ISSN 1682-296X (Print)
ISSN 1682-2978 (Online)

Bio Technology

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Biotechnology 4 (3): 163-172, 2005
ISSN 1682-296X

New Feature-extraction Criteria and Classification Algorithms for Cancer Gene Expression Datasets

[1,2]Wang Shitong, [2]F.L. Chung, [1]Deng Zhaohong, [3]L.I.N. Qing and [4]H.U. Dewen
[1]School of Information Engineering, Southern Yangtze University, Wuxi, China
[2]Department of Computing, HongKong Polytechnic University, Hong Kong, China
[3]Department of Computer Science and Engineering,
Nanjing University of Science and Technology, Nanjing, China
[4]School of Automation, National Defense University of Science and Technology, Changsha, China

Abstract: In this study, based on DCCM (Differential Capability Control Machine), the new feature-extraction criterion NFEC is developed using the first-order differential information and a new feature-extraction algorithm DCCFE is accordingly proposed for binary classification problems. NFEC and DCCFE are then extended to their multi-classification versions, i.e., m_NFEC and m_DCCFE, respectively. Present experimental results demonstrate that the new feature extraction criteria and algorithms outperform or have comparable performance with the current methods for cancer gene expression datasets. Furthermore, since the new algorithms here admit more general first-order differential functions as the basis functions instead of kernel functions in SVM-based method, they perhaps have more potential applications in bioinformatics in the future.

Key words: Bioinformatics, differential capability control machine, feature extraction, cancer gene expression datasets, classification

INTRODUCTION

Recent years, bioinformatics has been attracting more and more attentions. DNA micro-array techniques[1-5] play an important role in bioinformatics. With these techniques, great amount of gene expression data including cancer gene expression data have been collected. These data are often analysed using various clustering methods from different viewpoints. In this research, we focus study on classification of cancer gene expression datasets which are publicly available on the internet. It is well known that determining discriminant genes in cancer expression data is of fundamental and practical importance. Research in biology and medicine may greatly benefit from the examination of the discriminant genes to confirm recent advances in cancer research or explore new avenues to cancer diagnosis. Medical diagnostic tests may be derived from a small subset of discriminant genes.

Classification of cancer gene expression data has been extensively studied in the literature[1-4,6-8]. The classification techniques in this literature involve various machine-learning approaches, including unsupervised/supervised learning methods, e.g, the SVM-based method. All these approaches use the given feature-extraction criteria to extract a small subset of highly discriminant genes such that very reliable cancer classifiers can be designed well. Current feature-extraction criteria in these approaches deal with the concepts of correlation coefficients, sensitivity analysis based on second-order differential information and the weights in SVM. Due to the high dimensionality and very small size of cancer gene expression data, classification study on cancer gene expression data still remains a challenging topic.

In this study, we present the new feature-extraction criteria for the data. They are based on the recent differential capability control machine DCCM[9] which appropriately uses the first-order differential information of the decision function and has better generalization capability. This new criteria are quite different from the current criteria in the sense that they first time use the first-order differential information of the decision functions for the data to be classified. Accordingly, we present the new feature-extraction algorithms for the data. Experimental results demonstrate that the new algorithms here have comparable performance with the SVM-based classifier and outperform other classifiers. This application illustrates the new aspect of the applicability of the first-order differential information in decision

Corresponding Author: Wang Shitong, Professor, School of Information Engineering, Southern Yangtze University, Wuxi, China

functions for classification and the significance of this study here exists in that we develop the new feature-extraction criteria and algorithms for cancer gene expression data which may perhaps have potential applications in other gene expression data and even other research fields including data mining and pattern recognition in the future.

NEW FEATURE-EXTRACTION CRITERION NFEC

Related work: Cancer gene expression data has the distinctive characteristics of its high-dimensional number and a relatively small size. In general, its dimensional number corresponding to gene expression values often reaches several thousands to tens of thousands, however, its size is only a few dozen to several hundreds. For such a gene expression data, due to its large number of features and its small size, data overfitting often happens: one can easily find a decision function that separates the training data but then performs very poorly on the test data. In order to overcome the overfitting risk, we should apply machine-learning approaches to reduce its dimensionality. The PCA-based method[10] is often used to do so in which new features that are linear combinations of the original features are obtained by projecting on the first few principal features of the data. The main disadvantage of this method is that none of the original features can be removed among which many features in fact play very small or almost no role for classification. The other method attempts to use pruning techniques such that some of the original features are removed and a minimum subset of features is retained that yield the best classification result. Our concern here focuses on pruning techniques, due to the fact that for the reasons of cost effectiveness and ease of verification of the selected genes, it is very important for us to be able to choose a small subset of genes (i.e, features) in practice.

On applying pruning techniques for cancer gene expression data, ranking features using a given feature-extraction criterion is involved. With such a given criterion, a fixed number of top ranked features may be selected to design a classifier. A threshold can also be set on the feature extraction criterion such that only the features whose criterion exceeds the threshold are retained. At present, several criteria have been developed for cancer gene expression data. Golub *et al.*[1] suggested the following correlation-coefficient criterion:

$$w = (\mu^+ - \mu^-)/\sigma^+ + \sigma^-)$$

where, μ^+, μ^-, σ^+, σ^- denote the mean and standard deviation of the gene expression values of the current gene for all the patients of the positive class and the negative class. Large positive w indicates a strong correlation with the positive class while large negative w indicates a strong correlation with the negative class. One can use this criterion to select an equal number of genes (features) with positive and negative correlation coefficients. Similarly, other researchers[1,3,4] also suggested the following two criteria:

$$w = (\mu^+ - \mu^-)\Big/\sqrt{(\sigma^+ + \sigma^-)} \quad \text{or}$$
$$w = ((\mu^+ - \mu^-))^2/((\sigma^+)^2 + (\sigma^-)^2)$$

The disadvantage of these criteria is that each coefficient w is only computed with information about a single gene (feature) and does not take into account mutual information between genes.

Another type of the feature-extraction criterion is based on sensitivity analysis of the objective functions J for classification problems. LeCun *et al.*[11] gave its first version DJ(I) using second-order differential information of the decision functions for classification problems:

$$DJ(i) = (1/2)\frac{\partial^2 J}{\partial w_i^2}(Dw_i)^2$$

Accordingly, based on SVM (support vector machine), Guyon *et al.*[4] discussed this version with $J = \Sigma \|ax-y\|^2$ where, a is the vector of all weights and suggested the square value of the weight vector itself as a new feature-extraction criterion. Guyon *et al.*[4] also pointed out the relationship between their criterion and LeCun's criterion. Experimental results demonstrated its effectiveness of this new criterion.

It should be noted that LeCun's criterion is built on expanding J in Taylor series to second order, since at the optimum of J, the first order term can be neglected in this Taylor series. However, we can also use the median theorem to express J's expansion, based on the first-order differential information at some unknown point. This fact implies that we perhaps develop a new criterion using the first-order differential information. To date, it is hard to say what is the ideal feature-extraction criterion for cancer gene expression data. Also, due to their high dimensionality of gene expression data in bioinformatics, in order to classify them more accurately, one should process them using different criteria from different angles such that all the obtained results are evaluated to reformulate the final answer. Just like weights and /or second-order differential information, we believe that the first-order differential information should also play a crucial role in extracting features from cancer gene expression data. The experimental results on the New Feature-extraction Criterion (NFEC) below indeed justify this viewpoint.

Differential Capability Control Machine (DCCM) and the New Feature-extraction Criterion (NFEC): Based on the Differential Capability Control Machine (DCCM)[9,12], we will develop a new feature-extraction criterion NFEC for cancer gene expression datasets. First of all, let us introduce the principle that DCCM works briefly.

For the given binary-classification training dataset, $\{(x_i, y_i)|x_i \in R^d, y_i \in \{-1, +1\}, i=1,\cdots,l\}$ according to the SVM framework, it places the decision boundary where the margin is maximized. This decision boundary is represented by the from:

$$f(x, a) = \sum_i a_i K(x_i, x)+a_0$$

where, $a = (a_0, a_1,\cdots\cdots, a_l)$ is the weight vector and $K(x_i, x)$ denotes a kernel function and the support vectors are the learning samples closet to the decision boundary. In other words, SVM improves the generalization capability by maximizing the margin as regards the learning samples already learned. Improving generalization capability is a key step towards solving the learning problem and we believe that generalization capability can be further improved using the general risk functional:

$$C\cdot R_{emp}(a)+F(a) \quad (1)$$

where, $R_{emp}(a)$ and $F(a)$ denote the expected empirical risk function and a convex function with respect to the weigh vector a, respectively and *C* is a constant which controls the trade-off between $R_{emp}(a)$, $F(a)$. Except for the weight vector and the selected kernel functions, the differential information of f (x, a) should also play a key role in enhancing the generalization capability, therefore, in the Differential Capability Control Machine DCCM, we take

$$F(a) = \frac{1}{2}\sum_{i=1}^{l}\|f'(x, a)\|^2 \quad (2)$$

to compute the differential capability of f(x, a). Thus, the risk functional becomes

$$C\cdot R_{emp}(a)+\frac{1}{2}\sum_{i=1}^{l}\|f'(x, a)\|^2 \quad (3)$$

where:

$$\|f'(x_i, a)\|_2 = \left\| \frac{\partial f(x, a)}{\partial x}\Big|_{x=x_i} \right\|^2 \quad (4)$$

$$\frac{\partial f(x, a)}{\partial x}\Big|_{x=x_i} = \left[\frac{\partial f(x, a)}{\partial x^1}\Big|_{x^1=x_i^1} \cdots \frac{\partial f(x, a)}{\partial x^d}\Big|_{x^d=x_i^d}\right]^T \quad (5)$$

where, superscript T denotes the transpositions of the corresponding vector/matrix and subscript i (=1, 2, ……, d) denotes the ith component of x. The differential capability control machine attempts to obtain the weight factor a by minimizing (3) with the given dataset. Without loss of generality, we may take p 1-order differentiable functions gj (x), j = 1, 2,·····, p as the basis functions of f (x, a), that is,

$$f(x, a) = \sum_{i=0}^{p} a_i g_i(x) \quad (6)$$

where, $a = [a_1\ a_2\ \cdots\ a_p]^T$, $g(x) = [g_0(x) \cdots g_p(x)]^T$, p>1 and specially, $g_0(x)$ only takes 0 or 1. $g_0(x) = 1$ means that f (x, a) is a threshold function. In general, g (x) can take the following forms:

Linear type

$$g_i(x) = x_i, \quad i = 1,\cdots,d \quad (7)$$

where, x^i denotes the ith component of x and p = d.

Kernel type

$$g_i(x) = K(x, x_i), \quad i = 1,\cdots, l \quad (8)$$

where, p = l. The most used kernel functions are:

$$K(x, x_i) = \begin{cases} \exp(-\|x-x_i\|^2/2\sigma^2) \\ (1+x^T x_i)^m \\ x^T x_i \end{cases} \quad (9)$$

where, m and σ are the given constants.

Polynomial type

Let x ϵ R,

$$g(x) = [1\ \ x\ \ \cdots\ \ x^p]^T \quad (10)$$

Wavelet type

Let x ϵ R,

$$g_i(x) = h\left(\frac{x-b_i}{\delta_i}\right), \quad i = 1,\cdots, l \quad (11)$$

where, b_i and δ_i denote the parameters in the wavelet function. We can easily extend (10) and (11) to their multidimensional cases.

Let

$$H_{ij} = \sum_{k=1}^{l}\left(\left(\frac{\partial g_i(x)}{\partial x}\Big|_{x=x_k}\right)^T \left(\frac{\partial g_j(x)}{\partial x}\Big|_{x=x_k}\right)\right) \quad (12)$$

then, we can reformulate (3) as:

$$C\cdot R_{emp}(a)+\frac{1}{2}a^T H a \quad (13)$$

where, $a = [a_1\ \ a_2\ \cdots\ a_p]^T$.

If we take the following 1-norm function as the loss function:

$$L(y, f(x, a)) = |f(x, a) - y| \quad (14)$$

and let $\xi_i = L(y_i, f(x_i, a))$, then

$$R_{emp}(a) = \frac{1}{1}\sum_{i=1}^{1}\xi_i \quad (15)$$

Thus, the corresponding decision function can be formulated as:

$$f^*(x) = \mathrm{sgn}(a^T g(x) + a_0) \quad (16)$$

By substituting (15) into (13), we obtain the following optimization problem:

$$\text{PI: min } C\cdot\sum_{i=1}^{1}\xi_i + \frac{1}{2}a^T Ha$$
$$\text{s.t. } \begin{cases} y_i\,(a^T g(x_i) + a_0) \geq 1-\xi_i \\ \xi_i \geq 0,\ i = 1,\cdots,1 \end{cases} \quad (17)$$

and its dual optimization problem is:

$$\text{DI: min } \frac{1}{2}\lambda^T H^*\lambda - \sum_{i=1}^{1}\lambda_i$$
$$\text{s.t. } \begin{cases} \sum_{i=1}^{1}\lambda_i y_i = 0 \\ 0 \leq \lambda_i \leq C,\ i = 1,\cdots,1 \end{cases} \quad (18)$$

where, $\lambda = [\lambda_1, \lambda_2, \cdots, \lambda_1]^T$ is the vector of the Lagrangian multipliers corresponding to PI in (17) and $H^* = GH^+G^T$, where:

$$G_{ij} = g_j(x_i)\cdot y_i, \quad i = 1,\cdots,1,\ j = 1,\cdots,p \quad (19)$$

and H^+ is the general inverse matrix of the matrix H. We can easily derive that:

$$a = H^+ G^T \lambda \quad (20)$$

It should be pointed out that a_0 can be directly obtained by solving DI problem in (18). However, we can get its value using KKT constraints. In terms of KKT constraints, we have:

$$\lambda_i\,[y_i\,(a^T g(x_i) + a_0) - 1 + \xi_i] = 0 \quad (21)$$

If $0 < \lambda_i < C$ then $\xi_i = 0$, thus, we have:

$$a_0 = \frac{1}{|I|}\sum_{i\in I}(y_i - a^T g(x_i)) \quad (22)$$

where, $I = \{i \mid 0 < \lambda_i < C\}$.

By substituting (20) and (22) into (16), we can obtain the corresponding decision function $f^*(x) = \mathrm{sgn}(a^T g(x) + a_0)$. For a new data point to be classified, we can predict its classification easily using this decision function.

As it is easily seen in the above, the generalization capability of DCCM is controlled by the differential capability of the decision function. As shown in[9,12,13], whatever distribution data points in the dataset have, its annealing entropy will go down with the decrease of the differential norm of the corresponding loss function, thus the differential capability can be well controlled. Meanwhile, DCCM may use any first-order differential functions including kernel functions used in SVM as its basis functions and its generalization capability is bounded by the distribution of the data, therefore, it has a better generalization capability than SVM. The experimental results[9,12] also verify this statement. What is more important, based on DCCM, we can develop the NFEC, which is specially appropriate for cancer gene expression datasets.

For the given training dataset, let us observe the differential capability:

$$J = \frac{1}{2}a^T Ha \quad (23)$$

where, a, H are computed using (20) and (12), respectively. The contribution to the differential capability J in (23) from the ith dimension (i.e., the ith feature) can be measured using the following $\Delta J(i)$ in:

$$\Delta J(i) = \left|\frac{1}{2}a^T Ha - \frac{1}{2}a^T H_{-i}\,a\right| \quad (24)$$

where, H_{-i} denotes the matrix which is computed using (12) with the (d-1)-dimensional training dataset taken from the original training dataset without the *i*th dimension. Obviously, we can apply (24) to evaluate the important degree of one feature. According to their values of $\Delta J(i)$, we can rank all features in the increasing order and then remove these features whose do not exceed the given threshold. In other words, $\Delta J(i)$ in (24) provides us one New Feature-extraction Criterion. Experimental results below will demonstrate the effectiveness of this new criterion for cancer gene expression datasets.

FEATURE-EXTRACTION ALGORITHM DCCFE BASED ON THE NEW CRITERION NFEC

Here, we will state the new feature-extraction algorithm DCCFE, based on the Differential Capability Control Machine DCCM and the New Feature-extraction Criterion NFEC. Before the new feature-extraction algorithm is given, we introduce the training and test procedures DCCM_training and DCCM_test of DCCM as follows.

DCCM_training

Input: The binary-classification training dataset

$$\{(x_i, y_i) \mid x_i \in R^d,\ y_i \in \{-1, +1\},\ i = 1,\cdots,1\}$$

Begin

Choose appropriate basis functions and compute the matrix H^*, G;

Solve the PI probem in (18) and compute a and a_0 using (20);

Output the decision function $f^*(x) = sgn(a^T g(x)+a_0)$ using a and a_0 and (16)

End

DCCM_test

Input: The given test dataset and the decision function $f^*(x)$ obtained using the above procedure DCCM_training

Begin

Apply $f^*(x)$ to the test dataset;

Output the prediction accuracy and all the data points with their classification labels in the test dataset

End

The basic idea of the new feature-extraction algorithm is as follows. DCCFE begins with the procedure DCCM_training for the given training dataset. After the decision function is obtained, algorithm DCCFE applies it to the given test dataset using procedure DCCM_training. Meanwhile, algorithm DCCFE uses the new criterion NFEC in (24) to rank all features in decreasing order and then remove those features with the smallest ΔJ (i). According to the remaining features, algorithm DCCFE reorganizes its training dataset from the original training dataset, then repeat the above steps. Such a procedure will be repeated until the optimal feature set with the best prediction accuracy for the test dataset is extracted. Algorithm DCCFE is described in detail as follows.

Algorithm DCCFE:

Step 1 Given the d-dimensional training dataset $X_0 = [x_1, x_2, \cdots, x_l]^T$ and the vector of its classification label $y = [y_1, y_2, \cdots, y_l]^T$ and then the d-dimensional test dataset $XT_0 = [xt_1, xt_2, \cdots, xt_m]^T$ and the vector of its classification label $yt = [yt_1, yt_2, \cdots, yt_m]^T$.

Step 2 Initialize the vector s = [1, 2,⋯, d] that corresponds to all the features of the current training dataset and let the vector r = [ ] that is used to store the corresponding features according to the new criterion NFEC. Initialize the vector pac = [ ] that is used to store the corresponding prediction accuracies which are obtained using procedure DCCM_test for the test dataset. Initialize the vector $s_{optimal}$ = [ ] that is used to record the optimal feature set with the best prediction accuracy $pac_{optimal}$ for the test dataset. Let $pac_{optimal} = 0$.

Step 3 While s ≠ [ ] do

1) Reorganize the |s|-dimensional training dataset $X = X_0(:, s)$, where, " : " denotes all the data points in the dataset X_0 and s stores all the features that the current training dataset X should have.

2) [a, a_0] = DCCM_training (X, y).

3) Classify the test dataset using the decision function as obtained above and compute the corresponding prediction accuracy pac:

   [yt^*, pac] = DCCM_test (XT, yt, X, a, a_0).
   if pac >= $pac_{optimal}$
   {$s_{optimal}$ = s;
   $pac_{optimal}$ = pac;
   }
   Update pac: pac = [pac, pac]

4) Compute the contribution of every feature for the differential capability:
   DJ (s(i)) = ΔJ (s(i)) where:
   ΔJ (s(i)) is computed using (24):

5) s (t) = $\arg\min_{s(i)}$ (DJ);

6) Update r : r = [s (t), r]

7) Update s :
   s = s (1: t=1; t+1: length (s))

endwhile

Step 4 Output pac, r, $pac_{optimal}$, $s_{optimal}$.

It should be pointed out that the above algorithm can easily be generalized to remove more than one feature in each iteration to save its running time.

EXTENSION OF DCCM AND THE FEATURE-EXTRACTION ALGORITHM M_DCCFE FOR MULTI-CLASSIFICATION PROBLEMS

The Differential Capability Control Machine m_DCCM for multi-classification problems: As stated above, the differential capability control machine DCCM is appropriate for binary classification problems. We can easily extend it to its multi-classification version m_DCCM in the OVA (One vs All) way[10] such that m_DCCM can efficiently tackle with multi-classification problems. According to the OVA strategy, every classifier therein is a binary one and the training set of the *i*th classifier contains all data points in the original ith training set as the positive data points, all data points in other original training sets as the negative data points. The decision function F(x) in m_DCCM can be formulated as:

$$F(x) = \arg\max_i(f_1^*(x), \cdots f_i^*(x), \cdots f_m^*(x)) \quad (25)$$

where, $f_i^*(x)$ is the decision function of the corresponding

ith binary-classification DCCM. Here, $f_i^*(x)$ is a little different from the form of the decision function in (16) and it is defined as follows:

$$f_i^*(x) = \frac{a_i^T g(x) + a_{i0}}{\|a_i\|} \quad (26)$$

Unlike the decision function in (16), the value$f_i^*(x)$ here means the membership degree that x belongs to the ith class. The larger $f_i^*(x)$ is, the larger the possibility that x belongs to the *i*th class is.

The new feature-extraction algorithm m_DCCFE based on m_DCCM and the new criterion m_NFEC: Just like algorithm DCCFE, the new feature-extraction algorithm m_DCCFE uses two procedures m_DCCM_training and m_DCCM_test of multi-classification differential capability control machine m_DCCM to train and test the given multi-classification datasets.

m_DCCM_training
Input: The m-classification training dataset

$$\{(x_i, y_i) | x_i \in R^d, y_i \in \{-1, +1\}, i = 1, \cdots, l\}$$

Begin

For the ith binary classifier, choose appropriate basis functions and compute the matrix H^*, G;

Solve the PI probem in (18) and compute a_i and a_{i0} using (20);

Output $A = [a_1, \cdots, a_m]$ and $a_0 = [a_{10}, \cdots, a_{m0}]$ and the corresponding decision function F (x) in (25)
End

m_DCCM_test
Input: The given test dataset and all the decision functions $f_i^*(x)$ obtained using the above procedure m_DCCM_training
Begin

Apply $f_i^*(x)$ to the test dataset;

Output the prediction accuracy and all the data points with their classification labels in the test dataset
End

In order to be appropriate for multi-classification problems, we must extend the above NFEC to its multi-classification version m_NFEC. It is defined using the following (27).

$$\Delta J(i) = \sum_j^m \left| \frac{1}{2} a_j^T H_j a_j - \frac{1}{2} a_j^T H_j^{-i} a_j \right| \quad (27)$$

where, H_j^{-i} is obtained using (12) for the jth binary-classification training dataset which originates from the original m-classification training dataset. With the above criterion m_NFEC and the two procedures m_DCCM_training and m_DCCM_test, we extend algorithm FCCFE to its multi-classification version---- algorithm m_DCCFE as follows.

Algorithm m_DCCFE:
Step 1 Given the d-dimensional training dataset $X_0 = [x_1, x_2, \cdots, x_l]^T$ and the vector of its classification label $y = [y_1, y_2, \cdots, y_l]^T$ and then the d-dimensional test dataset $XT_0 = [xt_1, xt_2, \cdots, xt_m]^T$ and the vector of its classification label $yt = [yt_1, yt_2, \cdots, yt_m]^T$.

Step 2 Initialize the vector s = [1, 2,···, d] that corresponds to all the features of the current training dataset and let the vector r = [ ] that is used to store the corresponding features according to the new criterion m_NFEC. Initialize the vector pac = [ ] that is used to store the corresponding prediction accuracies which are obtained using procedure m_DCCM_test for the test dataset. Initialize the vector $s_{optimal}$ = [ ] that is used to record the optimal feature set with the best prediction accuracy $pac_{optimal}$ for the test dataset. Let $pac_{optimal} = 0$.

Step 3 While s ≠ [ ] do

1) Reorganize the | s |-dimensional training dataset $X = X_0(:, s)$, where, " : " denotes all the data points in the dataset X_0 and s stores all the features that the current training dataset X should have.

2) $[A, a_0]$ = m_DCCM_test (X, y):

3) Classify the test dataset using the decision function F(x) in (25) as obtained above and compute the corresponding prediction accuracy pac:

   $[yt^*, pac]$ = m_DCCM_test ((XT, yt, X, A, a_0)

   if pac> = $pac_{optimal}$
   { $s_{optimal}$ = s;
   $pac_{optimal}$ = pac;
   }

   Update pac: pac = [pac, pac].

4) Compute the contribution of every feature for the differential capability: DJ (s(i)) = ΔJ (s(i)) where, ΔJ (s(i)) is computed using (27)

5) $s(t) = \arg \min_{s(i)} (DJ)$

6) Update r : r = [s (t), r]

7) Update s : s = s (1: t-1; t+1: length (s));
endwhile

Step 4 Output pac, r, $pac_{optimal}$, $s_{optimal}$.

RESULTS

We present experimental results on two cancer gene expression datasets in which one is for binary classification and the other for multi-classification. Both datasets are high dimensional with very small sizes. Although both datasets seem to be relatively easy to separate, they will also face several difficulties, including very small sizes and different distributions between training and test datasets. Our experimental results will validate the effectiveness of the new feature-extraction criteria and algorithms here.

Experiment 1: This experiment is carried out on the benchmarking leukemia data[1,3,4,6,7] (www.genome.wi.mit.edu/cancer/data_set_ALL_AML.html). This data contains two aspects: the training dataset, used to select genes and tune the weights in the decision functions of the classifiers and the test dataset used to evaluate the performances of the obtained classifiers. The training dataset consists of 38 samples (27 ALL and 11 AML) from bone marrow specimens. The test dataset consists of 34 samples (20 ALL and 14 AML) including 24 bone marrow and 10 blood sample specimens. All samples have 7129 features (i.e., genes), corresponding to some normalized gene expression value extracted from the micro-array image. We preprocessed this leukemia data such that the mean of every feature is 0 and its variance is 1.

We extracted the features from this leukemia data respectively using the WV (weight voting) classifier[1-3], the KP (kernel perceptron) classifier[14,15] and the DCCM here and then carried out the comparison among their test performances. We arranged the following two test methods to assess the corresponding test performances.

TS method: In this method, the classifiers are built only from the training dataset and then the test dataset is passed to the classifiers to obtain their prediction accuracies.

LOU method: The so-called leave-one-out procedure LOU is used in this method. The leave-one-out procedure consists of removing one example from the training dataset, constructing the decision function only on the basis of the remaining training data and then testing on the removed example. In this way, all examples of the training data are tested and the prediction accuracies are measured for the classifiers.

Table 1 demonstrates the prediction accuracies of four classifiers for the case where the new algorithm DCCFE was used to extract features from the leukemia dataset. Table 2 shows the prediction accuracies of the four classifiers for the other case where algorithm SNRFE (Feature Extraction based on Signal Noise Rate) was used to extract features from the same dataset. We used the same linear kernel function for the KP, DCCM and SVM classifiers for ease of comparison. With Table 1, we can easily see that when the new feature-extraction algorithm DCCFE was used to extract features from the leukemia dataset, all the corresponding three classifiers DCCM, WV and KP could achieve 100% prediction accuracies for both test-set and LOU test methods, however, the DCCM classifiers is more stable in the sense that this prediction accuracy 100% is achieved with the more combinations of genes. In other words, the WV classifier and KP classifier achieved 100% prediction accuracy only when the number

Table 1: Prediction accuracies of WV, KP, DCCM and SVM based on algorithm DCCFE

	WV	Classifier	KP	Classifier	DCCM	Classifier	SVM	Classifier
No. of genes	TS pac	LOU pac	TS pac	LOU pac	TS pac	LOU pac	TS pac	LOU pac
7129	0.85	0.89	0.91	0.89	0.97	0.97	0.97	0.97
4096	0.88	0.97	0.85	0.94	0.97	1.00	0.97	1.00
2048	0.88	1.00	0.88	0.97	0.97	1.00	0.97	1.00
1024	0.94	1.00	0.91	1.0	**1.00**	**1.00**	**1.00**	**1.00**
512	0.91	1.00	0.88	0.97	**1.00**	**1.00**	**1.00**	**1.00**
256	0.94	1.00	0.91	1.00	0.97	1.00	0.97	1.00
128	0.94	1.00	0.97	1.00	**1.00**	**1.00**	**1.00**	**1.00**
64	1.00	1.00	0.97	1.00	**1.00**	**1.00**	**1.00**	**1.00**
32	0.94	1.00	0.94	1.00	0.94	1.00	0.97	1.00
16	0.94	1.00	0.97	1.00	**0.91**	1.00	0.94	1.00
8	0.97	1.00	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
4	0.94	0.97	0.88	1.00	0.91	1.00	0.85	1.00
2	0.91	0.97	0.91	0.97	0.91	1.00	0.91	1.00
1	0.91	0.92	0.41	0.26	0.91	0.97	0.91	0.97

of gense takes 64 and 8, respectively, however, the DCCM classifier did so when the number of gensetakes 1024, 512,128, 64, 8. Furthermore, Table 1 also indicates that both DCCM and SVM classifiers achieved 100% prediction accuracy in the same cases, however, the smallest prediction accuracy for DCCM using the test-set method is 0.91, which is bigger than 0.85 for SVM using the same test method. Table 2 clearly indicates that the DCCM classifier outperformed the WV and KP classifiers and has the comparable performance with the SVM classifier in the best prediction accuracy when algorithm SNRFE was used to extract features from this dataset. Especially, when the number of genes is bigger, both the DCCM and SVM classifiers have the same prediction accuracies. When the number of genes is relatively smaller, the SVM classifier seems to be better than the DCCM classifier, however, the worst case (0.82) of the DCCM classifier is better than that (0.79) of the SVM classifier. Therefore, on average, both the DCCM and the SVM classifiers are comparable in prediction accuracy. Table 3 summarizes all the cases of the numbers of genes where the WV, KP, DCCM and SVM achieved their best prediction accuracies with the uses of algorithm SNRFE and algorithm DCCFE. Obviously, algorithm DCCFE has the noticeable advantage over algorithm SNRFE

Let us give a little more remarks on our algorithm DCCFE here and the SVM-based method[4]. In summary, from the above results, we see that there is no significantly different performance between them. This fact demonstrates that our algorithm here has the comparable performance with the SVM-based method. Since the SVM-based method is based on the weight information in the corresponding decision function and our algorithm is based on the first-order differential information of the corresponding decision functions, so, this fact also indicates that the first differential information has the comparable role with the weight information and is worthy to cultivate its role in feature-extraction study.

Experiment 2: This experiment is used to validate the effectiveness of algorithm m_DCCFE for multi-classification cancer gene expression data. The benchmarking dataset can be downloaded from http://www.genome.wi.mit.edu/cancer/data_set_ALL_AML.html. The cancer gene expression dataset consists of 144 training samples and 46 test samples, containing different cancer gene expression data of 14 classes as shown in Table 4. Every sample has more than 16000 features (genes).

Table 2: Prediction accuracies of WV, KP, DCCM and SVM based on algorithm SNRFE

	WV	Classifier	KP	Classifier	DCCM	Classifier	SVM	Classifier
No. of Genes	TS pac	LOU pac	TS pac	LOU pac	TS pac	LOU pac	TS pac	LOU pac
7129	0.85	0.89	0.91	0.89	0.97	0.97	0.97	0.97
4096	0.85	0.94	0.88	0.94	0.97	1.00	0.97	1.00
2048	0.88	0.97	0.88	0.97	0.97	1.00	0.97	1.00
1024	0.94	1.00	0.91	1.00	0.97	1.00	0.97	1.00
512	0.94	1.00	0.91	0.97	0.97	1.00	0.97	1.00
256	0.91	1.00	0.97	0.97	0.97	1.00	0.97	1.00
128	0.91	1.00	0.94	0.97	**0.97**	**1.00**	0.97	1.00
64	0.94	1.00	0.94	0.97	0.94	1.00	**0.97**	**1.00**
32	**0.97**	**0.97**	0.94	1.00	0.91	1.00	0.94	1.00
16	0.91	1.00	**0.97**	**0.97**	0.91	1.00	0.91	1.00
8	0.91	0.97	0.91	1.00	0.91	1.00	0.91	1.00
4	0.88	1.00	0.91	0.94	0.91	0.92	0.94	0.97
2	0.79	0.97	0.76	0.97	0.82	0.92	0.85	0.97
1	0.76	0.97	0.82	0.86	0.82	0.86	0.79	0.92

Table 3: Best prediction accuracies of the WV, KP, DCCM and SVM classifiers using algorithm SNRFE and algorithm DCCFE

	Feature											
	WV			KP			DCCM			SVM		
Extraction algorithms	TS pac	LOU pac	No. genes	TS pac	LOU pac	No. genes	TS pac	LOU pac	No. of genes	TS pac	LOU pac	No. of genes
DCCFE	1.00	1.00	64	1.00	1.00	8	1.00	1.00	8	1.00	1.00	8
									64			64
									128			128
									512			512
									1024			1024
SNRFE	0.97	0.97	32	0.97	0.97	16	0.97	1.00	128	0.97	1.00	64

Table 4: 14-classc ancer gene expression dataset

Class No.	Cancer type	No. of the training samples	No. of the test samples
0	Breast	8	3
1	Prostate	8	2
2	Lung	8	3
3	Colorectal	8	5
4	Lymphoma	16	6
5	Bladder	8	3
6	Melanoma	8	2
7	Uterus	8	2
8	Leukemia	24	6
9	Renal	8	3
10	Pancreas	8	3
11	Ovary	8	3
12	Mesothelioma	8	3
13	Brain	16	4

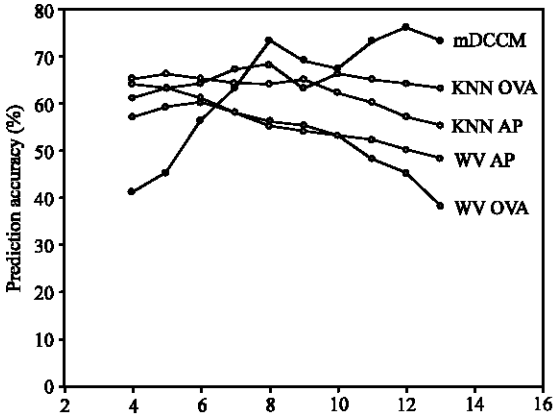

Fig. 1: Change curves of prediction accuracies obtained using several multi-classifiers

We applied the new algorithm m_DCCFE with the linear basis functions in (7) to the 114 training samples of the above. We applied the new algorithm m_DCCFE with the linear basis functions in (7) to the 114 training samples of the above dataset and then obtained the features to be extracted and the corresponding decision functions. The 46 test samples were passed to procedure m_DCCM-test to obtain the prediction accuracy. We carried out a comparison with the results[2] where, multi-classifiers WV OVA, WV AP, KNN OVA, KNN AP were used based on algorithm SNRFE. Figure 1 shows that with algorithm m-DCCFE, the best prediction accuracy of the m_DCCM classifier achieved 76%, however, all other classifiers had less than 70% best accuracies. Moreover, when the number of genes is bigger than 256, the best prediction accuracies of the m_DCCM always exceed those of all other classifiers. It is noted that too small number of features (genes) to be extracted should be inappropriate for such a dataset with more than 16000 features. The above statements strongly indicate that the new algorithm m_DCCFE has the distinctive advantage over the current algorithms for multi-classification cancer gene expression data.

CONCLUSIONS AND FUTURE WORK

In this study, we developed the new feature-extraction criteria and algorithms for the data with a large number of features and a small training size. The new criteria and algorithms utilize the first-order differential information of the decision functions for the data and are based on the differential capability control machine DCCM. We demonstrated experimentally on cancer gene expression datasets and the results indicated that the new feature-extraction algorithms outperform or have comparable performance with the current methods. This work here also reveals that the first-order differential information of the decision functions for the data has the same crucial role as the weight information in the SVM-based method.

In contrast to SVM and its variants, DCCM uses first-order differential functions as its basis functions instead of kernel functions, therefore, the new criteria and algorithms have more general significance. Future work includes experimenting with this criteria and algorithms to other data in bioinformatics and even perhaps to pattern recognition and data mining.

ACKNOWLEDGMENTS

This study was supported by the RGC Competitive Earmarked Research Grant (grant No. PolyU 5065/98E), Natural Science Foundation of China (grant No. 60225015), Natural Science Foundation of JiangSu Province (grant No. BK2003017), National Key Lab. of Novel Software Technologies at NanJing University and Nation Key Lab. of Computer Science at Institute of Software of CAS SINICA (SYSKF0406) and excellent teacher grant of Minstry Education of China.

REFERENCES

1. Golub, T.R., D.K. Slonim and P. Tamayo *et al.*, 1999. Molecular classification of cancer: Class discovery and class prediction by gene. Science, 286: 531-537.
2. Ramaswamy, S., P. Tamayo and R. Rifkin *et al.*, 2001. Multi-class cancer diagnosis using tumor gene expression signatures. PNAS, 98: 15149-15154.
3. Tamayo, P. and S. Ramaswamy, 2003. Cancer Genomics and Molecular Pattern Recognition. M. Ladanyi and W. Gerald (Eds.). Expression Profiling of Human Tumors: Diagnostic and Research Applications. Humana Press, 2003.

4. Guyon, I., J. Weston and S. Barnhill *et al*., 2002. Gene selection for cancer classification using support vector machines. Machine Learning, 46: 389-422.
5. Tamayo, P., D. Slonim and J. Mesirov *et al*.1999. Interpreting gene expression with self-organizing maps: Methods and application to hematopoeitic differentiation. Proc. Natl. Acad. Sci., USA., 96: 2907-2912.
6. Dudoit, S. and J. Fridly, 2003. Bagging to improve the accuracy of a clustering procedure. Bioinformatics, 19: 1090-1099.
7. Jornsten, R. and B. Yu, 2003. Simultaneous gene clustering and subset selection for sample classification via MDL. Bioinformatics, 19: 1100-1109.
8. Kohavi, R. and G. John, 1997. Wrappers for feature subset selection. Artificial Intelligence, pp: 273-324.
9. Zhang Li *et al*., 2003. The differential capability control machine. Chinese J. Elec., 31: 1526-1531.
10. Scholkopf, B., A. Smola and K.R. Muller, 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 10: 1299-1319.
11. LeCun, Y. and J. Denker *et al*., 1990. Optimal Brain Damage. In: Touretzky, D., (Ed.). Advances in Neural Information Processing Systems 2, CA: Morgan Kaufman, pp: 598-605.
12. Wang Shitong, 1998. Fuzzy Systems, Fuzzy Neural Networks and Their Programming. Press of Shanghai Science and Technology.
13. Vapnik, V., 1998. Statistical Learning Theory. New York, Wiley and Sons.
14. Xu, Ji., *et al*., 2002. A nonlinear kernel preceptron. Chinese J. Comp. Sci. and Technol., 25: 689-695.
15. Chen, J.H. and C.S. Chen, 2002. Fuzzy kernel perceptron. IEEE Trans. on Neural Networks, 13: 1364-1373.