Feature Selection and Gene Clustering from Gene Expression Data

Pabitra Mitra
Machine Intelligence Unit
Indian Statistical Institute
Kolkata 700108
e-mail: pabitra_r@isical.ac.in

Dwijesh Dutta Majumder
Electronics & Communication Sciences Unit
Indian Statistical Institute
Kolkata 700108
e-mail: ddm@isical.ac.in

Abstract

In this article we describe an algorithm for feature selection and gene clustering from high dimensional gene expression data. The method is based on measuring similarity between features/genes whereby redundancy therein is removed. This does not need any search and therefore is fast. A novel feature similarity measure, called maximum information compression index, is used. The feature selection algorithm also obtains gene clusters in a multiscale fashion. The superiority of the algorithm, in terms of speed and performance, is established on a real life molecular cancer classification dataset.

Keywords: Microarray, maximal information compression index, cancer classification, representation entropy, data mining.

1. Introduction

Analysis of the gene expression provide a rich source of data from which inferences about both overall cell function and the function of individual gene components can be drawn. Gene expression data is obtained by extraction of quantitative information from the images/patterns resulting from the readout of fluorescent or radioactive hybridizations in an microarray chip [1]. An important analysis task for this data is feature selection or identification of the genes which are significant or mostly associates with a tissue category or disease [4]. A related task is gene clustering or partitioning of genes into well separated and homogeneous groups based on the statistical behaviors of their expressions [3]. The objective of clustering analysis is convenience of understanding and visualization, or to find out the functional role and regulatory control of a novel gene, based on other better characterized members of its groups.

Challenges to feature selection and gene clustering include high dimensionality of the data, obtained from ten thousands of genes which are studied in a single experiment [7]. Many of these genes may be irrelevant and does not carry any information. Existing algorithms for feature selection searches out the best subset of features based on some evaluation criterion like information gain or classification accuracy. However, the search algorithms have high computational cost and are infeasible for high dimensional data. A Markov blanket based approach has been proposed in [7] to circumvent this problem. From the point of view of gene clustering, genome wide collection of expression trajectories lack natural clustering structure, giving rise to difficulty in determining the number of clusters to be generated. Here, one needs a clustering algorithm which is fast and scalable to high dimension, and can produce natural clusters at different scales of detail. Popular algorithms for clustering involve measuring similarity between a pair of genes and then using the similarity values to partition the genes using data clustering algorithms like k-medoid and PAM [3]. These algorithms often produce poor quality clusters which fail to capture the natural grouping of the genes.

In this article we describe a feature selection algorithm based on the principle of redundancy reduction, which can also produce natural gene clusters at different levels of detail i.e., in a multiscale manner. A novel gene similarity measure, called maximal information compression index, is used in clustering. Experimental results are presented for a benchmark cancer classification problem. Before we explain the feature selection and gene clustering algorithm we describe the similarity measure in the next section.

2. Maximal Information Compression Index

In this section we present an index for measuring dissimilarity between expression levels of two genes/features over a series of experiments. The two expression levels are considered as two random variables x and y, and the measure is computed from the linear (in)dependency between them. The index is used for subsequent gene clustering.

Let Σ be the covariance matrix of the random variables x and y. Define, *maximal information compression index* as

IEEE COMPUTER SOCIETY

$\lambda_2(x,y) =$ smallest eigenvalue of Σ [5], i.e.,

$$2\lambda_2(x,y) = (\text{var}(x) + \text{var}(y) - \sqrt{(\text{var}(x) + \text{var}(y))^2 - 4\text{var}(x)\text{var}(y)(1-\rho(x,y)^2)}. \qquad (1)$$

The value of λ_2 is zero when the features are linearly dependent and increases as the amount of dependency decreases. It may be noted that the measure λ_2 is nothing but the eigenvalue for the direction normal to the principle component direction of feature pair (x, y). It is shown in [2] that maximum information compression is achieved if a multivariate (in this case bivariate) data is projected along its principal component direction. The corresponding loss of information in reconstruction of the pattern (in terms of second order statistics) is equal to the eigenvalue along the direction normal to the principal component. Hence, λ_2 is the amount of reconstruction error committed if the data is projected to a reduced (in this case reduced from two to one) dimension in the best possible way. Therefore it is a measure of the *minimum amount of information loss* or the *maximum amount of information compression*, possible.

The significance of λ_2 can also be explained geometrically in terms of linear regression. It can be easily shown [6] that the value of λ_2 is equal to the sum of the squares of the perpendicular distances of the points (x, y) to the best fit line $y = \hat{a} + \hat{b}x$, obtained by minimizing the sum of the squared perpendicular distances. The coefficients of such a best fit line are given by $\hat{a} = \bar{x}cot\theta + \bar{y}$ and $\hat{b} = -cot\theta$, where $\theta = 2\tan^{-1}\left(\frac{2\text{cov}(x,y)}{\text{var}(x)^2 - \text{var}(y)^2}\right)$.

3. Feature Selection and Gene Clustering Method

The feature selection algorithm involves two steps, namely, partitioning the original set of genes/features into a number of homogeneous subsets (clusters) and selecting a representative gene from each such cluster [5]. Partitioning of the genes is done based on the k nearest neighbor (k-NN) principle using maximal information compression index as the feature similarity measure. In doing so, we first compute the k nearest features of each feature. Among them the feature having the most compact subset (as determined by its distance to the farthest neighbor) is selected, and its k neighboring features are assigned to its cluster. The process is repeated for the remaining features until all of them are either selected or assigned to a cluster.

While determining the k nearest neighbors of features we a assign a constant error threshold (ϵ) which is set equal to the distance of the kth nearest neighbor of the feature selected in the first iteration. In subsequent iterations, we check the λ_2 value, corresponding to the subset of a feature, whether it is greater than ϵ or not. If yes, then we decrease the value of k. Therefore k may be varying over iterations. The algorithm may be stated as follows:

Let the original number of genes/features be D, and the original feature set be $O = \{F_i, i = 1, \ldots, D\}$. Represent the similarity between features F_i and F_j by $S(F_i, F_j)$. Higher the value of S is, more dissimilar are the features. The maximal information compression index (Equation 2) may be used for S. Let r_i^k be the similarity between feature F_i and its kth nearest neighbor feature in R. Then

```
Step 1: Choose an initial value of $k \leq D - 1$. Initialize the selected feature subset $R$ to the original feature set $O$, i.e., $R \leftarrow O$.
Step 2: For each feature $F_i \in R$, compute $r_i^k$.
Step 3: Find feature $F_{i\prime}$ for which $r_{i\prime}^k$ is minimum. Retain this feature in $R$ and assign to its cluster $k$ nearest features of $F_{i'}$. (Note: $F_{i'}$ denotes the feature for which removing $k$ nearest neighbors will cause minimum error among all the features in $R$). Let $\epsilon = r_{i'}^k$.
Step 4: If $k >$ cardinality$(R) - 1$:  $k =$ cardinality$(R) - 1$.
Step 5: If $k = 1$: Go to Step 8.
Step 6:  While $r_{i'}^k > \epsilon$ do:
            (a) $k = k - 1$.
                $r_{i'}^k = \inf_{F_i \in R} r_i^k$.
            (b) If $k = 1$: Go to Step 8.
         End While
Step 7: Go to Step 2.
Step 8: Return the clusters corresponding to each of the features in $R$, and the feature set $R$ as the reduced feature set.
```

Computational complexity: The algorithm has low computational complexity. With respect to the dimension (or, number of genes) (D) the method has complexity $\mathcal{O}(D^2)$. Among the existing search based feature selection schemes only sequential forward and backward search have complexity $\mathcal{O}(D^2)$, though each feature subset evaluation is more time consuming. Other algorithms like plus-l-take-r, sequential floating search and branch and bound algorithm have complexity higher than quadratic.

Notion of scale in clustering/feature selection and choice of k: In our algorithm k controls the size of the reduced set. Since k determines the error threshold (ϵ), the representation of the data at different degrees of details is controlled by its choice. This characterstic is useful in data mining where *multiscale* representation of the data is often necessary. Note that the said property may not always be possessed by other algorithms where the input is usually the desired size of the reduced feature set. The reason is that changing the size of the reduced set may not necessarily result in any change in the levels of details. In contrast, for the proposed algorithm, k acts as a scale parameter which controls the degree of details in a more direct manner. An interesting fact observed in all the datasets considered is that, for high values of k the size of the selected subset varies linearly with k. Further, it is seen in those cases, $d + k \approx D$, where d is the size of the reduced subset and D is the size of the original feature set.

Non-metric nature of similarity measure: The similarity measures used in the above algorithm need not be a metric. Unlike conventional agglomerative clustering algorithms it does not utilise metric property of the similarity measures. Also unlike other clustering method used previously for feature selection, our clustering algorithm is partitional and non-hierarchical in nature.

The nature of both the clustering algorithm and the maximal information compression index is geared towards two goals - minimising the information loss (in terms of second order statistics) incurred in the process of feature reduction, and minimising the redundancy present in the reduced feature subset.

4. Experimental Results

Experimental results are presented for a benchmark microarray classification problem. The data [4] is a collection of 72 samples from leukemia patients, with each sample giving the expression levels of 7130 genes. According to pathological criteria, these samples include 47 type I Leukemias (called acute lymphoblastic leukemias, ALL) and 25 type II Leukemias (called acute myeloid leukemias, AML). The task is to design a classifier for these two classes based on the expression patterns. The samples are split into two sets with 38 samples serving as training set and remaining 34 as a test set. The data was used as a contest data in Critical Assessment of Techniques for Microarray Data Analysis (CAMDA'00) conference.

4.1. Classification accuracy

Training set and test set accuracies using three different classifiers, namely, k-nearest neighbor (k-NN, $k = 3$), naive Bayes with Gaussian distribution, and a linear support vector machine, are reported. The classifiers are trained using reduced sets of 50 features (from original 7130) obtained using (a) the proposed method, (b) Markov blanket (M-B) based method [7]. The comparative results are presented in Table 1 along with the computational time taken for feature selection on a Sun UltraSparc 350MHz workstation. The accuracy of the informative gene class predictor used in [4] is also provided in Table 1 for convenience. The reduced feature set is taken to be of size 50, as it was observed in [4] that 50 genes most closely correlated with AML-ALL distinction in the known samples.

Algorithm	time	k-NN		naive Bayes		SVM	
	(sec)	Train	Test	Train	Test	Train	Test
Proposed	1079	100%	94.5%	100%	93.5%	94%	87%
M-B [7]	3074	100%	93%	100%	91%	91%	82%
Class predictor of [4]		Test					
		85%					

Table 1. Classification performances

It can be seen from Table 1 that the proposed method provides a higher classification accuracy compared to the other two methods. It was found that the amount of overlap with the informative genes identified in [4] is higher for our method as compared to the Markov blanket based approach. While our method generates 31 common genes with the informative set, the Markov blanket based method obtains 21 common genes. The computation time of our method is also substantially lower.

4.2. Gene clustering performance

Gene clustering performance is quantitatively evaluated using an index called *representation entropy* [2], which measures the information compression obtained in the gene clustering process. The index is defined below.

Let the eigenvalues of the $d \times d$ covariance matrix of a feature/gene set of size d be $\lambda_j, j = 1, \ldots, d$. Let $\tilde{\lambda}_j = \frac{\lambda_j}{\sum_{j=1}^{d} \lambda_j}$. $\tilde{\lambda}_j$ has similar properties like probability, namely, $0 \leq \tilde{\lambda}_j \leq 1$ and $\sum_{j=1}^{d} \lambda_j = 1$. Hence, an entropy function can be defined as

$$H_R = -\sum_{j=1}^{d} \tilde{\lambda}_j \log \tilde{\lambda}_j. \tag{2}$$

The function H_R attains a minimum value (zero) when all the eigenvalues except one are zero, or in other words when all the information is present along a single coordinate direction. If all the eigenvalues are equal, i.e., information is equally distributed among all the features, H_R is maximum and so is the uncertainty involved in feature reduction.

The above representation entropy index is a property of the *dataset as represented by a particular set of features*, and is a measure of the amount of information compression possible by dimensionality reduction. This is equivalent to the amount of redundancy present in that particular representation of the dataset. Since the proposed algorithm involves partitioning of the original feature set into a number of homogeneous (highly compressible) clusters, it is expected that representation entropy of the genes in individual clusters are as low as possible, while that of the final reduced set of features/genes has low redundancy i.e., a high value of representation entropy.

Algorithm	Avg. gene cluster entropy (H_R^g)	Selected gene set entropy (H_R^s)
Proposed	0.14	0.84
k-medoid	0.27	0.72

Table 2. Representation entropy of gene clusters

The representation entropies of the clustering obtained by the proposed method is compared with those obtained by a method described in [3], which uses correlation between two genes as their similarity measure and the k-medoid algorithm to cluster the genes based on the similarity measure. The number of gene clusters were specified to be 50. Let us denote the value of H_R computed for the genes in a single cluster by H_R^g and the value of H_R for the final selected genes/features by H_R^s. Average value of H_R^g computed over all the clusters are reported in Table 2. It is observed from Table 2 that the gene/feature subset selected by the proposed scheme is less redundant, and the genes within a cluster have higher homogenity among them signifying minimum information loss in the process of feature selection.

5. Conclusion and Discussions

An algorithm for gene clustering and feature selection using a novel feature similarity measure and a multiscale k nearest neighbor based clustering algorithm is described.

Unlike other feature selection approaches which are based on optimizing classification performance explicitly, here we determine a set of maximally independent features by discarding the redundant ones. This enhances the applicability of the resulting features to modelling and visualization in addition to classificatory analysis.

We have used linear dependency as gene dissimalirity measures. More sophisticated techniques like independent component analysis may be used for computing the dissimilarity. Use of the clustering algorithm presented here along with such measures will provide better results.

References

[1] P. Baldi and G. W. Hatfield. *DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling*. Cambridge University Press, Cambridge, UK, 2002.

[2] P. A. Devijver and J. Kittler. *Pattern Recognition, A Statistical Approach*. Prentice–Hall, Inc., London, 1982.

[3] M. Eisen, P. Spellman, P. Brown, and D. Botstein. Cluster analysis and display of genome wide expression data. *Proc. Natl. Acad. Sci.*, 95:14863–14868, 1998.

[4] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

[5] P. Mitra, C. A. Murthy, and S. K. Pal. Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(3):301–312, 2002.

[6] C. R. Rao. *Linear Statistical Inference and its Applications*. John Wiley, 1973.

[7] E. P. Xing, M. I. Jordan, and R. M. Karp. Feature selection for high-dimensional genomic microarray data. In *Proc. 18th International Conf. on Machine Learning*, pages 601–608. Morgan Kaufmann, San Francisco, CA, 2001.