Abstract

Gene expression data play an important role in the development of efficient cancer diagnoses and classification. The genes identified are subsequently used to classify independent test set samples. The different feature selection methods are investigated and most frequent features are selected among all methods. This paper provides gene selection strategies for multi-class classification that can be used to reach high prediction accuracies with a tiny low number of selected genes. In this paper, a multi-objective biogeography based optimization method is proposed to select the small subset of informative gene relevant to the classification. In the proposed algorithm, firstly, the KNN (K's Nearest Neighbour) algorithm is used to choose the 60 top gene expression data. Secondly, to make biogeography based optimization suitable for the discrete problem, binary biogeography based optimization, as called BBBO, is proposed based on a binary migration model and a binary mutation model. Then Core Vector Machine (CVM), is proposed by integrating the non-dominated sorting method and the crowding distance method into the BBBO framework. In order to show the effective and efficiency of the algorithm, the proposed algorithm is tested on ten gene expression dataset benchmarks. Experimental results demonstrate that the proposed method is better or at least comparable with previous particle swarm optimization (PSO) algorithm and support vector machine (SVM) from literature when considering the quality of the solutions obtained.

1. Introduction

Microarray gene expression experiments help to measure the expression levels of thousands of genes simultaneously in diagnosing various types of tumors with better accuracy [1]. These experiments usually generate a lot of complex data, which happens to be its major limitation. Gene expression data have characteristics of high-dimensional, high-noise and small-sample size, which make it difficult to develop an efficient classifier [2]. In this sense, gene selection, as called feature selection in computational intelligence field [3], [4], is often considered as a necessary preprocess step to analyze these data, because this method can reduce the dimensionality of the data and often conduct to better analyze. An important application of gene expression is to classify samples according to their gene expression profiles, such as the diagnosis or the classification of different types or subtypes of cancer. Different classification methods from statistical and machine learning have been applied to the classification of cancer. However, high dimensionality and a small number of noisy samples pose great challenges to existing methods. Most of the classifiers involve complex models containing numerous genes. This has limited the interpretability of the classifiers and this lack of interpretability the acceptance of diagnostic tools. Classification models based on numerous genes can also be more difficult to transfer to other platforms, which may be more suitable for clinical application. Moreover, some more complex algorithms based on numerous genes for classification often over fit the data.
Prior to classification, a variety of gene selection strategies have been used. The aim of gene selection is to select a small subset of genes from a larger pool. Gene selection methods are classified into three types: (1) filter methods, (2) wrapper methods, and (3) embedded methods. Filter methods evaluate a subset of genes by looking at the intrinsic characteristics of data with respect to class labels, while wrapper methods evaluate the goodness of a gene subset by the accuracy of its learning or classification. Embedded methods are generally referred to as algorithms, where gene selection is embedded in the construction of the classifier. In the gene selection process, an optimal feature subset is always relative to a certain criterion. Every criterion measures the discriminating ability of a gene or a subset of genes to distinguish different class labels. To measure the gene–class relevance, different statistical and theoretical measures such as the t-test, entropy and mutual information are typically used, and different metrics including the Euclidean distance and correlation coefficient are employed to calculate the gene–gene redundancy. In filters, the characteristics in the feature selection are uncorrelated to that of the learning methods, therefore they have better generalization property[1]. The filters, wrapper and embedded are then analyzed to identify the most frequently appearing genes which would correspond to the most predictive genes[2].

In wrapper type methods, feature selection is "wrapped" around a learning method and a feature is directly judged by the estimated accuracy of the learning method [11]: One can often obtain a set with a very small number of non-redundant features, which gives high accuracy, because the characteristics of the features match well with the characteristics of the learning method [14].Wrapper methods can use different performance metrics and objective functions. And also the wrapper methods select the “minimum” subset of features that provides the highest sensitivity. Embedded methods differ from other feature selection methods in the way feature selection and learning interact [26]. For example trees can be applied in ubiquitous scenarios so that they provide a good entry point for feature selection for interdisciplinary on large data sets. It can be applied to either a numerical or a categorical response. Decision tree also provides an embedded measure of variable importance that can be obtained from the number and the quality of splits that are generated from a predictor variable [27]. In contrast to filter and wrapper approaches, in embedded methods the learning part and the feature selection part cannot be separated - the structure of the class of functions under consideration plays a crucial role [22].

II. Materials and Methods

A. K’s Nearest Neighbor

K Nearest Neighbor (KNN) is a simple algorithm, which stores all cases and classify new cases based on similarity measure. K-nearest neighbor is a supervised learning algorithm where the result of new instance query is classified based on majority of K-nearest neighbor category. The purpose of this algorithm is to classify a new object based on attributes and training samples. KNN uses neighborhood classification as the prediction value of the new query instance.

The number of nearest neighbors can be specified clearly in the object editor or determined automatically using leave-one-out cross-validation focus to an upper limit given by the specified value. Different kinds of search algorithms can be used to speed up the task of finding the nearest neighbors.

In structure less KNN techniques the whole data is classified into training and testing sample data. From training point to sample point, distance is evaluated, and the point with lowest distance is called nearest neighbor. Structure based NN techniques are based on structures of data like orthogonal structure tree (OST), ball tree, k-d tree, axis tree, nearest future line and central line. Nearest neighbor classification is used mainly when all the attributes are in continues sequence.

K-nearest neighbor is a non-parametric classification method that predicts the sample of a test case. To apply K-nearest neighbor each sample was represented by a pattern of expression that consists of genes with tissues (Hua et al 2001). Each sample was then classified according to the class memberships of its k nearest neighbors, as determined by the Euclidean distance in the d-dimensional space.

The distance function is used as a parameter of the search method. The remaining thing is the same as for IBL—that is, the Euclidean distance; other options include Chebeshev, Manhattan, and Minkowski distances. Predictions from more than one neighbor can be weighted according to their distance from the test instance and two different formulas are implemented for converting the distance into a weight.

B. Support Vector Machine

The ability of support vector machine is to deal with high dimensional data. The four different kernels are used for testing the genes. SVM try to find an optimal gene separating hyperplane between the classes. When the classes are linearly separable, the hyperplane is located so that it has maximal margin which should lead to better performance on data not yet seen by the SVM.
When the data are not separable, there is no separating hyperplane; in this case it tries to maximize the positive genes but allow some classification errors to the constraint that the total error is less than a negative gene. There are several possible approaches; In support vector machine method "one against- one" approach, as implemented in "libsvm" [12] genes as predictors tended to perform as well as, or better than, smaller numbers. Guyon used the support vector machine as a tool for discovering informative patterns[4]. SujunHua et al[23] , represented a new approach to supervised pattern classification applied to a pattern recognition problems, including object recognition, speaker identification, gene function prediction with microarray expression profile, etc. Minh N. Nguyen et al[14]., investigates the multi-class SVM methods involved to resolve a much larger optimization problem and are applicable to small datasets. The multi-class SVM methods are more suitable for prediction[25] than the other methods. Duval extend the prediction accuracy by adding a second-stage multi-class SVM to capture the information among the genes [12].

Cross-validation: Cross-Validation (CV) is very helpful in evaluating and comparing learning algorithms. It is a statistical technique used during the training process of the classifier where its task is to divide the train dataset into two segments; one is used for training and the other is used for validation [12].

SVM-classifier: Support vector machine uses the SVM structure to classify the test data into the predefined classes. As the cross validation and SVM parameters are accurately chosen, as the classification accuracy of the test samples increases [18].The cross-validation coupled with the SVM-trainer runs several times in a continuous loop until reaching maximum train classification accuracy [22].

In particular we identify and describe pairs of genes that are much better suited for separating the diseased and the healthy samples as compared to other single genes .The high dimensionality, over fitting, sensitivity and a small number of noisy samples pose significant challenges in classification.

C. MOBBBO-based feature selection and CVM parameters optimization

When the SVM is used, two problems need to be solved: how to choose the optimal input feature subset for SVM, and how to choose the suitable kernel parameters. These two problems are important because the feature subset affects the parameters. Based on the analysis of the multi-objective binary biogeography based optimization, BBO algorithm has the potential to generate both the optimal feature subset and SVM parameters at the same time. The research objective is to optimize the parameter and feature subset simultaneously, without reducing the testing accuracy rate of the SVM.

In the existing work a hybrid multi-objective binary biogeography based optimization with SVM, called MOBBBO SVM is developed, for parameter determination and feature selection using gene expression data. Despite of the parameters of feature selection, two parameters of kernel function, designated c and r, are required. The position of each individual is represented by a binary (0/1) string except the two dimensions of c and r. In the first phase, the MOBBBO algorithm provides a binary encoded individual where each bit represents a gene. If a bit is 1, it denotes this gene is kept in the subset; else if at bit is 0, it represents a non-selected feature. Therefore, the individual length in the initial microarray dataset. Then, the fitness each individual is assessed by the accuracy of leave-one-out cross-validation method (LOOCV). The leave-one-out cross-validation method can be described as follows: when there are data to be classified, the data are divided into one testing sample and n-1 training samples. Each individual will be selected as a testing sample in turn. The other n-1 individuals serve as the training data set to determine the prediction parameter of the model.

To overcome the limitations in the existing system, an efficient method is proposed by combing MOBBBO with CVM (Core Vector Machine). Our novel approach is called as IMOBBBO-CVM (Improved Multi-objective binary biogeography based optimization- Core Vector Machine). CVM is a fast classification algorithm orientated to large-scale sample data. The CVM classification method can improve the accuracy of classification even in a large scale of gene expression data. In addition, IMOBBBO-CVM models need less numbers of support vectors and core vectors. This is a great advantage in processing large-scale gene expression data. This proposed approach can reduce the time complexity. Standard SVM training has $O(n^2)$ time and $O(n^3)$ space complexities, where $m$ is the training set size. It is thus computationally infeasible on very large data sets. The proposed Core Vector Machine (CVM) algorithm can be used with nonlinear kernels and has a time complexity that is linear in $m$ and a space complexity that is independent of $m$. CVM solution becomes closer to the exact optimal solution. The proposed CVM procedure is simple, and does not require sophisticated heuristics as in other decomposition methods. Moreover, despite its simplicity, CVM has small asymptotic time and
space complexities. This can be further improved when probabilistic speedup is used. Experimentally, it is as accurate as existing SVM implementations, but is much faster and produces far fewer support vectors (and thus faster testing) on large data sets.

CVM, proposed, is a fast classification algorithm oriented to large-scale sample data. It first converts problem of SVM into Minimum Enclosing Ball (MEB) problem, and then uses an iterative \((1 + \epsilon)\)-approximation algorithm to solve the MEB problem. It has obvious advantages in pattern recognition problems with large-scale sample data and complex nonlinearity. Suppose a set of points \(x_1, x_2, \ldots, x_n\) is given, where \(x_i \in \mathbb{R}^d\). All these points can be mapped into a high-dimension feature space by a function \(\phi\) and thus a set of mapped points \(\{\phi(x_1), \ldots, \phi(x_n)\}\) are produced. In the feature space, its MEB can be denoted by \(B(c^*, R^*)\) with the center \(c^*\) and radius \(R^*\). Then, the problem of searching the smallest ball that encloses all the mapped points is

\[
e^\epsilon(c', R') = \arg \min_{c, R} \|c - \phi(x_i)\| \leq R, \quad \forall i = 1, \ldots, n
\]

After formulating the kernel method as a MEB problem, we obtain a transformed kernel \(\tilde{k}\), together with the associated feature space \(\tilde{F}\), mapping \(\phi\) and constant \(k = \langle k(z, z)\). To solve this kernel-induced MEB problem, we adopt the approximation algorithm. The idea is to incrementally expand the ball by including the point furthest away from the current center. In the following, we denote the core set, the ball’s center and radius at the \(t\)th iteration by \(S_t, c_t\) and \(R_t\) respectively. Also, the center and radius of a ball \(B\) are denoted by \(c_B\) and \(r_B\). Given an \(\epsilon > 0\), the CVM then works as follows:

1. Initialize \(S_0, c_0\) and \(R_0\).
2. Terminate if there is no training point \(z\) such that \(\phi(z)\) falls outside the \((1+\epsilon)\)-ball \(B(c_t, (1+\epsilon)R_t)\).
3. Find \(z\) such that \(\phi(z)\) is furthest away from \(c_t\). Set \(S_{t+1} = S_t \cup \{z\}\).
4. Find the new MEB \(S_{t+1}\) and set \(c_{t+1} = c_{\text{MEB}}(S_{t+1})\) and \(R_{t+1} = r_{\text{MEB}}(S_{t+1})\).
5. Increment \(t\) by 1 and go back to Step 2.

In the sequel, points that are added to the core set will be called core vectors. Despite its simplicity, CVM has an approximation guarantee and small time and space complexities.

### III. Results And Discussions

Selecting relevant genes for gene expression classification is a common challenge in bioinformatics. In this study, ten gene expression data, including tumor samples, brain tumor, leukemia, lung cancer, and prostate tumor samples, are used in our experiments and their characteristics are summarized in Table II, which can be downloaded from http://www.gensys-system.org. The data sets are partitioned into one testing sample-1 and training samples. Note that each individual will be selected as a testing sample in turn. The remaining \(n-1\) individuals will then serve as the training data set.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Number of samples</th>
<th>Number of classes</th>
<th>Number of genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tumor</td>
<td>9</td>
<td>9</td>
<td>3720</td>
</tr>
<tr>
<td>Tumors</td>
<td>11</td>
<td>11</td>
<td>12333</td>
</tr>
<tr>
<td>Brain tumor</td>
<td>90</td>
<td>5</td>
<td>5320</td>
</tr>
<tr>
<td>Brain tumor2</td>
<td>50</td>
<td>4</td>
<td>18407</td>
</tr>
<tr>
<td>Leukemia</td>
<td>77</td>
<td>2</td>
<td>5469</td>
</tr>
<tr>
<td>Leukemia2</td>
<td>72</td>
<td>3</td>
<td>5327</td>
</tr>
<tr>
<td>Lungcancer</td>
<td>203</td>
<td>5</td>
<td>12122</td>
</tr>
<tr>
<td>Prostate tumor</td>
<td>105</td>
<td>2</td>
<td>16200</td>
</tr>
<tr>
<td>SEBCT</td>
<td>65</td>
<td>4</td>
<td>2398</td>
</tr>
</tbody>
</table>

The methods are used in the existing scenario is called MOBBBO with SVM and proposed system is named as MOBBBO-CVM which measures the performance metric. The performance metrics are such as accuracy, precision, recall, f-measure and time factor values. From the experimental result we can conclude that our proposed system superior rather than existing scenario.

**Accuracy**

Accuracy can be calculated from formula given as follows

\[
\text{Accuracy} = \frac{\text{True positive} + \text{True negative}}{\text{True positive} + \text{True negative} + \text{False positive} + \text{False negative}}
\]

**Precision**

Precision value is calculated based on the retrieval of information at true positive prediction, false positive. In healthcare data, precision is calculated based on the percentage of positive results returned that are relevant.
Precision = \frac{\text{True positive}}{\text{True positive} + \text{False positive}}

Recall

Recall value is calculated based on the retrieval of information at true positive prediction, false negative. In healthcare data recall is calculated as the percentage of positive results returned that are Recall in this context is also referred to as the True Positive Rate. Recall is the fraction of relevant instances that are retrieved.

Recall = \frac{\text{True positive}}{\text{True positive} + \text{False negative}}

- **TP (True positive)**

If the outcome from a prediction is p and the actual value is also p, then it is called a true positive (TP). It is the ability which is used to find the high true-positive rate. The true-positive rate is also called as sensitivity. The true positive rate or the recall rate in some fields measures the proportion of actual positives which are correctly identified.

True positive rate (TPR) = \frac{TP}{P}

P = (TP + FN)

Where, P is the positive. TP is the True Positive.

- **TN (True negative)**

A true negative (TN) has occurred when both the prediction outcome and the actual value are n in the number of input data.

- **FP (False positive)**

If the outcome from a prediction is p and the actual value is n then it is said to be a false positive (FP). The False Positive rate, also known as the false alarm ratio, usually refers to the probability of falsely rejecting the null hypothesis for a particular test. The false positive rate usually refers to the expectancy of the false positive ratio.

False positive rate (FP) = \frac{FP}{FP + TN}

- **FN (False negative)**

False negative (FN) is when the prediction outcome is n while the actual value is p.

**F-Measure**

A measure that combines precision rate and recall rate is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

\[ F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \]

There are several reasons that the F-score can be criticized in particular circumstances due to its bias as an evaluation metric. This is also known as the F_{1} measure, because recall and precision are evenly weighted.

**Time:**

The proposed system is evaluated in terms of the time complexity in other words computation time of the feature selection technique with the existing system and proposed system. It is defined as the time taken for the feature selection process.
### IV. CONCLUSION

In this paper, a Core Vector Machine (CVM) is proposed for gene selection on ten gene expression datasets. Experimental results show that the algorithm can simplify feature selection by finding a smaller number of features needed effectively and a higher classification accuracy compared with other previous methods. The proposed algorithm can obtain the highest accuracy in nine of the ten microarray dataset problems since the multi-objective approach can find a diverse solution. Moreover, the results show that there are many irrelevant genes in gene expression data and some of them are not relevant to a given cancer. For further work, the proposed algorithm can be applied to some problems in other fields.

### REFERENCES


