JCSE Journal of Computing Science and Engineering

Special Issue
Journal of Computing Science and Engineering,
Vol. 11, No. 4, December 2017, pp. 121-129

Feature Selection via Embedded Learning Based on Tangent Space Alignment for Microarray Data

Xiucai Ye* **and Tetsuya Sakurai**
Department of Computer Science, University of Tsukuba, Tsukuba, Japan
yexiucai@mma.cs.tsukuba.ac.jp, sakurai@cs.tsukuba.ac.jp

Abstract
Feature selection has been widely established as an efficient technique for microarray data analysis. Feature selection aims to search for the most important feature/gene subset of a given dataset according to its relevance to the current target. Unsupervised feature selection is considered to be challenging due to the lack of label information. In this paper, we propose a novel method for unsupervised feature selection, which incorporates embedded learning and $l_{2,1}$-norm sparse regression into a framework to select genes in microarray data analysis. Local tangent space alignment is applied during embedded learning to preserve the local data structure. The $l_{2,1}$-norm sparse regression acts as a constraint to aid in learning the gene weights correlatively, by which the proposed method optimizes for selecting the informative genes which better capture the interesting natural classes of samples. We provide an effective algorithm to solve the optimization problem in our method. Finally, to validate the efficacy of the proposed method, we evaluate the proposed method on real microarray gene expression datasets. The experimental results demonstrate that the proposed method obtains quite promising performance.

Category: Bioinformatics

Keywords: Unsupervised feature selection; Embedded learning; Sparse regression; Tangent space alignment; Microarray gene expression data

I. INTRODUCTION

In bioinformatics, many large projects together with new techniques, such as DNA microarray techniques, have created an enormous amount of data. DNA microarray techniques enable biologists to simultaneously measure the expression level of thousands of genes in specific samples at a given time and under certain conditions [1]. Microarray data often comes with high dimensionality, which involves several thousands of genes, far exceeding the limited number of samples. An important research topic in microarray data is the discovery of genes which are relevant to a particular target annotation. Usually, only a small number of genes show a strong correlation, while a large number of genes are irrelevant and act like noise to decrease the performance [2]. Furthermore, the high data dimensionality can significantly increase the computational burden, and can even make some data mining approaches impossible [3]. Thus, finding the small set of informative genes is of paramount importance to microarray data analysis.

Feature selection is one of the most important computational techniques in processing the analysis of microarray data. Feature selection aims at searching for the most

Open Access http://dx.doi.org/10.5626/JCSE.2017.11.4.121 http://jcse.kiise.org

Received 25 May 2017; **Accepted** 25 November 2017
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

discriminant feature/gene subset to distinguish dfferent classes. Based on the way of utilizing label information, feature selection can be classified into supervised and unsupervised methods [4]. Unsupervised feature selection is more challenging than supervised feature selection, since without label information the relevance of features is unclear [5]. With the rapid accumulation of high-dimensional data, the obtained data usually lack any label information [6]. Thus, it is of great importance to develop unsupervised methods for the unlabeled data.

Unsupervised feature selection has attracted much attention in recent years and a large number of unsupervised feature selection methods have been proposed [7, 8]. The filter and embedded methods are two kinds of widely used unsupervised feature selection methods. In the filter methods, the relevance of features is assessed by looking only at the intrinsic properties of data. In most cases, a relevance score of each feature is calculated separately, and the features with low scores are removed. Typical filter methods include the max variance (MaxVar) method, the Laplacian score (LapScore) method [9] and the spectral feature selection (SPEC) method [10]. The filter methods are usually computationally simple and fast. However, a common disadvantage of the filter methods is that they ignore the feature dependencies, which may lead to poorer clustering or classification performance.

In contrast to the unsupervised filter methods, unsupervised embedded methods have been developed to search for an optimal subset of features by considering the correlation of features with a learning model simultaneously. A number of methods have been proposed to maintain the important underlying data structure in the embedded learning processes [11, 12]. The importance of preserving local structure has been well recognized in the recent development of unsupervised feature selection methods. Cai et al. [13] explored manifold learning and l_1 regularization to select the features that can best preserve the multi-cluster structure. Hou et al. [14] used similarity based on locally linear approximation to construct graph, and used unified embedded learning and sparse regression to perform feature selection. Li et al. [15] utilized spectral clustering and a nonnegative constraint for feature selection by learning the cluster labels.

In this paper, we propose a novel unsupervised feature selection method, which utilizes embedded learning and $l_{2,1}$-norm sparse regression into a framework to select genes in microarray gene expression data. We apply local tangent space alignment to preserve the local data structure during embedded learning. The tangent space at each data sample provides a low-dimensional linear approximation of the local geometric structure of the nonlinear manifold. The $l_{2,1}$-norm sparse regression in the model acts as a constraint to learn the gene weights correlatively. The resultant formulation of the proposed method optimizes for selecting the most discriminative features that can better capture the underlying data structure, and thus to select the most informative genes that can better capture the interesting natural clusters of samples. We develop an iterative algorithm to effectively solve the optimization problem in the proposed method. Experimental results on real microarray gene expression datasets demonstrate the effectiveness of the proposed method.

The rest of this paper is organized as follows. Some preliminaries are presented in Section II. We present the proposed unsupervised feature selection method in Section III. The experimental results are shown in Section IV. Finally, we conclude the paper in Section V.

II. PRELIMINARIES

A. Notations

In a gene expression microarray study, the output of the microarray experiment is recorded as a gene expression data matrix X. Let $X = [x_1, x_2, \ldots, x_n]^T \in \mathbb{R}^{n \times m}$, where $x_i \in \mathbb{R}^m$ ($i = 1, \ldots, n$) denote the n unlabeled samples. For the sake of convenience, we use $g_1, g_2, ..., g_m$ to denote the m genes. The data matrix can also be denoted as $X = [g_1, g_2, ..., g_m]$. We would like to select the most informative d ($d < m$) genes to represent the original sample. For a matrix $A = (a_{ij}) \in \mathbb{R}^{u \times v}$, its $l_{2,1}$-norm is defined as

$$\|A\|_{2,1} = \sum_{i=1}^{u} \sqrt{\sum_{j=1}^{v} a_{i,j}^2}. \quad (1)$$

The Frobenius norm of A is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{u} \sum_{j=1}^{v} a_{i,j}^2}. \quad (2)$$

B. Local Tangent Space Alignment

The basic idea of local tangent space alignment is to use the tangent space in the neighborhood of each data point to represent the local geometry, and then align those local tangent spaces to construct the global coordinate system for the nonlinear manifold [16].

Firstly, the k nearest neighbors of x_i ($i = 1, , n$) are found and denoted as a set $N_k(x_i) = [x_{i_1}, x_{i_2}, ..., x_{i_k}]$. Secondly, the k data points in $N_k(x_i)$ are projected into the tangent space of the manifold at x_i by

$$v_j^{(i)} = P_i^T (x_{i_j} - m_i), \quad j = 1, ..., k, \quad (3)$$

where $v^{(i)}$ is a local coordinate of x_i, $m_i = \frac{1}{k}\sum_{j=1}^{k} x_{i_j}$ is a mean vector, and $P_i \in \mathbb{R}^{m \times q}$ is a tangent space projection matrix. P_i can be calculated by performing the optimal rank approximation of the centered data matrix [16]. Then, the local coordinates are linearly aligned into a single global coordinate system in $\mathbb{R}^q$ by

$$y_{i_j} = C_i v_j^{(i)} + t_i, \quad j = 1, ..., k, \quad (4)$$

where $C_i \in \mathbb{R}^{q \times q}$ is an affine transformation matrix, and t_i is a translation vector in $\mathbb{R}^q$. The optimal C_i and t_i in Eq. (4) can be computed as [11, 16]

$$C_i = Y_i^T\left(I_k - \frac{1}{k}e_k e_k^T\right)V_i^+,$$

$$t_i = \frac{1}{k}\sum_{j=1}^{k} y_{i_j}, \tag{5}$$

where $Y_i = [y_{i_1}, y_{i_2}, ..., y_{i_k}]^T \in \mathbb{R}^{k \times q}$, $V_i = [v_1^{(i)}, v_2^{(i)}, ..., v_k^{(i)}] \in \mathbb{R}^{q \times k}$, I_k is a $k \times k$ identity matrix, $e_k = [1, 1, \ldots, 1] \in \mathbb{R}^k$, and V_i^+ is the Moore-Penrose generalized inverse of V_i. The k squared errors from $N_k(x_i)$ are calculated as

$$\varepsilon_j^{(i)} = \left\| y_{i_j} - C_i v_j^{(i)} - t_i \right\|_2^2, \quad j = 1, ..., k. \tag{6}$$

Summing all costs from the n patches obtains

$$\varepsilon(Y) = \sum_{i=1}^{n}\sum_{j=1}^{k}\varepsilon_j^{(i)} = \sum_{i=1}^{n} tr(Y_i^T B_i Y_i), \tag{7}$$

where $B_i = \left(I_k - \frac{1}{k}e_k e_k^T\right)(I_k - V_i^+ V_i)\left(I_k - \frac{1}{k}e_k e_k^T\right)$.

According to the row selection matrices $S_i \in \mathbb{R}^{k \times n}$ (i = 1, …, n), we know that $Y_i = S_i Y$. Thus, Eq. (7) can be rewritten as

$$\varepsilon(Y) = \sum_{i=1}^{n} tr(Y^T S_i^T B_i S_i Y). \tag{8}$$

The objective of local tangent space alignment is to minimize $\varepsilon(Y)$ as

$$\min_{Y^T Y = I_q} tr(Y^T L Y), \tag{9}$$

where

$$L = \sum_{i=1}^{n} S_i^T B_i S_i. \tag{10}$$

III. THE PROPOSED METHOD

In this section, we propose a novel unsupervised feature selection method for microarray data analysis. The proposed method utilizes local Tangent Space Alignment in embedded learning for unsupervised Feature Selection. Thus, we refer to it as the TSAFS method.

A. Formulations

In the proposed method, the original sample x_i is embedded in a low-dimensional space by a transformation matrix $W \in \mathbb{R}^{m \times q}$. We use $Y = [y_1, y_2, ..., y_n]$ to denote the embedded data matrix of X. We utilize local tangent space alignment during the process of embedded learning to preserve the local data structure.

The objective function of the proposed TSAFS method is formulated as

$$\min_{W, Y^T Y = I_q} \|XW - Y\|_F^2 + \alpha tr(Y^T L Y) + \beta \|W\|_{2,1}, \tag{11}$$

where α and β are two balanced parameters. $tr(Y^T LY)$ is calculated as in Eq. (9), which is a promoting regularization term to preserve the local data structure by local tangent space alignment.

The third term in Eq. (11) is the $l_{2,1}$-norm of the transformation matrix W to promote row sparsity. Denote w_i as the i^{th} row of W, i.e., $W = [w_1, ..., w_m]^T \in \mathbb{R}^{m \times q}$. Since the i^{th} row w_i corresponds to the weight of gene g_i, the sparsity constraint on rows makes W suitable for gene selection. Each gene is ranked according to $\|w_i\|_2$ in descending order and the top d genes are selected.

B. Solutions

The optimization problem in Eq. (11) is not convex when both W and Y are optimized simultaneously.

Furthermore, the $l_{2,1}$-norm of W makes the problem non-smooth. Inspired by [8] and [14], we solve this problem in an alternative way.

For the sake of convenience, we denote $\Theta(W) = \|w\|_{2,1}$. The derivative of Θ(W) with respect to W is

$$\frac{\partial \Theta(W)}{\partial W} = 2UW, \tag{12}$$

where $U \in \mathbb{R}^{m \times m}$ is a diagonal matrix with the i^{th} diagonal element as

$$U_{ii} = \frac{1}{2\|w_i\|_2}. \tag{13}$$

Then, we construct an auxiliary function to solve the problem in Eq. (11).

$$\Gamma(W, U, Y) = \min_{W, Y^T Y = I_q} \|XW - Y\|_F^2 + \alpha tr(Y^T LY) + \beta tr(W^T UW). \tag{14}$$

Note that when U is fixed, the derivative in Eq. (11) can also be regarded as the derivative of Eq. (14). Thus, we try to solve the problem in Eq. (14) to approximate the solution in Eq. (11).

The derivative of $\Gamma(W, U, Y)$ with respect to W is

$$\frac{\partial \Gamma(W, U, Y)}{\partial W} = 2X^T XW - 2X^T Y + 2\beta UW. \tag{15}$$

By setting $\frac{\partial \Gamma(W, U, Y)}{\partial W} = 0$, we can obtain

$$W = (X^T X + \beta U)^{-1} X^T Y. \tag{16}$$

By substituting the above W into Eq. (14), we have

$$\begin{aligned}&\Gamma(W, U, Y)\\ &= \min_{W, Y^T Y=I_q} tr(W^T X^T XW) - 2tr(Y^T XW)\\ &\quad + tr(Y^T Y) + \alpha tr(Y^T LY) + \beta tr(W^T UW)\\ &= \min_{W, Y^T Y=I_q} tr(Y^T Y) + \alpha tr(Y^T LY)\\ &\quad - tr(W^T (X^T X + \beta U)W). \end{aligned} \tag{17}$$

Let $M = X^T X + \beta U$, Eq. (17) becomes

$$\begin{aligned}&\Gamma(W, U, Y)\\ &= \min_{W, Y^T Y=I_q} tr(Y^T Y) + \alpha tr(Y^T LY)\\ &\quad - tr(T^T XM^{-1} X^T Y)\\ &= \min_{W, Y^T Y=I_q} tr(Y^T (I_q + L - XM^{-1} X^T)Y). \end{aligned} \tag{18}$$

When M and L are fixed, the solution of Eq. (18) can be obtained by solving the following eigenproblem.

$$(I_q + L - XM^{-1} X^T) y_i = \lambda y_i . \tag{19}$$

The solution of Eq. (19) is the matrix Y which contains the eigenvectors corresponding to the q smallest eigenvalues as the column vectors.

In summary, we solve the optimization problem in Eq. (11) in an alternative way. When W is fixed, U can be updated according to Eq. (13). When U is fixed, Y can be updated according to Eq. (19). Then, W can be updated according to Eq. (16). After that, U can be updated again based on the new W. The updating process will be repeated until the objective function converges. We summarize the procedure of the proposed TSAFS method in Algorithm 1. Algorithm 1 will terminate when the objective function of Eq. (11) tends to a constant or the change is smaller than a threshold. The threshold is set very close to zero.

Algorithm 1 The proposed TSAFS method

Input: Gene expression data matrix $X \in \mathbb{R}^{n \times m}$; Parameters α, β, k, q; Number of features to select d;
Output: d selected features;
1: Construct the k-nearest neighbor graph;
2: Calculate L as in equation (10);
3: Initialize $U_0 = I_q$ and set $t = 0$;
4: **repeat**
5: Calculate $M_t = X^T X + \beta U_t$;
6: Calculate Y_t by solving the eigenproblem in equation (19);
7: Calculate W_t according to equation (16);
8: Calculate U_{t+1} according to equation (13);
9: $t = t + 1$;
10: **until** convergence
11: Sort each gene g_i according to $\|w_i\|_2$ in descending order and select the top d ranked ones.

To optimize the objective function of TSAFS, the most time consuming operation is to solve the generalized eigenproblem in Eq. (19), which has a time complexity of $O(m^3)$, where m is the number of features/genes.

C. Convergence Analysis

We show that Algorithm 1 will monotonically decrease the value of the objection function of Eq. (11) in each iteration. In Algorithm 1, when U is fixed as U_t in the t^{th} iteration to calculate W_{t+1} and Y_{t+1}, the following inequality holds:

$$\begin{aligned}&\left\|XW^{t+1} - Y^{t+1}\right\|_F^2 + \alpha tr((Y^{t+1})^T LY^{t+1})\\ &+ \beta tr((W^{t+1})^T U^t W^{t+1})\\ &\le \left\|XW^t - Y^t\right\|_F^2 + \alpha tr((Y^t)^T LY^t)\\ &+ \beta tr((W^t)^T U^t W^t). \end{aligned} \tag{20}$$

Since $\|W\|_{2,1} = \sum_{i=1}^m \|w_i\|_2$, we obtain

$$\begin{aligned}&\left\|XW^{t+1} - Y^{t+1}\right\|_F^2 + \alpha tr((Y^{t+1})^T LY^{t+1})\\ &+ \beta \left\|W^{t+1}\right\|_{2,1} + \beta \sum_{i=1}^m \left(\frac{\left\|w_i^{t+1}\right\|_2^2}{2\left\|w_i^t\right\|_2} - \left\|w_i^{t+1}\right\|_2 \right)\\ &\le \left\|XW^t - Y^t\right\|_F^2 + \alpha tr((Y^t)^T LY^t)\\ &+ \beta \left\|W^t\right\|_{2,1} + \beta \sum_{i=1}^m \left(\frac{\left\|w_i^t\right\|_2^2}{2\left\|w_i^t\right\|_2} - \left\|w_i^t\right\|_2 \right). \end{aligned} \tag{21}$$

According to a Lemma in [17], we know

$$\frac{\left\|w_i^{t+1}\right\|_2^2}{2\left\|w_i^t\right\|_2} - \left\|w_i^{t+1}\right\|_2 \ge \frac{\left\|w_i^t\right\|_2^2}{2\left\|w_i^t\right\|_2} - \left\|w_i^t\right\|_2 . \tag{22}$$

Combining Eqs. (21) and (22), we have

$$\begin{aligned}&\left\|XW^{t+1} - Y^{t+1}\right\|_F^2 + \alpha tr((Y^{t+1})^T LY^{t+1}) + \beta \left\|W^{t+1}\right\|_{2,1}\\ &\le \left\|XW^t - Y^t\right\|_F^2 + \alpha tr((Y^t)^T LY^t) + \beta \left\|W^t\right\|_{2,1}. \end{aligned} \tag{23}$$

This inequality indicates that the objective function of Eq. (11) will monotonically decrease in each iteration. Since the objective function has lower bounds, such as zero, the above iteration will converge. Empirical results show that the convergence is fast and only several iterations (fewer than 10 iterations in the experiments) are needed for convergence to occur. Thus, the proposed method scales well in practice.

IV. EXPERIMENTS

In this section, we test the performance of the proposed TSAFS method on microarray gene expression datasets. We test the performance in terms of clustering and

classification, i.e., using the *K*-means clustering and the Nearest Neighbors (NN) classifier. We compare the proposed method with several existing feature selection methods, i.e., LapScore [9], MCFS [13], JELSR [14] and NDFS [15]. We also compare these feature selection methods with the baseline method which uses all the features for clustering and classification. In the experiments, the number of selected genes is ranged over {20, 40, 60, 80, 100, 120, 140, 160, 180, 200}. The parameters are tuned over $\{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 1, 10^{2}, 10^{4}, 10^{6}, 10^{8}\}$. The number of nearest neighbors is set as $k = 5$. We report the best result of all the methods by using different parameters.

A. Data Set Description

In the experiments, we use four public gene expression datasets to illustrate the performance of different feature selection methods. The datasets are Lung, Glioma, Lymphoma and ALLAML, which were downloaded from http://featureselection.asu.edu/datasets.php. The properties of the datasets are summarize in Table 1 and briefly introduced as follows.

- Lung: 203 samples contain an expression level of 12,600 genes. The samples consist of 5 clusters with 139, 21, 20, 6, and 17 samples, respectively. The genes with standard deviations smaller than 50 expression units are removed and 3,312 genes are retained for the 203 samples.
- Glioma: 50 different samples are taken from 4 clusters with 14, 7, 14, and 15 samples, respectively. The samples contain an expression level of 4,434 genes.
- Lymphoma: 4,026 genes are taken from 96 different samples. The samples consist of 9 clusters with 46, 11, 10, 9, 6, 6, 4, 2, and 2 samples, respectively.
- ALLAML: 7,129 genes are collected from 72 samples, which belong to patients suffering from acute myeloid leukemia (AML: 25 samples) and acute lymphoblastic leukemia (ALL: 47 samples).

Table 1. Properties of datasets

Dataset	# of samples	# of genes	# of clusters
Lung	203	3,312	5
Glioma	50	4,434	4
Lymphoma	96	4,026	9
ALLAML	72	7,129	2

B. Clustering Results

In the first group experiment, we apply the *K*-means clustering method to evaluate the performance of the proposed method. Two widely used evaluation metrics, i.e., normalized mutual information (NMI) and accuracy (ACC), are applied to evaluate the clustering results. Denote $H = \{H_i\}_{i=1}^{h}$ as the ground truth clustering configuration of a dataset, where h is the ground truth cluster number. Denote $H' = \{H_i'\}_{i=1}^{h'}$ as the clustering configuration obtained by a clustering algorithm, where h' is the obtained cluster number. n is the cardinality of the whole dataset. n_i is the cardinality of H_i. n_i' is the cardinality of H_i'. And, n_{ij} is the cardinality of the intersection of H_i and H_i'. The NMI criteria are defined as

$$NMI(H, H') = \frac{\sum_{i=1}^{h}\sum_{j=1}^{h'} n_{ij}\log(n \cdot n_{ij}/(n_i \cdot n_j'))}{\sqrt{(\sum_{i=1}^{h} n_i \log(n_i/n))(\sum_{j=1}^{h'} n_j' \log(n_j'/n))}}. \quad (24)$$

A larger value of NMI denotes better performance. Let l_i denote the ground truth label of x_i and l_i' denote the index of the clustering result of x_i. ACC is defined as [12]

$$ACC(H, H') = \frac{\sum_{i=1}^{h} \delta(l_i, map(l_i'))}{n}, \quad (25)$$

where $\delta(a, b) = 1$ if $a = b$ and $\delta(a, b) = 0$ otherwise, and $map(l_i')$ is the best mapping function that permutes clustering labels to match the ground truth labels using the Kuhn-Munkres algorithm. A larger value of ACC denotes a better clustering result.

Each feature selection method is first performed to select genes on the gene expression datasets. After selecting the genes, *K*-means clustering is performed by using only the selected genes. We repeat each experiment 20 times with random initializations and report the mean performance with standard deviation.

First, we compare the performance of the feature selection methods on the four gene expression datasets. The experimental results in terms of NMI and ACC evaluation metrics are shown in Tables 2 and 3, respectively. We can

Table 2. Normalized mutual information (%) of different feature selection methods

Dataset	Baseline	LapScore	MCFS	JELSR	NDFS	TSAFS
Lung	48.07 ± 3.58	52.73 ± 4.93	58.77 ± 5.24	60.83 ± 4.18	60.78 ± 4.12	62.62 ± 4.61
Glioma	41.06 ± 2.13	48.54 ± 2.68	47.02 ± 3.13	49.83 ± 3.25	50.62 ± 2.88	53.03 ± 3.45
Lymphoma	60.41 ± 4.20	65.29 ± 2.72	65.02 ± 2.93	59.66 ± 3.59	60.82 ± 3.81	66.12 ± 3.94
ALLAML	9.83 ± 4.43	13.59 ± 4.17	13.20 ± 5.01	10.67 ± 4.50	12.02 ± 5.33	20.94 ± 4.15

Values are presented as mean ± standard deviation.

Table 3. Accuracy (%) of different feature selection methods

Dataset	Baseline	LapScore	MCFS	JELSR	NDFS	TSAFS
Lung	67.00 ± 1.64	60.37 ± 1.89	81.28 ± 3.24	78.74 ± 4.17	77.49 ± 4.13	82.76 ± 2.83
Glioma	52.00 ± 3.12	60.12 ± 3.24	60.40 ± 2.73	59.45 ± 3.41	58.38 ± 3.28	61.10 ± 3.26
Lymphoma	56.54 ± 5.26	62.50 ± 3.64	60.67 ± 3.84	58.33 ± 4.17	58.63 ± 4.50	65.63 ± 4.37
ALLAML	68.17 ± 6.48	73.65 ± 5.12	73.44 ± 5.82	70.54 ± 6.05	72.61 ± 5.71	81.94 ± 6.02

see that most of the unsupervised feature selection methods perform better than the baseline method. Gene selection can improve the accuracy of clustering results. The proposed TSAFS method performs better than the other methods on the four datasets, especially on the ALLAML dataset. This is because TSAFS utilizes local tangent space alignment for local structure preservation to learn the weight of genes.

Then, we evaluate the performance of the clustering results by varying the number of selected genes. The performance of the clustering results in term of NMI and ACC evaluation metrics are shown in Figs. 1 and 2, respectively. We can see from the figures that the proposed TSAFS method performs better than other methods in most cases when selecting a different number of genes. Note that for different datasets and different methods, the numbers of selected genes to obtain the best results are different. For example, in Fig. 1, on the Lung dataset, for TSAFS and JELSR the optimized gene number is about 140, while for LapScore and NDFS the optimized gene number is about 160.

This is because in different microarray datasets, the correlations of genes are different. Different methods provide different ways to explore the underlying data structure. In Fig. 2, the performance trend by using the ACC evaluation metric is very similar to that using the NMI evaluation metric. The proposed TSAFS method obtains better performance than other methods when both NMI and ACC evaluation metrics are applied.

C. Classification Results

In the second group, we apply Nearest Neighbors (NN) classifier to test the performance. We utilize 5-fold cross-

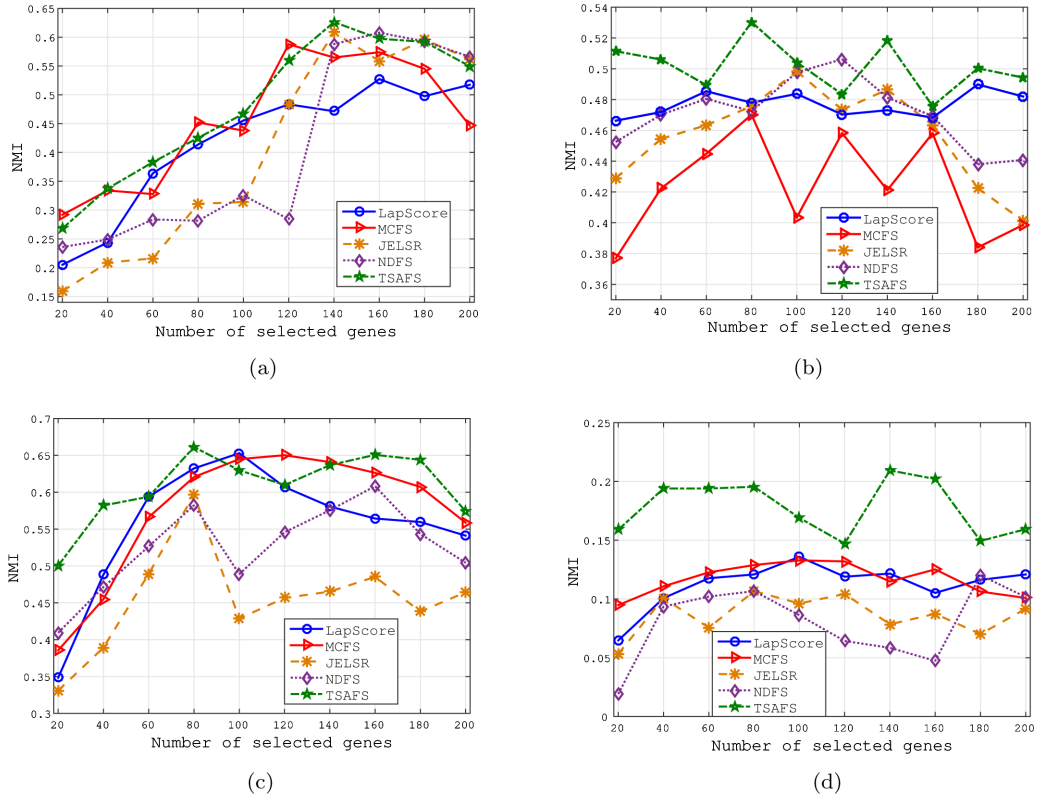


Fig. 1. Normalized mutual information (NMI) by varying the number of selected genes. (a) Lung, (b) Glioma, (c) Lymphoma, and (d) ALLAML.

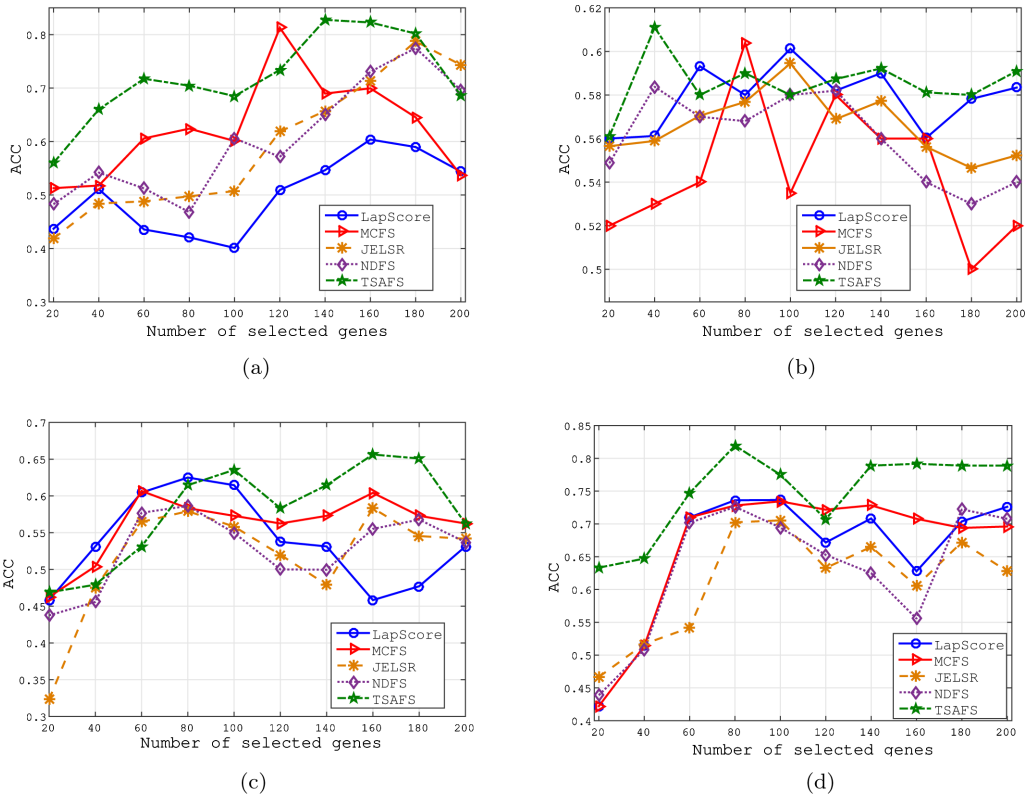


Fig. 2. Accuracy (ACC) by varying the number of selected genes. (a) Lung, (b) Glioma, (c) Lymphoma, and (d) ALLAML.

Table 4. Classification error (%) of different feature selection methods

Dataset	Baseline	LapScore	MCFS	JELSR	NDFS	TSAFS
Lung	5.93 ± 1.63	8.90 ± 1.73	6.84 ± 1.22	7.33 ± 2.04	8.37 ± 1.62	6.90 ± 2.10
Glioma	22.02 ± 3.00	22.20 ± 3.18	19.22 ± 2.81	20.12 ± 3.06	22.00 ± 2.73	19.03 ± 3.02
Lymphoma	16.67 ± 3.04	15.58 ± 2.33	15.51 ± 2.57	16.63 ± 2.93	16.14 ± 3.22	10.42 ± 3.11
ALLAML	18.68 ± 4.63	13.89 ± 3.52	13.59 ± 4.00	17.28 ± 3.66	15.28 ± 4.07	6.56 ± 3.98

validation by which the original samples are randomly partitioned into 5 equal-sized subsets. Of the 5 subsets, a single subset is retained as the validation data for testing the model, and the remaining 4 subsets are used as training data. The cross-validation process is then repeated 5 times (the folds), with each of the 5 subsets used exactly once as the validation data. We perform gene selection by using the training data, and evaluate the performance of the selected genes on the test data. The experiments are repeated 20 times on the best parameter combination. We report the mean classification error with standard deviation.

The classification results of different methods on the four datasets are listed in Table 4. We can see from the table that the proposed TSAFS method has a lower classification error than the other methods on most of the datasets, except for the Lung dataset. On the Lung dataset, the baseline method obtains the best result, and gene selection cannot reduce the classification error. On the other three datasets, most of the unsupervised feature selection methods perform better than the baseline method. In particular, on the ALLAML dataset, the proposed method outperforms other methods significantly. This is because, on the ALLAML dataset, by the proposed method the selected genes have less redundancy and have a higher accuracy in predictive results.

The detailed classification performances for the selected features are shown in Fig. 3. We can see from Fig. 3 that the proposed TSAFS method has a lower classification error than other methods on most of the selected features on the four datasets. The proposed TSAFS method also has a better stability than other methods on the four datasets. On three of the four datasets, i.e., on the Lung, Lymphoma and ALLAML datasets, the proposed method outperforms other methods significantly. In particular, on the Lung and ALLAML datasets, TSAFS has the best performance on all of the selected features.

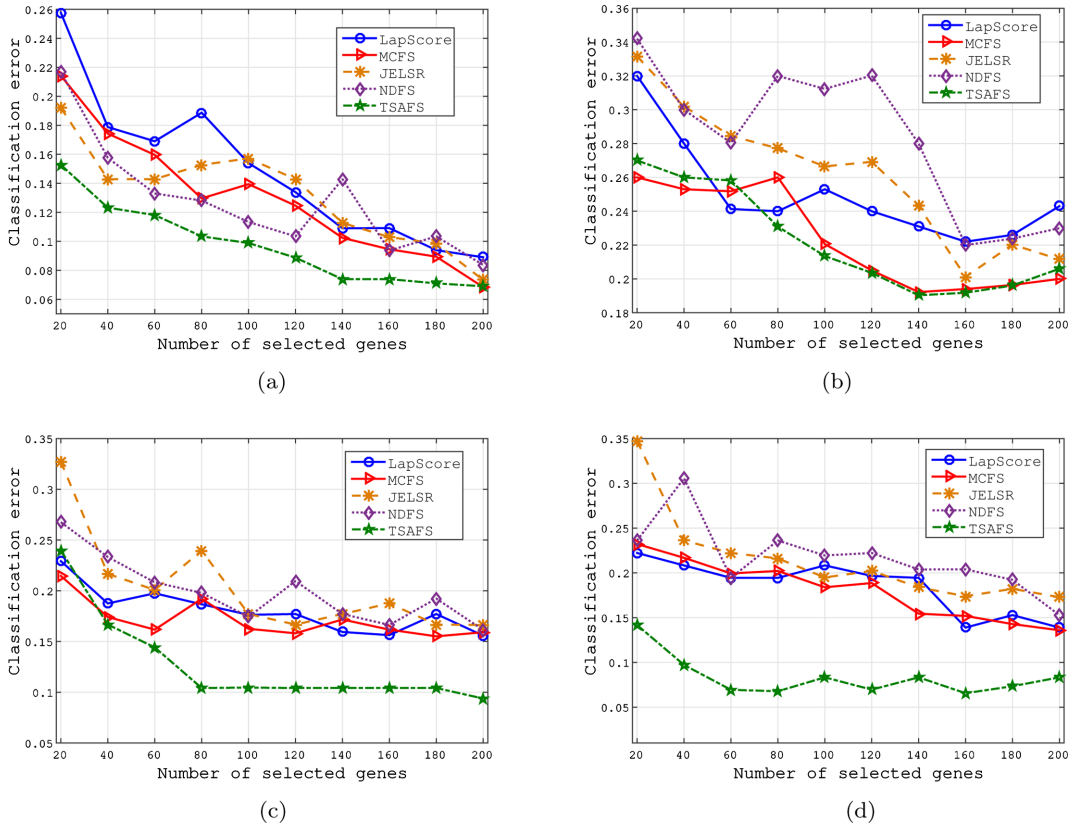


Fig. 3. Classification error by varying the number of selected genes. (a) Lung, (b) Glioma, (c) Lymphoma, and (d) ALLAML.

V. CONCLUSION

In this paper, we proposed a novel unsupervised feature selection method for gene selection in microarray data analysis. The proposed method incorporates embedded learning and $l_{2,1}$-norm sparse regression into a framework for feature learning. Local tangent space alignment is applied during embedded learning to preserve the local data structure. The proposed method optimizes for selecting the informative genes which better capture the interesting natural classes of samples. We provided an effective algorithm to solve the optimization problem in our method. Experiments on four real microarray gene expression datasets demonstrated that the proposed method not only achieves good performance, but also outperforms other state-of-the-art unsupervised feature selection methods. In future work, we will consider an efficient method to reduce the time complexity of the proposed algorithm to solve the optimization problem.

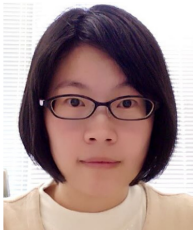
ACKNOWLEDGMENTS

This work was supported in part by JST/CREST and MEXT KAKENHI (Grant No. 25286097).

REFERENCES

1. E. De Rinaldis and A. Lahm, *DNA Microarrays: Current Applications*. Norfolk, UK: Horizon Scientific Press, 2007.
2. T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, et al., "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531-537, 1999.
3. R. L. Somorjai, B. Dolenko, and R. Baumgartner, "Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions," *Bioinformatics*, vol. 19, no. 12, pp. 1484-1491, 2003.
4. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, no. 1, pp. 389-422, 2002.
5. J. G. Dy and C. E. Brodley, "Visualization and interactive feature selection for unsupervised data," in *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, MA, 2000, pp. 360-364.
6. S. Zhang, H. S. Wong, Y. Shen, and D. Xie, "A new unsupervised feature ranking method for gene expression data based on consensus affinity," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 9, no. 4, pp. 1257-1263, 2012.

7. P. Zhu, W. Zuo, L. Zhang, Q. Hu, and S. C. Shiu, "Unsupervised feature selection by regularized self-representation," *Pattern Recognition*, vol. 48, no. 2, pp. 438-446, 2015.
8. X. Ye, K. Ji, and T. Sakurai, "Unsupervised feature selection with correlation and individuality analysis," *International Journal of Machine Learning and Computing*, vol. 6, no. 1, pp. 36-41, 2016.
9. X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Advances in Neural Information Processing Systems*, vol. 18, pp. 507-514, 2006.
10. Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proceedings of the 24th International Conference on Machine Learning*, Corvalis, OR, 2007, pp. 1151-1157.
11. X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1083-1095, 2014.
12. X. Ye, K. Ji, and T. Sakurai, "Global discriminant analysis for unsupervised feature selection with local structure preservation," in *Proceedings of the 29th International Florida Artificial Intelligence Research Society Conference*, Key Largo, FL, 2016, pp. 454-459.
13. D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, 2010, pp. 333-342.
14. C. Hou, F. Nie, D. Yi, and Y. Wu, "Feature selection via joint embedding learning and sparse regression," in *Proceedings of International Joint Conference on Artificial Intelligence*, Barcelona, Spain, 2011, pp. 1324-1329.
15. Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, Toronto, Canada, 2012, pp. 1026-1032.
16. Z. Zhang and H. Zha, "Nonlinear dimension reduction via local tangent space alignment," in *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning*, Hong Kong, 2003, pp. 477-481.
17. F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint $l_{2,1}$-norms minimization," *Advances in Neural Information Processing Systems*, vol. 23, pp. 1813-1821, 2010.



Xiucai Ye

Xiucai Ye received her Ph.D. in computer science from the University of Tsukuba, Tsukuba Science City, Japan, in 2014. She is currently working as a postdoctoral research fellow at the Department of Computer Science, University of Tsukuba, Tsukuba Science City, Japan. Her current research interests include clustering, feature selection, machine learning and its application fields.

Tetsuya Sakurai

Tetsuya Sakurai is a Professor of Department of Computer Science, and the Director of Center for Artificial Intelligence Research (C-AIR) at the University of Tsukuba. He is also a visiting professor at the Open University of Japan, and a visiting researcher of Advanced Institute of Computational Science at RIKEN. He received a Ph.D. degree in Computer Engineering from Nagoya University in 1992. His research interests include high performance algorithms for large-scale simulations, data and image analysis, and neural network computations. He is a member of the Japan Society for Industrial and Applied Mathematics (JSIAM), the Mathematical Society of Japan (MSJ), Information Processing Society of Japan (IPSJ), and Society for Industrial and Applied Mathematics (SIAM).