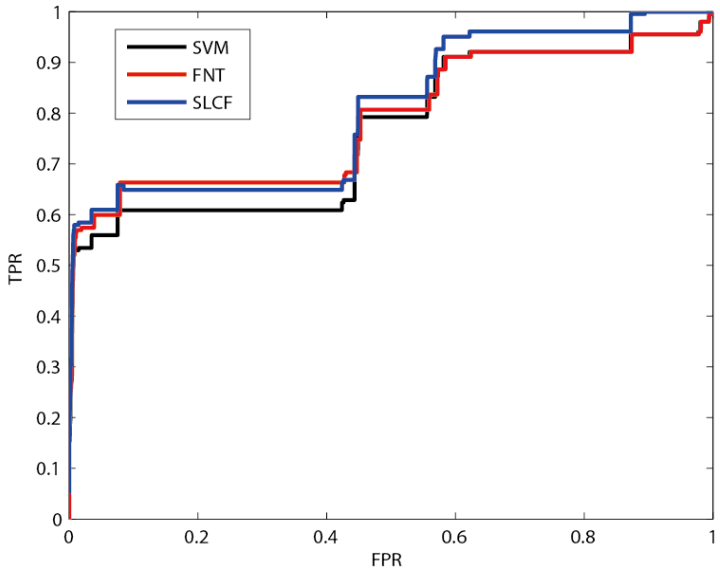

Figure 4. ROC Curves of Three Methods with Knockdown Data

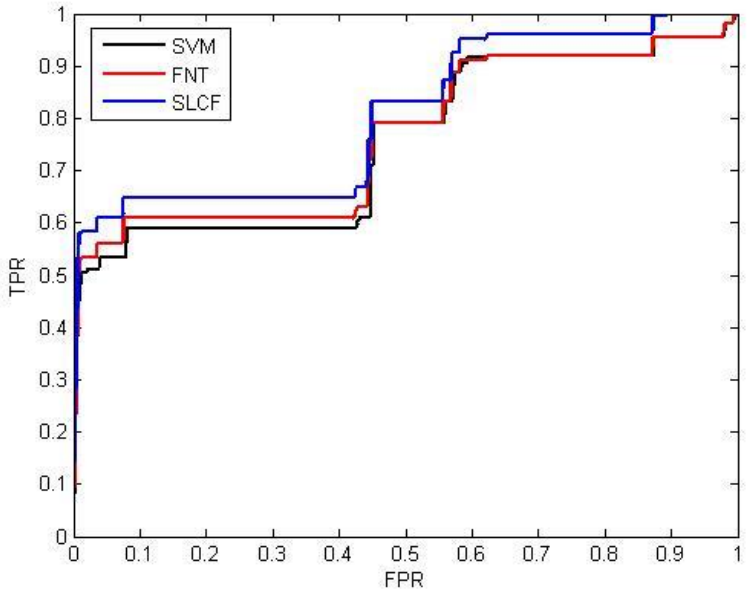

Figure 5. ROC Curves of Three Methods with Knockouts Data

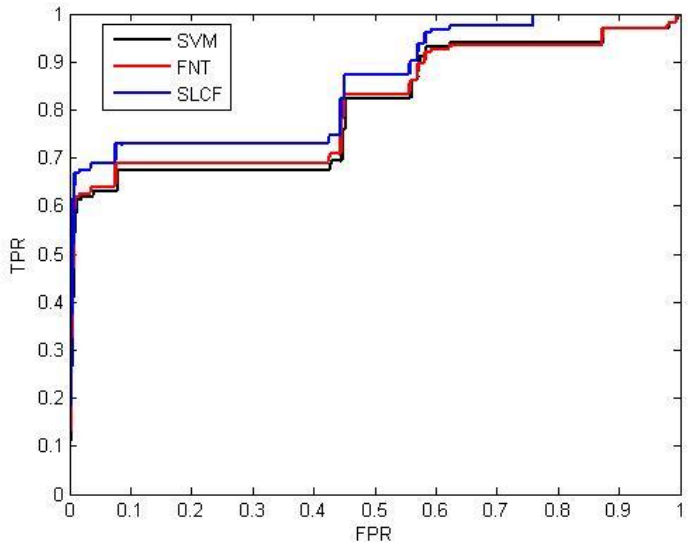

Figure 6. ROC Curves of Three Methods with Multifactorial Data

4. Conclusions

To summarize, a new supervised approach based on the fusion of direct classification, minimal distance method and flexible neural tree is proposed for inference of gene regulatory network. The sub network with 150 genes from E.coli network is used to validate our method. TPR, FPR, PPV, ACC, F-score and ROC curves reveal that our method could gain higher accuracy for biological datasets (knockout, knockdown and multifactorial) than CLR, GENIE, SVM and FNT.

In the future, we will apply SLCF to more large-scale real gene regulatory network identification and develop the parallel program in order to improve the runtime.

Acknowledgements

This work was supported by the PhD research startup foundation of Zaozhuang University (No.2014BS13), Zaozhuang University Foundation (No. 2015YY02), and Shandong Provincial Natural Science Foundation, China (No. ZR2015PF007).

References

[1] J. Wu, X. Zhao, Z. Lin and Z. Shao, "Large scale gene regulatory network inference with a multi-level strategy", Mol Biosyst., vol. 12, no. 2, **(2016)**, pp. 588-97.
[2] S. Mandal, A. Khan, G. Saha and R.K. Pal, "Reverse engineering of gene regulatory networks based on S-systems and Bat algorithm", J Bioinform Comput Biol., vol. 4, **(2016)**, pp. 1650010.
[3] N. Omranian, J.M. Eloundou-Mbebi, B. Mueller-Roeber and Z. Nikoloski, "Gene regulatory network inference using fused LASSO on multiple data sets", Sci Rep, vol. 6, **(2016)**, pp. 20533.
[4] D.C. Ellwanger, J.F. Leonhardt and H.W. Mewes, "Large-scale modeling of condition-specific gene regulatory networks by information integration and inference", Nucleic Acids Res, vol. 42, no. 21, **(2014)**.
[5] P. Vera-Licona, A. Jarrah, L.D. Garcia-Puente, J. McGee and R. Laubenbacher, "An algebra-based method for inferring gene regulatory networks", BMC Syst Biol., vol. 8, **(2014)**, pp. 37.
[6] Y. Xie, R. Wang and J. Zhu, "Construction of breast cancer gene regulatory networks and drug target optimization", Arch Gynecol Obstet, vol. 290, no. 4, **(2014)** pp. 749-55.
[7] C.A. Penfold, J.B. Millar and D.L. Wild, "Inferring orthologous gene regulatory networks using interspecies data fusion", Bioinformatics, vol. 31, no. 12, **(2015)**, pp. i97-105.

[8] B. Baur and S. Bozdag, "A canonical correlation analysis-based dynamic bayesian network prior to infer gene regulatory networks from multiple types of biological data", J Comput Biol., vol. 22, no. 4, **(2015)**, pp. 289-99.
[9] M. Yang, R. Li and T. Chu, "Construction of a Boolean model of gene and protein regulatory network with memory", Neural Netw, vol. 52, **(2014)**, pp. 18-24.
[10] E.S. Adabor, G.K. Acquaah-Mensah and F.T. Oduro, "SAGA: a hybrid search algorithm for Bayesian Network structure learning of transcriptional regulatory networks", J Biomed Inform, vol. 53, **(2015)**, pp. 27-35.
[11] M. Sun, X. Cheng, J.E. Socolar, "Causal structure of oscillations in gene regulatory networks: Boolean analysis of ordinary differential equation attractors", Chaos, vol. 23, no. 2, **(2013)**, pp. 025104.
[12] J. Wang, B. Chen, Y. Wang, N. Wang, M. Garbey, R. Tran-Son-Tay, S.A. Berceli, R. Wu, "Reconstructing regulatory networks from the dynamic plasticity of gene expression by mutual information", Nucleic Acids Res, vol. 41, no. 8, **(2013)**, pp. e97.
[13] S.R. Maetschke, P.B. Madhamshettiwar, M.J. Davis and M.A. Ragan, "Supervised, semi-supervised and unsupervised inference of gene regulatory networks", Brief Bioinform, vol. 15, no. 2, **(2014)**, pp. 195-211.
[14] L. Cerulo, C. Elkan and M. Ceccarelli, "Learning gene regulatory networks from only positive and unlabeled data", BMC Bioinformatics, vol. 11, **(2010)**, pp. 228.
[15] F. Mordelet and J.P. Vert, "SIRENE: supervised inference of regulatory networks", Bioinformatics, vol. 24, no. 16, **(2008)**, pp. i76-82.
[16] Z. Gillani, M.S. Akash, M.D. Rahaman and M. Chen, "CompareSVM: supervised, Support Vector Machine (SVM) inference of gene regularity networks", BMC Bioinformatics, vol. 15, **(2014)**, pp. 395.
[17] Y.H. Chen, B. Yang, J. Dong and A. Abraham, "Time series forecasting using flexible neural tree model", Inf. Sci., vol. 174, no. 3/4, **(2005)**, pp. 219–235.
[18] Y.H. Chen, B. Yang and Q.F. Meng, "Small-time scale network traffic prediction based on flexible neural tree", Appl. Soft Comput., vol.12, **(2012)**, pp. 274–279.
[19] X.S. Yang, "Firefly algorithms for multimodal optimization. Stochastic Algorithms: Foundations and Applications", Lecture Notes in Computer Sciences, vol. 5792, **(2009)**, pp. 169-178.
[20] A.J. Butte, P. Tamayo, D. Slonim, T.R. Golub and I.S. Kohane, "Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks", Proc Natl Acad Sci USA, vol. 97, no. 22, **(2000)**, pp. 12182-12186.

Authors

Bin Yang, he is the teacher of Zaozhuang University. He has pursued his Ph.D. in School of Information Science and Engineering from Shandong University, Jinan, China. He received his B.Sc. and Master degree in School of Information Science and Engineering from University of Jinan. His research interests include hybrid computational intelligence and its applications in time-series prediction, system identification and gene regulatory network.

Wei Zhang, he is the dean of School of information science and engineering in Zaozhuang University. In 1996 he received the master degree in Qufu Normal University. His research interests include data mining, network traffic prediction and network security.