Available online at www.sciencedirect.com
ScienceDirect
Computational Biology and Chemistry 32 (2008) 29–38
Computational Biology and Chemistry
www.elsevier.com/locate/compbiolchem

Improved binary PSO for feature selection using gene expression data

Li-Yeh Chuang [a], Hsueh-Wei Chang [b], Chung-Jui Tu [c], Cheng-Hong Yang [c,*]

[a] *Department of Chemical Engineering, I-Shou University, Kaohsiung 840, Taiwan*
[b] *Department of Biomedical Science and Environmental Biology, and Graduate Institute of Natural Products, College of Pharmacy, Kaohsiung Medical University, Kaohsiung, 807, Taiwan*
[c] *Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung 807, Taiwan*

Received 24 December 2006; accepted 10 September 2007

Abstract

Gene expression profiles, which represent the state of a cell at a molecular level, have great potential as a medical diagnosis tool. Compared to the number of genes involved, available training data sets generally have a fairly small sample size in cancer type classification. These training data limitations constitute a challenge to certain classification methodologies. A reliable selection method for genes relevant for sample classification is needed in order to speed up the processing rate, decrease the predictive error rate, and to avoid incomprehensibility due to the large number of genes investigated. Improved binary particle swarm optimization (IBPSO) is used in this study to implement feature selection, and the *K*-nearest neighbor (*K*-NN) method serves as an evaluator of the IBPSO for gene expression data classification problems. Experimental results show that this method effectively simplifies feature selection and reduces the total number of features needed. The classification accuracy obtained by the proposed method has the highest classification accuracy in nine of the 11 gene expression data test problems, and is comparative to the classification accuracy of the two other test problems, as compared to the best results previously published.

Keywords: Improved binary particle swarm optimization; Feature selection; Gene expression data

1. Introduction

DNA microarray examples are generated by a hybridization of mRNA from sample tissues or blood and cDNA (in the case of a spotted array), as well as hybridization of oligonuclectides of DNA (in the case of Affymetrix chips, this is done on the surface of the chip-array). DNA microarray technology allows for the simultaneous monitoring and measurement of thousands of gene expression activation levels in a single experiment. Class memberships are characterized by the production of proteins, i.e. gene expressions refer to the production level of proteins specific for a gene. Thus, microarray data can provide valuable results for a variety of gene expression profile problems, and contribute to advances in clinical medicine. The application of microarray data on cancer type classification has recently gained in popularity. Coupled with statistical techniques, gene expression patterns have been used in screening for potential tumor markers. Differential expressions of genes are analyzed statistically and genes are assigned to various classes, which may (or may not) enhance the understanding of the underlying biological processes.

Microarray gene expression technology has opened the possibility of investigating the activity of thousands of genes simultaneously. Gene expression profiles show the measurement of relative abundance of mRNA corresponding to the genes. Thus, discriminant analysis of microarray data has great potential as a medical diagnosis tool. The goal of microarray data classification is to build an efficient and effective model that identifies the differentially expressed genes and may be used to predict class membership for any unknown samples. The challenges posed in microarray classification are the limited size of samples in comparison to the high dimensionality of the sample, along with experimental variations in measured gene expression levels.

The classification of gene expression data samples involves feature selection and classifier design. Generally, only a small number of genes show a strong correlation with a certain phenotype compared to the total number of genes investigated. Thus, in order to analyze gene expression profiles correctly, feature (gene) selection is most crucial for the classification process. The goal of feature selection is to identify the subset of differentially

* Corresponding author. Tel.: +886 7 370 6752.
E-mail addresses: chuang@isu.edu.tw (L.-Y. Chuang), changhw@kmu.edu.tw (H.-W. Chang), 1093320134@cc.kuas.edu.tw (C.-J. Tu), chyang@cc.kuas.edu.tw (C.-H. Yang).

1476-9271/$ – see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compbiolchem.2007.09.005

expressed genes that are potentially relevant for distinguishing the sample classes. A good method of selecting genes relevant for sample classification is needed in order to accelerate the processing rate, decrease the predictive error rate, and to avoid incomprehensibility due to spurious data correlations; it should be based on the number of genes investigated. Several methods have been used to perform feature selection on the training and testing data, e.g. genetic algorithms (Raymer et al., 2000; Yang and Honavar, 1998), branch and bound algorithms (Narendra and Fukunage, 1997; Yu and Yuan, 1993), sequential search algorithms (Pudil et al., 1994), mutual information (Roberto, 1994), tabu search (Zhang and Sun, 2002) entropy-based methods (Liu et al., 2005), regularized least squares (Ancona et al., 2005), random forests (Diaz-Uriarte and Alvarez de Andres, 2006), instance-based methods (Berrar et al., 2006), and least squares support vector machines (Tang et al., 2006).

In this paper, improved binary particle swarm optimization (IBPSO) is used to implement the feature selection process. A *K*-nearest neighbor (*K*-NN) serves as an evaluator of the IBPSO for gene expression data classification problems taken from the literature. The results reveal that the proposed classification method achieves superior predictive error rate when applied to 11 data sets from the literature, as compared to methods previously published. Furthermore, the number of genes selected can be significantly decreased.

2. Methods

2.1. Improved Binary Particle Swarm Optimization (IBPSO)

Particle swarm optimization (PSO) is a population-based stochastic optimization technique, which was developed by Kennedy and Eberhart (1995). PSO simulates the social behavior of organisms, such as birds in a flock and fish in a school. This behavior can be described as an automatically and iteratively updated system. In PSO, each single candidate solution can be considered a particle in the search space. Each particle makes use of its own memory and knowledge gained by the swarm as a whole to find the best solution. All of the particles have fitness values, which are evaluated by a fitness function to be optimized. During movement, each particle adjusts its position by changing its velocity according to its own experience and according to the experience of a neighboring particle, thus making use of the best position encountered by itself and its neighbor. Particles move through the problem space by following a current of optimum particles. The process is then iterated a fixed number of times or until a predetermined minimum error is achieved (Kennedy et al., 2001).

PSO was originally introduced as an optimization technique for real-number spaces. PSO has been successfully applied in many areas: function optimization, artificial neural network training, fuzzy system control, and other application problems. A comprehensive survey of the PSO algorithms and their applications can be found in Kennedy et al. (2001). However, many optimization problems occur in a space featuring discrete, qualitative distinctions between variables and between levels of variables. Kennedy and Eberhart introduced binary PSO (BPSO), which can be applied to discrete binary variables. In a binary space, a particle may move to near corners of a hypercube by flipping various numbers of bits; thus, the overall particle velocity may be described by the number of bits changed per iteration (Kennedy and Eberhart, 1997).

Gene expression data characteristically have a high dimension, so we expect superior classification results in different dimension areas. Each particle adjusts its position according to two fitness value, *pbest* and *gbest*, to avoid being trapped in a local optimum by fine-tuning the inertia weight. *pbest* is a local fitness value, whereas *gbest* constitutes a global fitness value. If the *gbest* value is itself trapped in a local optimum, a search of each particle limit in the same area will occur, thereby preventing superior results of classification. Thus, we propose a method that retires *gbest* under such circumstances and uses an improved binary particle swarm optimization (IBPSO). By resetting *gbest* we can avoid IBPSO getting trapped in a local optimum, and superior classification result can be achieved with a reduced number of selected genes.

Fig. 1a shows that almost all particles converged near *gbest* after a certain period. If the *gbest* value does not change after three iterations, it can be considered stuck at a local optimum. Under such circumstances, the current *gbest* fitness value (classification accuracy and selected features) is reset to zero, i.e. retired (Fig. 1b). This form of IBPSO skips the local optimum, and searches for superior classification results in an area with a lower number of genes. Fig. 1c shows that the individual particles will converge towards the reset *gbest* value and thus leave the local optimum by searching for the new *gbest* value in a region with a lower number of genes (Fig. 1d). This process achieves superior classification and effectively reduces the number of genes that need to be selected.

In this paper, an improved form of binary PSO (IBPSO) was used since the position of each individual particle can be given in binary form (0 or 1), which adequately reflects the straightforward "yes/no" choice of whether a feature needs to be selected

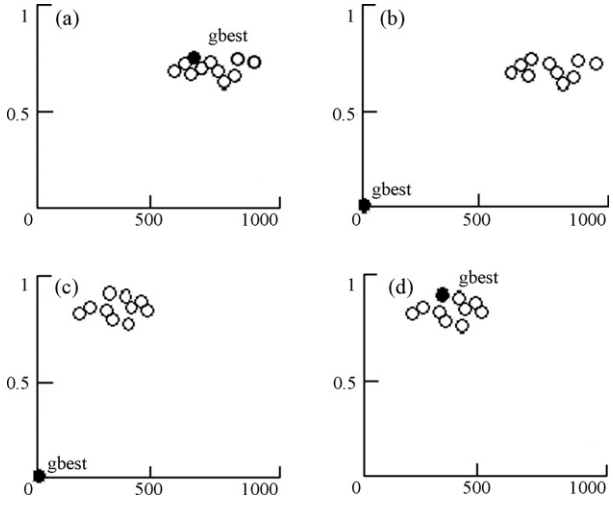

Fig. 1. (a) *gbest* is trapped in a local. (b) *gbest* is rest to zero. (c) Particle movement after resetting of *gbest*. (d) Particles congregated towards the updated *gbest* value, improving the individual position.