IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 4, 2018, accepted March 8, 2018, date of publication March 14, 2018, date of current version April 4, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2815606

Supervised Feature Selection With a Stratified Feature Weighting Method

RENJIE CHEN[1], NING SUN[2], XIAOJUN CHEN[3], MIN YANG[4], AND QINGYAO WU[1]

[1]School of Software Engineering, South China University of Technology, Guangzhou 510006, China
[2]Laboratory and Equipment Management Department, South University of Science and Technology, Shenzhen 518055, China
[3]College of Computer Science and Software, Shenzhen University, Shenzhen 518060, China
[4]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China

Corresponding authors: Xiaojun Chen (xjchen@szu.edu.cn) and Qingyao Wu (wuqingyao.china@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61502177 and Grant 61773268, in part by the Guangdong Provincial Scientific and Technological funds under Grant 2017B090901008 and Grant 2017A010101011, in part by the Fundamental Research Funds for the Central Universities under Grant D2172500, in part by the Pearl River S&T Nova Program of Guangzhou under Grant 201806010081, in part by CCF-Tencent Open Research Fund under Grant RAGR20170105, and in part by the Tencent Rhinoceros Birds-Scientific Research Foundation for Young Teachers of Shenzhen University.

ABSTRACT Feature selection has been a powerful tool to handle high-dimensional data. Most of these methods are biased toward the highest rank features which may be highly correlated with each other. In this paper, we address this problem proposing stratified feature ranking (SFR) method for supervised feature ranking of high-dimensional data. Given a dataset with class labels, we first propose a subspace feature clustering (SFC) to simultaneously identify feature clusters and the importance of each feature for each class. In the SFR method, the features in different feature clusters are separately ranked according to the subspace weight produced by SFC. After that, we propose a stratified feature weighting method for ranking the features such that the high rank features are both informative and diverse. We have conducted a series of experiments to verify the effectiveness and scalability of SFC for feature clustering. The proposed SFR method was compared with six feature selection methods on a set of high-dimensional datasets and the results show that SFR was superior to most of these feature selection methods.

INDEX TERMS Data mining, computational and artificial intelligence, clustering algorithms, feature selection.

I. INTRODUCTION

High-dimensional data present a big challenge to supervised learning due to the "curses of high-dimensionality" [12]. For example, a gene expression data which measures the expression levels of genes in experiments, often consists of thousands of genes. In classifying such data, learning models tend to occur overfitting phenomenon and become less comprehensible, because it is often found that only a small portion of genes are highly correlated to the samples, while most genes are irrelevant. To deal with such problem, feature selection is one effective means to selected optimal feature set which contains discriminative features in high-dimension data.

Over the past decades, feature selection has been playing a important role in dealing with high-dimensional data, such as removing irrelevant features [3], [20], [26], [32]. Among them, feature ranking is a type of popular feature selection method which computes the degrees of dependency of individual features with respect to class and select features according to the degrees. Generally speaking, feature selection methods can be mainly classified into three families, i.e., filter methods, wrapper methods and embedded methods. The filter methods select feature subsets according to the intrinsic characteristics of the data without involving any learning algorithm. The typical supervised filter methods include Fisher score [31] and Relief-F [22], [24]. In wrapper methods, the predictor is treated as a black box while the predictor performance as the objective function to evaluate the feature subset [16]. Despite these type of methods can get good predictor performance, such methods are usually time-consuming. Embedded methods include feature selection as part of the training process. Among the three types of methods, embedded methods are superior to others in many respects, and have received more and more attention [5], [15], [27], [28], [35]. Typical criteria to evaluate the degrees of dependency include the measures of correlation between the feature and the class, or the uncertainty measures used in information theory.

However, the above methods are most effective for statistically independent features, but have low ability in identifying group features that can be used to predict the class. They are biased toward the high rank features, but such features may be highly correlated with each other. Since the correlated features may share similar properties and are redundant, we wish to select more discriminant information with minimum correlations for classification tasks. Kong *et al.* proposed an uncorrelated feature selection (exclusive $\ell_{2,1}$) [23]. In their method, a 2-feature group is formed if the pearson correlation between two features is higher than a user defined threshold. Then the standard $\ell_{2,1}$ regularization will be introduced for each feature group in order to depress the high correlated feature pairs. However, it is difficult to set proper threshold and it is time-consuming to construct feature groups from high-dimensional data since the candidate number of feature groups is d^2 where d is the number of features.

In this paper, we propose a Stratified Feature Ranking (SFR) method for supervised feature selection from high-dimensional data. In this method, we first propose a Subspace Feature Clustering (SFC) to simultaneously identify feature clusters and the importance of each feature for each class. SFC extends the Subspace Weighting Co-Clustering (SWCC) [4] by consuming the class labels. With the co-clustering result of SFC, features in different feature clusters are separately ranked according to the subspace weights learned by SFC. Since features in the same feature cluster are higher correlated than features in different feature clusters, we propose a stratified feature weighting method for ranking the features such that the high rank features are both informative and diverse.

We conducted experiments on both synthetic data and benchmark datasets to investigate the performance of our methods. We compared SFR with six feature ranking methods on 12 high-dimensional datasets, including 5 gene expression datasets and 7 image datasets. The results show that SFR outperformed other feature ranking methods on most results. We also investigate the relationship between the performance and parameters of SFR. Experimental results show that our method can select features which are both informative and diverse. Therefore, SFR is effective for high-dimensional data.

The rest of this paper is organized as follows. We review related work on feature selection and co-clustering on section II. Then, we present the stratified feature selection method in Section III. The feature selection results are presented in Section V. Conclusions and future work are given in Section VI.

II. RELATED WORK

In this section, we give a brief review of related work on both feature selection and co-clustering.

A. FEATURE SELECTION

Feature selection, also often called as variable selection, is a process to determine the "best" subset of features for prediction. This task can date back to 1940's [19], and research in this area gained substantial momentum starting in the early 1960's due to increased computing power. The early research on feature selection mainly focuses on linear regression. Gradually, research on this area has been expanded to cover classification and clustering problems. Over the past decades, a number of feature selection methods have been proposed. Various research and widespread applications indicated the efficiency of feature selection methods to remove irrelevant features and gain great improvement in performance [3], [20], [32].

Feature selection methods can be mainly classified into three groups, i.e., filter methods, wrapper methods and embedded methods. The filter methods select feature subsets according to intrinsic characteristics of the data without involving any learning algorithm. The typical supervised filter methods include Fisher score [31] and Relief-F [22], [24]. In wrapper methods, the predictor is treated as a black box while the predictor performance as the objective function to evaluate the feature subset [16], but such methods are usually time-consuming. Embedded methods include feature selection as part of the training process. Among the three types of methods, embedded methods are superior to others in many respects, and have received more and more attention [15], [27], [28], [35].

Let $\mathbf{X} \in \mathcal{R}^{d\times n}$ be a dataset with n objects $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where $\mathbf{x}_i \in \mathcal{R}^{d\times 1}$. $\mathbf{X}$ is associated to nc classes $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_{nc}\}$ in which $\mathbf{c}_l$ consists of all objects in the l-th class. Let μ_l be the mean vector of the l-th class and μ be the overall mean vector of the original data. He *et. al* proposed a Laplacian Score method which evaluates the features according to their locality preserving power [18]. The Laplacian Score of the j-th feature is defined as follows

$$\mathcal{L}_r = \frac{\widetilde{\mathbf{f}}_r^T \mathbf{L}_A \widetilde{\mathbf{f}}_r}{\widetilde{\mathbf{f}}_r^T \mathbf{D}_A \widetilde{\mathbf{f}}_r} \tag{1}$$

where $\mathbf{L}_A = \mathbf{D}_A - \mathbf{A}$ is the graph Laplacian, in which $\mathbf{D}_A \in \mathcal{R}^{d\times d}$ is a diagonal matrix in which the j-th diagonal element $d_{jj} = \sum_{i=1}^{n} a_{ji}$. $\mathbf{A} = \{a_{ij}\}_{i,j=1}^{n}$ shares the same meaning as the sparse affinity matrix in LPP. $\widetilde{\mathbf{f}}_r$ is defined as

$$\widetilde{\mathbf{f}}_r = \mathbf{f}_r - \frac{\mathbf{f}_r^T \mathbf{D}_A \mathbf{1}}{\mathbf{1}^T \mathbf{D}_A \mathbf{1}} \mathbf{1} \tag{2}$$

where $\mathbf{f}_r$ is the r-th feature.

Sugiyama *et al.* proposed a local Fisher discriminant analysis (LFDA) [34]. LFDA preserves the local structure by maximizing the local between-class separability and minimizing the local within-class scatters simultaneously. LFDA finds a projection matrix $\mathbf{W} \in \mathcal{R}^{n\times d}$ by solving the following objective function

$$\max_{\mathbf{W}} Tr((\mathbf{W}^T \overline{\mathbf{S}}^w \mathbf{W})^{-1} \mathbf{W}^T \overline{\mathbf{S}}^b \mathbf{W}) \tag{3}$$

where $\overline{\mathbf{S}}^w$ is the local within-class scatter matrix defined as

$$\overline{\mathbf{S}}^w = \frac{1}{2} \sum_{i,j=1}^{n} \overline{a}_{ij}^w (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \tag{4}$$

where

$$\overline{a}_{ij}^{w} = \begin{cases} \frac{a_{ij}}{|\mathbf{c}_l|} & \text{if } \mathbf{x}_i, \mathbf{x}_j \in \mathbf{c}_l \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$\overline{\mathbf{S}}^b$ is the local between-class scatter matrix defined as

$$\overline{\mathbf{S}}^b = \frac{1}{2}\sum_{i,j=1}^{n} \overline{a}_{ij}^{b}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \tag{6}$$

where

$$\overline{a}_{ij}^{b} = \begin{cases} a_{ij}(\frac{1}{n} - \frac{1}{|\mathbf{c}_l|}) & \text{if } \mathbf{x}_i, \mathbf{x}_j \in \mathbf{c}_l \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Here a_{ij} in Eqs. (5) and (7) have the same meanings as the sparse affinity matrix in LPP.

Yan *et. al* proposed a graph embedding framework to extract features [38]. In this framework, the graph embedding aims to learn a projection matrix $\mathbf{W} \in \mathcal{R}^{d \times m}$ by solving the following objective function

$$\min_{\mathbf{W}^T\mathbf{B}\mathbf{W}=\mathbf{I} \text{ or } \mathbf{W}^T\mathbf{W}=\mathbf{I}} Tr(\mathbf{W}^T\mathbf{X}^T\mathbf{L}_A\mathbf{X}\mathbf{W}) \tag{8}$$

With different combinations of **A** and **B**, the linear graph embedding framework leads to many popular linear dimensionality reduction methods [38]. For example, $a_{ij} = \frac{1}{n}$ for $i \neq j$ and $\mathbf{B} = \mathbf{I}$ for PCA, $a_{ij} = \frac{\delta_{c_i,c_j}}{n_{c_i}}$ and $\mathbf{B} = \mathbf{I} - \frac{1}{n\mathbf{1}\mathbf{1}^T}$ for LDA where n_{c_i} is the number of objects in the class that the i-th object belonging to and the binary value $\delta_{c_i,c_j} = 1$ indicates that the i-th and j-th objects are in the same class.

Based on the graph embedding framework, Gu et. proposed a joint feature selection and subspace learning method FSSL, which minimizes the graph-preserving criterion and uses $\ell_{2,1}$ of the projection matrix for regularization [15]. FSSL finds a projection matrix $\mathbf{W} \in \mathcal{R}^{n \times d}$ by solving the following problem

$$\min_{\mathbf{W}^T\mathbf{X}\mathbf{D}_A\mathbf{X}^T\mathbf{W}=\mathbf{I}} \left[\|\mathbf{W}\|_{2,1} + \gamma Tr(\mathbf{W}^T\mathbf{X}\mathbf{L}_A\mathbf{X}^T\mathbf{W}) \right] \tag{9}$$

where $\mathbf{A}$, $\mathbf{D}_A$ and $\mathbf{L}_A$ have the same meanings as in the graph embedding framework.

Nie *et al.* proposed a Robust Supervised Feature selection model (RFS) model [28], by minimization $\ell_{2,1}$-norms of both loss of least square regression and regularization term as

$$\min_{\mathbf{W},\mathbf{b}} \left(\left\| X^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} \right\|_{2,1} + \gamma \|\mathbf{W}\|_{2,1} \right) \tag{10}$$

where $\mathbf{W} \in \mathcal{R}^{d \times c}$ and $\mathbf{b} \in \mathcal{R}^{c}$ are to be estimated and $\gamma > 0$ is the regularized parameter.

However, the above methods are biased toward the high rank features, but such features may be highly correlated with each other. Peng *et al.* [29] proposed a feature selection method based on the principle of Max-Relevance and Min-Redundancy. They used a first-order incremental process to attain optimal feature set. Yan *et al.* incorporated the correlation bias reduction (CBR) strategy into the process of support vector machine recursive feature elimination to boost the performance of supervised feature selection [37]. Das *et al.* [10] suggested several reasons for choosing diverse features : 1) it increases the interpretability of the selected features, since we are assured that they not redundant and are more representative of the original feature space and 2) the correlated features can slow down the convergence of algorithms such as the stochastic gradient. Since the correlated features may share similar properties and are redundant, we wish to select more discriminant information with minimum correlations for classification tasks.

Das *et al.* proposed a wrapper feature selection method, which aims to predict the class labels using linear regression on a small subset of features and uses a greedy and local search based approximation algorithm to obtain the selected features. But their method is time-consuming. Kong *et al.* [23] proposed an uncorrelated feature selection (exclusive $\ell_{2,1}$). In their method, a 2-feature group will be formed if the pearson correlation between two features is higher than a user defined threshold. Then the standard $\ell_{2,1}$ regularization will be introduced for each feature group in order to depress the high correlated feature pairs and select at most one feature from most feature groups. They propose to optimize the following objective function

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times c}} f(\mathbf{W}) + \alpha \sum_{t=1}^{m} \|\mathbf{W}_{\mathcal{G}_t}\|_{2,1}^2 \tag{11}$$

where $\mathbf{W} \in \mathbb{R}^{d\times c}$, $f(\mathbf{W})$ is cost function and $\{\mathcal{G}_1, \cdots, \mathcal{G}_m\}$ are m feature groups. However, it is difficult to set proper threshold. If the threshold is too small, the number of feature groups m will be too large such that most important features will be buried. If the threshold is too big, the number of feature groups m will be too small such that the correlations between most features will be ignored.

B. CO-CLUSTERING

Co-clustering [14], also called bi-clustering [9], is a process of simultaneously clustering rows and columns of a data matrix. Recently, it has been applied in a variety of areas such as text mining [1], bioinformatics [30] and recommendation systems [13].

Compared to traditional clustering methods which are often proposed to cluster samples based on their distribution on feature space, co-clustering methods are proposed to make full use of the duality information between samples and features. For example, in document data, it is reasonable to assume that document clusters are formed based on their association with word clusters, and in the meanwhile, word clusters can be constructed based on their link in document clusters. In those special type of data, co-clustering methods have been widely used to analyze the latent structure which exists between samples with features, and gain better clustering performance. Several co-clustering models have been formulated, including hierarchical co-clustering, spectral co-clustering [36] and partitional co-clustering [2].

Partitional co-clustering is a classical co-clustering method which has been verified to be effective in clustering large data [17]. It iteratively partitions a data matrix into $k \times l$ disjoint co-clusters, where k is the number of object clusters and l is the number of feature clusters. Based on a partition process, quite a few partitional co-clustering algorithms have been proposed. Banerjee *et al.* [2] introduced minimum Bregman information (MBI) to co-clustering and proposed a Bregman Block Average co-clustering algorithm (BBAC). It attained optimal matrix approximation which simultaneously generalizes the maximum entropy and the standard least squares. In this method, approximation error is measured by Bregman divergences, which is a class of loss function. The squared Euclidean distance is a special case of Bregman divergences. However, since it didn't distinguish the row vectors and column vectors, BBAC cannot identified the noise values which widespreadly exists in high-dimensional data.

Dhillon *et. al* [11] proposed an information-theoretic co-clustering (ITCC). They regarded rows and columns in a data matrix as two discrete random variables and data matrix as a joint probability distribution between these two variables. In their method, optimal co-clustering result was attained by minimizing the mutual information loss between the original random variables and the clustered random variables.

Recently, inspired by soft subspace clustering [6], [8], [21], weighting technique has gained more attention and was gradually introduced into co-clustering. Sarazin *et al.* [33] proposed a feature group weighting co-clustering method on topological maps model, which assigns weights to co-clusters and learns the weights during the topological biclustering process. Chen *et al.* [4] proposed a subspace weighting partitional co-clustering method. In their method, they introduced a subspace weight matrix into co-clustering to indicate the importance of each feature on each object cluster. They can find optimal feature subsets which have strong relationships with a class by using this subspace weighting matrix. Further, Chen *et al.* [7] proposed a two-way subspace weighting partitional co-clustering method which is robust and efficient to clustering high dimensional data. In their method, one more subspace weight matrix is defined to importance of each object on each feature cluster.

III. STRATIFIED FEATURE RANKING METHOD

Inspired by Chen *et al.* [4], [7] which can cluster high correlated features in same subspace, we can cluster high correlated features and select features in same subspace from different clusters to reduce redundancy. In this paper, we propose a stratified feature selection method. In the new method, we first cluster the features into a set of feature clusters. In order to attain a final ranked feature list from multiple feature clusters, we propose a stratified weighting ranking method to generate a ranked feature list, which ranks the features according to the subspace feature weights and feature clusters.

The procedure of the stratified feature ranking method is shown in Figure 1. Given a labeled dataset $\mathbf{X}$ with m features $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_m\}$, we first cluster $\mathbf{F}$ to l disjoint feature clusters $\{\mathbf{Q}_1, \ldots, \mathbf{Q}_l\}$, such that $\mathbf{Q}_j \bigcap \mathbf{Q}_i = \emptyset (\forall i \neq j)$ and $\bigcup_{j=1}^{l} \mathbf{Q}_j = \mathbf{F}$. Finally, we rank the m features with a stratified weighting feature ranking method.

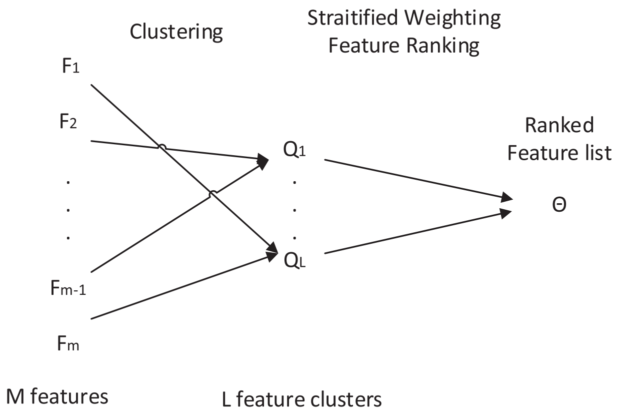


FIGURE 1. Illustration of the procedure of the stratified feature ranking method.

In the following, we describe the feature clustering and stratified weighting feature ranking.

A. FEATURE CLUSTERING

Let $\mathbf{X} \in \mathcal{R}^{n \times m}$ be a labeled data matrix with n objects and m features. To cluster $\mathbf{X}$ into k row clusters and l column clusters, Chen introduced a subspace weight matrix $\mathbf{C} \in \mathcal{R}^{k \times l}$ in which c_{gj} is the weight of the j-th column in the g-th row cluster. The objective function of SWCC is as follows [4]

$$\begin{aligned}
\min_{\mathbf{U},\mathbf{V},\mathbf{Z},\mathbf{C}} & \frac{1}{mn} \sum_{g=1}^{k} \sum_{h=1}^{l} \sum_{i=1}^{n} \sum_{j=1}^{m} u_{ig} v_{jh} c_{gj} (x_{i,j} - z_{g,h})^2 \\
& + \frac{\eta}{m} \sum_{g=1}^{k} \sum_{j=1}^{m} c_{gj} \log c_{gj} \\
s.t. & \sum_{g=1}^{k} u_{ig} = 1,\ u_{ig} \in \{0,1\},\ \sum_{h=1}^{l} v_{jh} = 1, \\
& v_{jh} \in \{0,1\},\ \sum_{j=1}^{m} c_{gj} = 1,\ c_{gj} \in (0,1)
\end{aligned} \tag{12}$$

In supervised feature selection task, since the class labels in X are known, $\mathbf{U} \in \mathcal{R}^{n \times k}$ can be directly constructed with the given class labels by setting $u_{ig} = 1$ if $\mathbf{x}_i$ belongs to the g-th class, and 0 otherwise. The objective of feature clustering is to partition m features in $\mathbf{X}$ into l feature clusters. To achieve this goal, we extend problem (11) to the following Subspace Feature Clustering objective function

$$\begin{aligned}
\min_{\mathbf{V},\mathbf{Z},\mathbf{C}} & \frac{1}{mn} \sum_{g=1}^{k} \sum_{h=1}^{l} \sum_{i=1}^{n} \sum_{j=1}^{m} u_{ig} v_{jh} c_{gj} (x_{i,j} - z_{g,h})^2 \\
& + \frac{\eta}{m} \sum_{g=1}^{k} \sum_{j=1}^{m} c_{gj} \log c_{gj} \\
s.t. & \sum_{h=1}^{l} v_{jh} = 1,\ v_{jh} \in \{0,1\},\ \sum_{j=1}^{m} c_{gj} = 1,\ c_{gj} \in (0,1)
\end{aligned} \tag{13}$$

Algorithm 1 Subspace Feature Clustering (SFC)

```
Input: the labeled dataset $\mathbf{X}$, the number of feature clusters $l$ and the regularization parameter $\eta$.
Output: the feature clustering result $\mathbf{V}$ and the subspace weight matrix $\mathbf{C}$.
Construct a binary matrices $\mathbf{U} \in \mathcal{R}^{n\times k}$ from the given class labels, in which $u_{ig} = 1$ indicates that the $i$-th object belongs to the $g$-th class.
$i := 0$
Randomly initialize $\mathbf{Z}$ and let $c_{gj} = \frac{1}{m}$ for $\forall$ $g$ and $j$.
repeat
    Update $\mathbf{V}^{i+1}$ by (14).
    Update $\mathbf{Z}^{i+1}$ by (15).
    Update $\mathbf{C}^{i+1}$ by (16) and (17).
    $i := i + 1$
until (13) obtains its local minimum value
```

Apparently, problem (13) has the same solutions of **V**, **Z** and **C** as problem (12). According to the work in [4], we know the solutions of **V**, **Z** and **C** to problem (13) as as follows. If **Z** and **C** are fixed, the optimal solution to **V** is

$$\begin{cases} v_{jh} = 1 \quad \text{if } P_{(h)} \leq P_{(t)} \text{ for } 1 \leq t \leq L \text{ where} \\ P_{(t)} = \sum_{g=1}^{k}\sum_{i=1}^{n} u_{ig}c_{gj}(x_{ij} - z_{gt})^2 \\ v_{jt} = 0 \quad \text{for } t \neq h \end{cases} \tag{14}$$

If **V** and **C** are fixed, the optimal solution of **Z** is

$$z_{gh} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} u_{ig}v_{j,h}c_{gj}x_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{m} u_{ig}v_{j,h}c_{gj}} \tag{15}$$

If **Z** and **V** are fixed, the optimal solution of **C** is

$$c_{gj} = \frac{\exp\{-\frac{E_{gj}}{\eta}\}}{\sum_{j'=1}^{m}\exp\{-\frac{E_{gj'}}{\eta}\}} \tag{16}$$

where

$$E_{gj} = \frac{1}{n}\sum_{h=1}^{l}\sum_{i=1}^{n} u_{ig}v_{jh}\,(x_{ij} - z_{gt})^2 \tag{17}$$

We summarize the detailed algorithm to the objective function (13) in Algorithm 1, which is denoted as Subspace Feature Clustering (SFC). In this algorithm, **V**, **Z** and **C** are alternately updated until convergence. Since in each step we obtain the minima of problem (13), it is strictly decreasing to local minima during the optimization process. Supposing that the algorithm converges in r iterations, the computational complexity of SFC is $O(rnmkl)$. Since the computational cost of SFC has linear relationship with the number of the objects and size of dimension, which is the same with k-means and BBAC, we can know that it can be efficient to cluster large high-dimensional data. Since the SFC algorithm is sensitive to the initial cluster centers, we can run *SFC* multiple times with different initial cluster centers to produce multiple feature clusters. Given each l and η, we run $SFC(\mathbf{X}, l, \eta)$ with different initial cluster centers to produce a co-clustering result set $\mathcal{H}$, evaluate each co-clustering result $\mathbf{H} \in \mathcal{H}$ and select $\mathbf{H}^* \in \mathcal{H}$ as the best clustering result for feature selection. To evaluate a co-clustering result **H**, we use the learned $\mathbf{V}^*$, $\mathbf{Z}^*$ and $\mathbf{C}^*$ to predict a label for each object $\mathbf{x}_i \in \mathbf{X}$, by assigning $\mathbf{x}_i$ to the class with minimal weighted distance, i.e.,

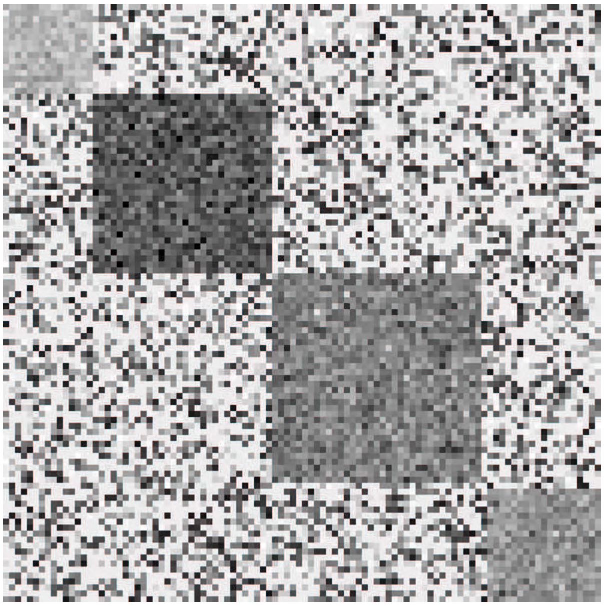


FIGURE 2. Plot of a typical synthetic dataset D_1.

$$label(\mathbf{x}_i) = \arg\min_{g}\left[\sum_{h=1}^{l}\sum_{j=1}^{m} v^{*}_{jh}c*_{gj}(x_{ij} - z^{*}_{gh})^2\right] \tag{18}$$

After that, different evaluation indices can be used to evaluate the classification result obtained from **H** by comparing the predicted labels with the given class labels, including *NMI*, accuracy, recall and so on. Usually, the number of feature clusters l is given by user. In practice, we can also select multiple l to produce multiple co-clustering results and select the best co-clustering result. Finally, we select a best co-clustering result $\mathbf{H}^*$ for each parameter setting in which l disjoint feature clusters $\{\mathbf{Q}_1, \ldots, \mathbf{Q}_l\}$ are used for feature ranking.

B. STRATIFIED WEIGHTING FEATURE RANKING

Since the learned weight matrix **C** in $\mathbf{H}^*$ identifies contribution of each feature to each class, a natural way is to rank the features according to **C**. In the commonly-used least square regression based feature selection method, a projection matrix $\mathbf{W} \in \mathcal{R}^{m\times k}$ is learnt and the importances of the features can be estimated as $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^m\|_2\}$ [4], [28]. Since the subspace weight matrix **C** in SFC is non-negative, we can evaluate the importances of features according to $\{\|\mathbf{c}_1\|_1, \ldots, \|\mathbf{c}_m\|_1\}$.

To select r import features, if we just select r high-rank features according to **C**, the selected feature may be concentrated on a small number of feature clusters and they are highly correlated with each other. To select features which

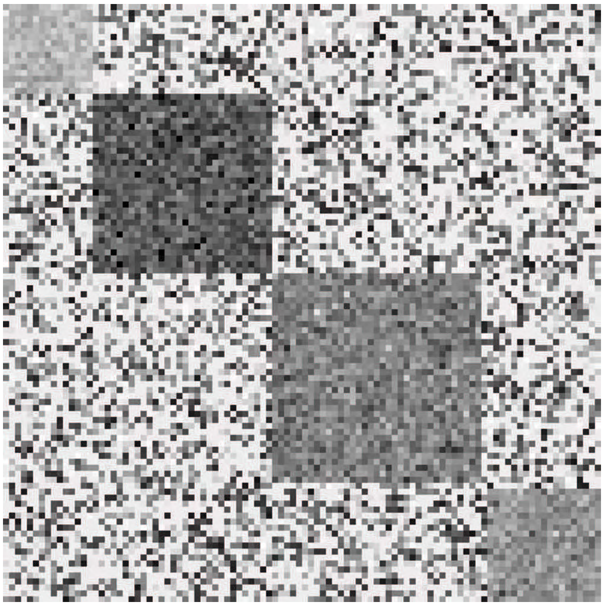


FIGURE 3. The average entropies of **C** versus η from the results of SFC on D_1.

are both informative and diverse, we propose a stratified weighting method for ranking features. In the new method, we first sort features in each feature cluster in ascending order order according to $\{\|\mathbf{c}_1\|_1, \ldots, \|\mathbf{c}_m\|_1\}$. Assume the index of the j-th feature in the corresponding feature cluster is ℓ_j, we compute a stratified weighting feature ranking vector $\theta \in \mathcal{R}^m$ for m feature in which θ_j is defined as

$$\theta_j = \left\|\mathbf{c}_j\right\|_1 \lambda^{\ell_j} \tag{19}$$

where $\lambda \in (0, 1]$ is the stratified weighting parameter which is given by user. Here, λ^{ℓ_j} is used to geometrically decrease the weights in a feature cluster. If $\lambda = 1$, θ_j degenerate to $\left\|\mathbf{c}_j\right\|_1$ which is the conventional ranking method without stratification. If $\lambda < 1$, the features in a feature cluster will be assigned to a set of geometrically decreased weights such that the features with lower order will be deemphasized. In such way, we can avoid selecting too many features from a feature cluster. Therefore, we can select features which are both informative and diverse according to θ.

C. THE FEATURE RANKING ALGORITHM

The detailed procedure of above method is summarized in in Algorithm 2, which is denoted as Stratified Feature Ranking (SFR). In the new method, we first use SFC to cluster m features in $\mathbf{X}$ into l disjoint feature clusters. Finally, we rank the m features with a stratified weighting feature ranking method.

IV. EXPERIMENTAL RESULTS AND ANALYSIS ON FEATURE CLUSTERING

In this section, we conduct a series of experiments on the a typical synthetic dataset to demonstrate the performance and investigate the Stratified Feature Ranking (SFR)

Algorithm 2 Stratified Feature Ranking (SFR)

```
Input: the labeled dataset $\mathbf{X}$, the number of feature clusters $l$, the regularization parameter $\eta$, the stratified weighting parameter $\lambda$ and repeated number of clustering $rep$.
Initialize an empty clustering result list $\mathcal{H}$.
for $j = 1$ to $rep$ do
    Call $SFC(\mathbf{X}, l, \eta)$ with randomly initialized cluster centers to produce a clustering results $\mathbf{H}$.
    Add $\mathbf{H}$ into $\mathcal{H}$.
end for
Validate each clustering result $\mathbf{H} \in \mathcal{H}$, and select $\mathbf{H}^* \in \mathcal{R}$ which has the best clustering result.
Compute the $\ell_1$-norm of $\mathbf{C}$ as $\{\|\mathbf{c}_1\|_1, \ldots, \|\mathbf{c}_m\|_1\}$ and sort features in each feature cluster in ascending order order according to these values.
Compute $\theta \in \mathcal{R}^{m\times 1}$ according to Eq. (19).
Sort $\theta$ in descending order, and select top $r$ ranked features as ultimate result.
```

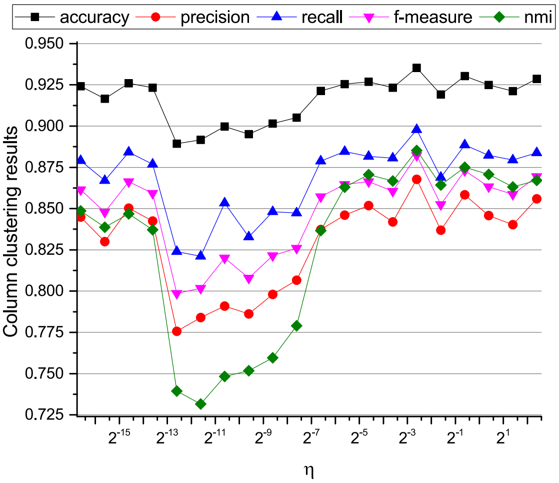
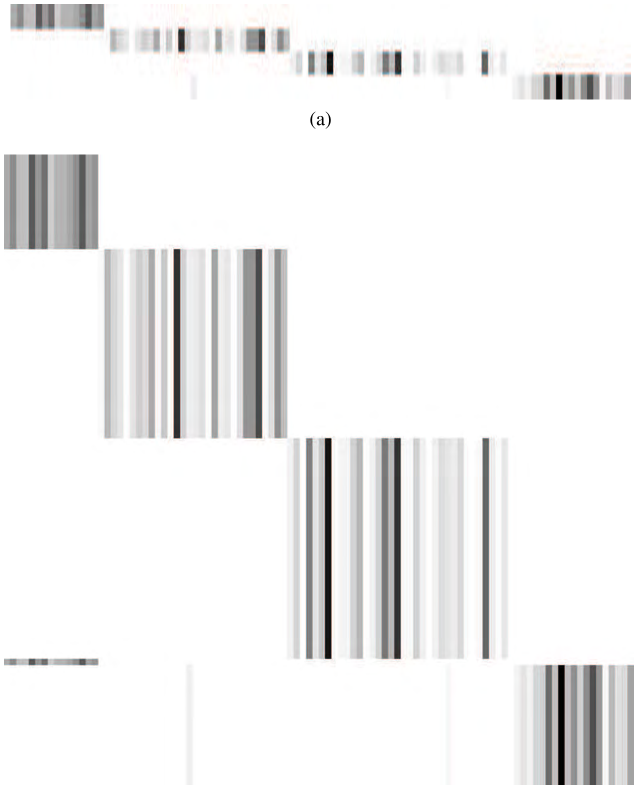


FIGURE 4. The feature clustering results of SFC versus η on D_1.

A. EXPERIMENT SETUP

We generated a dataset D1 which has 100 rows and 100 columns. We present it in figure 2, where higher values mainly exist near diagonal blocks, and we plotted them in the darker color while lower values were plotted in lighter color. From figure 2 ,we can see D_1 can be divided into 16 blocks. The four co-clusters existed in the diagonal blocks, while noise random exists in other blocks.

In the experiments, we used D1 to investigate the subspace weights of the SFC algorithm. As it exist 4 co-clusters in data, we set $L = 4$ and chose 20 real values $\{1.2^0E-4, \ldots, 1.2^{19}E-4\}$ for η. Since final clustering results are affected by initial clusters, we randomly generated 100 initial cluster centers and produced results with respect to different initial clusters. In the end, we totally got 2, 000 results to analyze the impact of the parameters to final co-clustering result in Stratified Feature Clustering.



(a)

(b)

FIGURE 5. Weight matrix of the selected clustering result by SFC on D_1. (a) Weight matrix of **C**. (b) Weight matrix **W**.

TABLE 1. Characteristics of 5 gene expression data sets.

Name	Abbr.	#Genes	#Patients	#Classes
breast.3.class	**BR3**	4,869	96	3
SRBCT	**ST**	2,308	83	4
Brain-tumor2	**BT2**	10,367	50	4
11-tumors	**11T**	12,533	174	11
14-tumors	**14T**	15,009	308	26

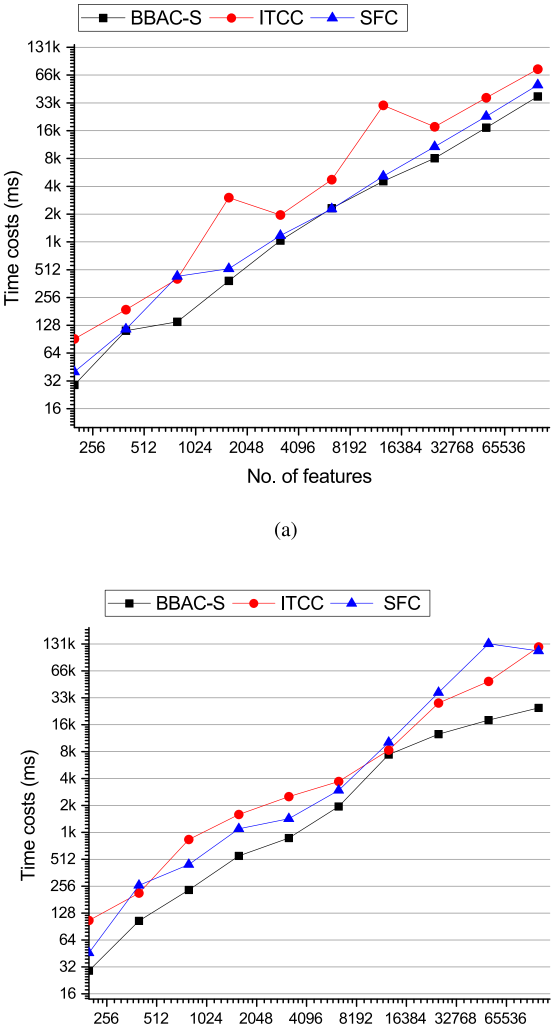
B. IMPACTS OF η ON C

We computed the average entropy of C for each clustering result. The results were plotted in Figure 3. From figure 3, we can see that the average entropy of C was affected by η when η is small. It increased with η, and then decreased immediately. After that, it increased rapidly as η increase. When η is large, the object function was mainly affected by entropy regularizer which force weights evener, so the average entropy of **C** didn't change too much in the end.

C. IMPACTS OF η ON FEATURE CLUSTERING RESULTS

We use the five common used evaluation indices to measure the quality of all feature clustering results. Since clustering result was sensitive to initial clusters, we computed the average value of each evaluation from 100 results, and reported the average results in Figure 4

From Figure 4, we can see that when η is small, all evaluation indices are low, then they increased rapidly. Next they slightly dropped. Finally, they didn't change too much with the increase of η. When η is small, the regularization is slightly, the weights are mainly concentrated on a few variables. On the contrary, the weights are evenly distributed when η is setting to a relatively big value. In both cases, SFC finally often produced poor clustering results.



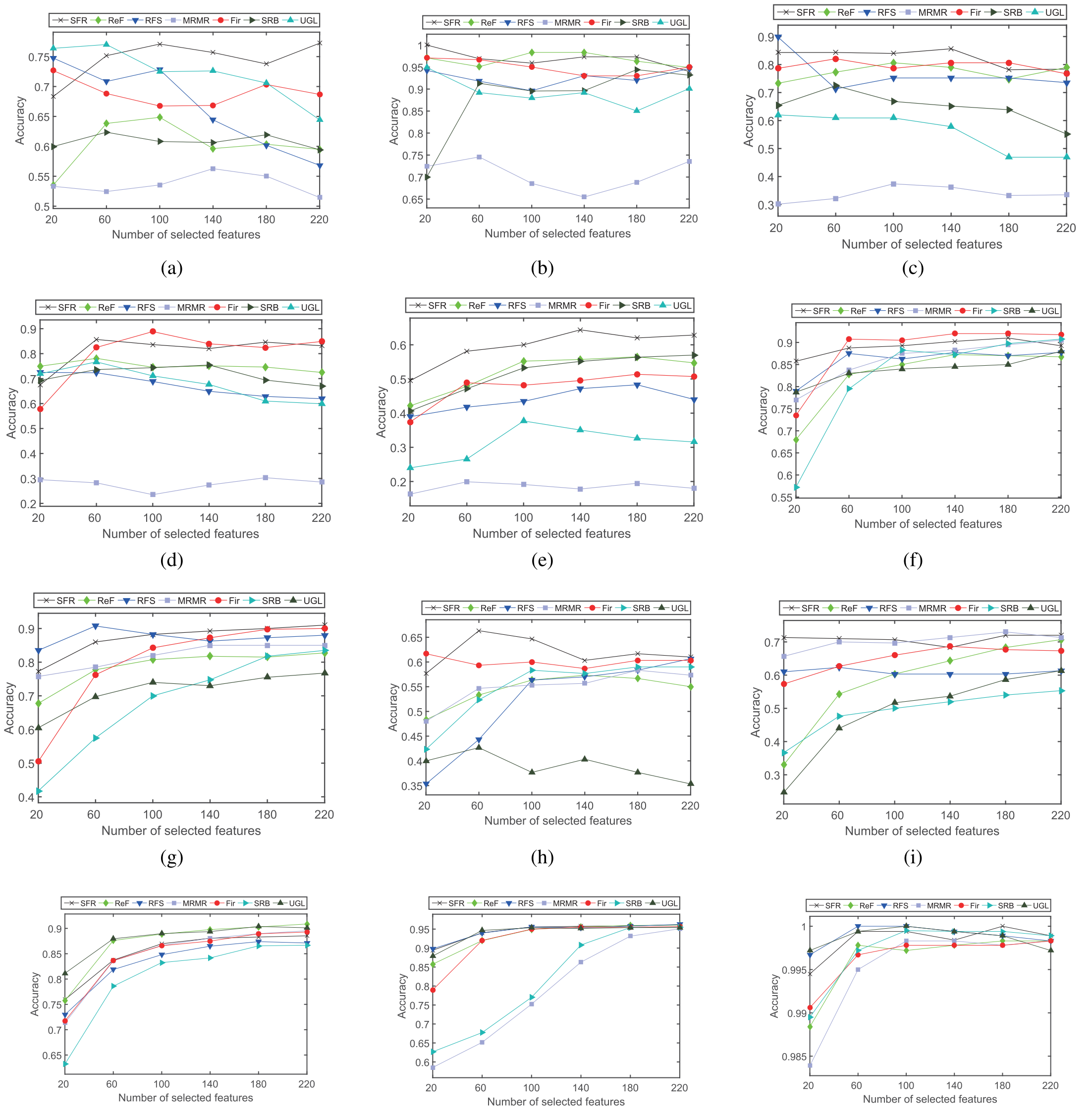
(a)

(b)

FIGURE 6. Average time costs of SFC on 20 synthetic datasets. (a) Average time costs versus the number of objects. (b) Average time costs versus the number of features.

TABLE 2. Characteristics of 7 image data sets.

Name	Abbr.	#Samples	#Features	#Classes
ORL-32x32	**OR3**	400	1024	40
ORL-64x64	**OR6**	400	4096	40
Yale-32x32	**YA3**	165	1024	15
Yale-64x64	**YA6**	165	4096	15
YaleB-32x32	**YAB**	2414	1024	38
USPSdata-20	**USPS**	1854	256	10
MSRA25	**MSRA**	1799	256	12



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

FIGURE 7. The accuracies versus the number of selected features by 7 feature selection methods on 12 data sets. Here, ReF represents Relief-F, Fir represents Fisher, SRB represents SVM-RFE-CBR. (a) Results on the BR3 data set. (b) Results on the ST data set. (c) Results on the BT2 data set. (d) Results on the 11T data set. (e) Results on the 14T data set. (f) Results on the OR3 data set. (g) Results on the OR6 data set. (h) Results on the YA3 data set. (i) Results on the YA6 data set. (j) Results on the YAB data set. (k) Results on the MSRA data set. (l) Results on the USPS data set.

TABLE 3. The average classification accuracies of 7 feature selection methods on 12 benchmark datasets (the best two results on each dataset are highlighted in bold).

Data	Relief-F	RFS	MRMR	Fisher	SVM-RFE-CBR	UGL	SFR
$BR3$	60.3 ± 4.0	66.6 ± 7.3	53.7 ± 1.7	69.0 ± 2.2	60.9 ± 1.1	**72.2 ± 4.6**	**74.5 ± 3.3**
ST	**96.7 ± 1.5**	92.5 ± 1.8	70.6 ± 3.5	95.0 ± 1.7	88.0 ± 9.0	89.4 ± 3.2	**96.9 ± 2.0**
$BT2$	77.3 ± 2.8	76.7 ± 6.6	33.8 ± 2.7	**79.6 ± 1.9**	64.8 ± 5.6	55.9 ± 7.1	**82.5 ± 3.3**
$11T$	75.0 ± 1.8	67.2 ± 4.7	27.9 ± 2.4	**80.1 ± 11.2**	71.5 ± 3.4	68.1 ± 6.6	**81.1 ± 6.8**
$14T$	**52.0 ± 5.8**	43.9 ± 3.4	18.4 ± 1.3	47.7 ± 5.2	51.6 ± 6.4	31.3 ± 5.2	**59.5 ± 5.4**
$OR3$	82.8 ± 7.4	85.9 ± 3.4	86.1 ± 5.0	**88.4 ± 7.3**	82.1 ± 12.8	83.9 ± 3.0	**89.0 ± 1.8**
$OR6$	78.7 ± 5.6	**87.3 ± 2.4**	81.9 ± 4.0	79.7 ± 15.2	68.2 ± 16	71.6 ± 5.9	**87.0 ± 5.1**
$YA3$	54.5 ± 3.3	52.0 ± 10.0	54.9 ± 3.6	**60.1 ± 1.0**	54.8 ± 6.6	38.9 ± 2.6	**61.9 ± 3.1**
$YA6$	58.5 ± 13.8	60.9 ± 0.8	**70.2 ± 2.5**	64.9 ± 4.3	49.3 ± 6.8	49.0 ± 13.4	**70.9 ± 1.4**
$YB3$	**87.2 ± 5.7**	83.4 ± 5.5	84.7 ± 6.9	84.6 ± 6.6	80.4 ± 8.9	**88.0 ± 3.5**	85.3 ± 4.9
$USPS$	93.4 ± 4.0	**94.5 ± 2.4**	78.9 ± 15.1	92.1 ± 6.6	81.5 ± 14.4	94.0 ± 3.0	**94.5 ± 2.6**
$MSRA$	99.6 ± 0.4	**99.9 ± 0.1**	99.5 ± 0.6	99.7 ± 0.3	99.7 ± 0.4	**99.9 ± 0.1**	99.8 ± 0.2

D. SUBSPACE WEIGHT

The NMI value for each of $2,000$ feature clustering results was computed and the highest value is 0.885. For the co-clustering result with the highest NMI value, the subspace weight matrix $\mathbf{C}$ is drawn in Figure 5(a). Here we used a weight matrix $\mathbf{W} \in \mathcal{R}^{n \times m}$ to indicate the relation between

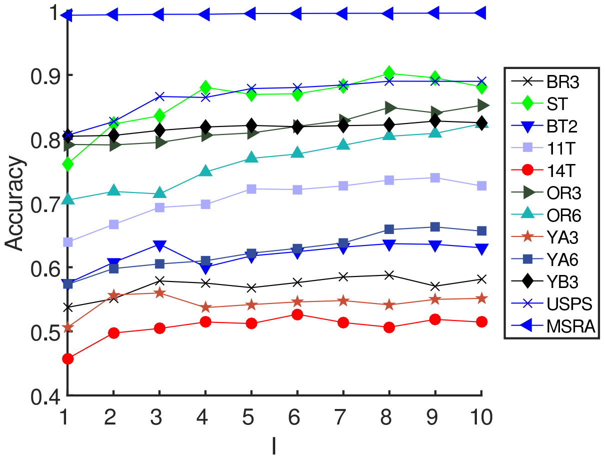


FIGURE 8. Average accuracies of SFR versus the number of feature clusters *l* on 12 datasets.

features and object, where w_{ij} reflect the importance of the j-th feature to the i-th object which is defined as

$$w_{ij} = \sum_g \sum_h u_{ig} v_{jh} c_{gj} \tag{20}$$

We plotted the weight matrices in Figure 5(b) where the darker color corresponded to higher weight. From the two figures, we can see that subspace structure of D1 can be well revealed by the subspace weights of SFC , which helps us to identify separated feature clusters.

E. SCALABILITY ANALYSIS

For analyzing the scalability of SFC to high-dimensional data, we randomly generated 10 synthetic datasets, where the number of objects were fixed to 1000. In this part of experiment, we controlled the numbers of features ranging from 200 to 12800 and generated different datasets. For fairy comparison, we set regularization parameter to 0.01 in all data. For each data, we randomly generated 100 initial feature clusters and produced different clustering results. The average time costs were plotted in Figure 6. From this figure, we can see that the execution time of every algorithm has a nearly linear relationship with the number of objects and the size of dimension. Beside, the runtime of SFC were comparable to that of BBAC-S which is BBAC with squared Eculidean distance. Since latter has been verified to be scalable for handling high-dimensional data, we can know that SFC can also scale well to high-dimensional data.

V. EXPERIMENTAL RESULTS AND ANALYSIS ON FEATURE SELECTION

In this section, we present experimental results on 12 real-life datasets to demonstrate the performance and investigate the properties of the proposed SFR method.

A. BENCHMARK DATA SETS

In this part of experiments, 12 real-life data sets were used to investigate the performance of our proposed method, including 5 gene expression data sets which were selected from http://gems-system.org/ and 7 image data sets which were downloaded from Feiping Nie's page.[1] We listed these two types of datasets in Table 1 and 2 separately.

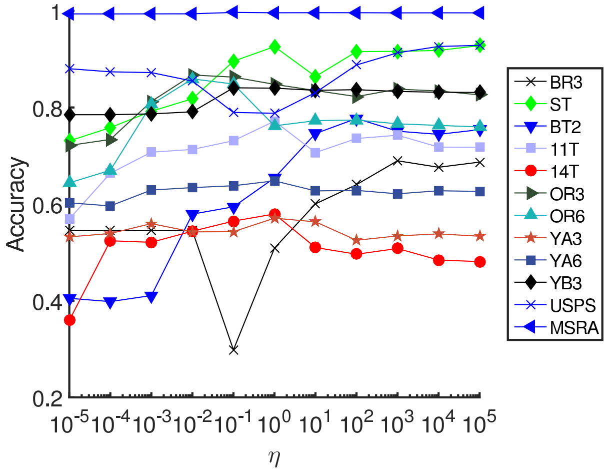


FIGURE 9. Average accuracies versus η on 9 datasets.

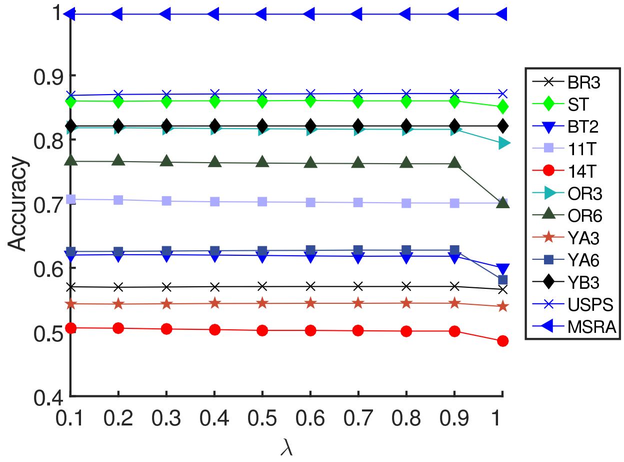
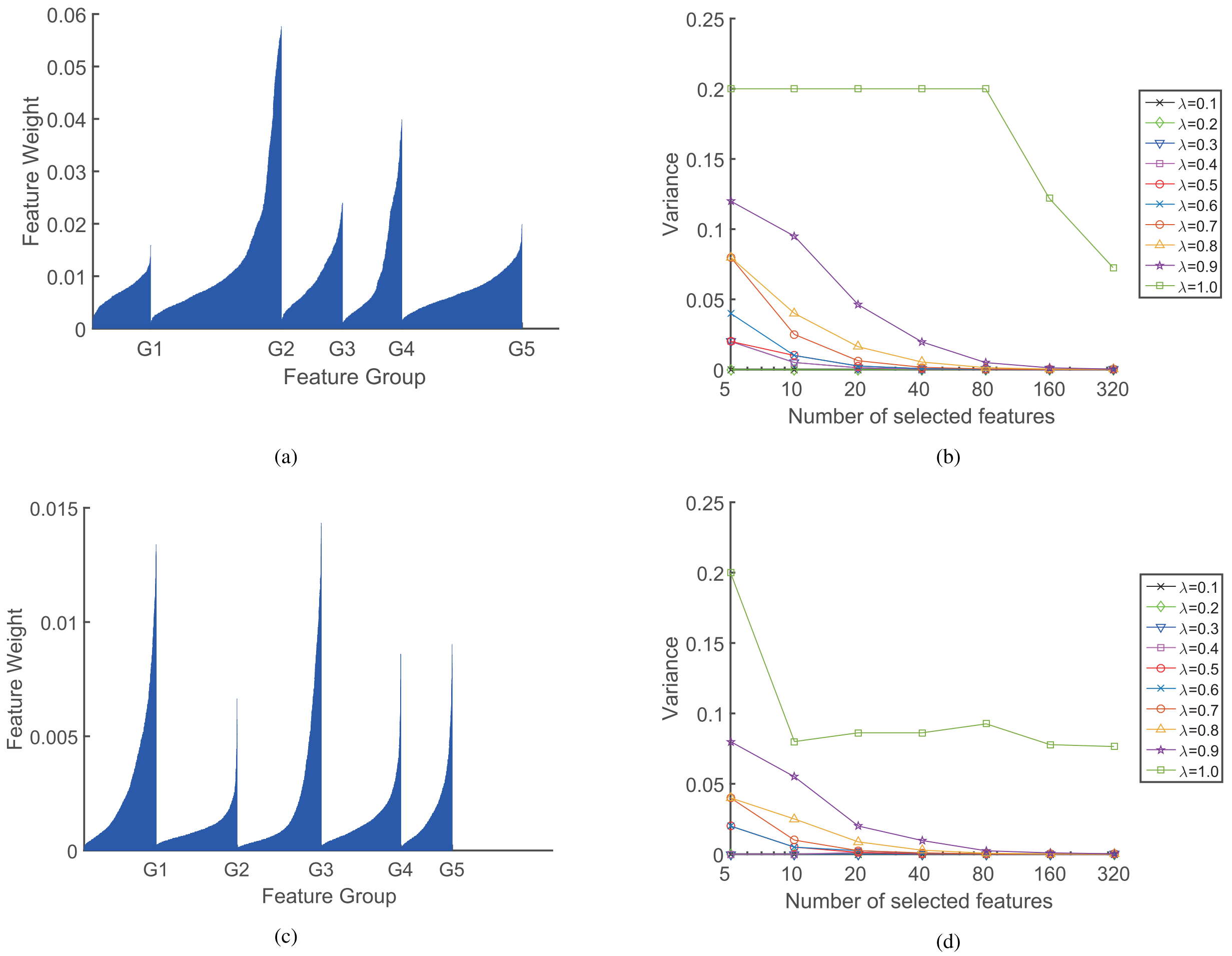


FIGURE 10. Average accuracies versus λ on 12 datasets.

B. RESULTS AND ANALYSIS

We compared SFR with six supervised feature selection methods to validate the effectiveness, including Relief-F [22], [25], RFS [28], MRMR [29], Fisher Score [31], SVM-RFE-CBR [37],UGL [23]. We set parameters of all methods in the same strategy to make the experiments fair enough, i.e., 11 values varying from 10^{-5} to 10^{5}. We set different thresholds of highly correlated feature pairs varying from 0.6 to 0.9 in UGL and SVM-RFE-CBR. We remove half of features in each iteration until 60 features are left in SVM-RFE-CBR and remove each feature in the end. For each data set, we selected a set of 10 numbers from 1 to 10 for l and 10 numbers from 0.1 to 1 for lambda to run *SFR*. The repeated number of clustering *rep* was set as 20 and *NMI* was used as the evaluation index to evaluate a clustering result in our experiments.

[1] http://www.escience.cn/system/file?fileId=82035



(a)

(b)

(c)

(d)

FIGURE 11. Feature ranking results. (a) Original feature ranking result $\{\|\mathbf{c}\|_1, \cdots, \|\mathbf{c}\|_m\}$ on the OR6 dataset ($l = 5$, $\eta = 1$). (b) Variance of percentages of selected features in feature clusters according to the feature weights in Figure 11(a). (c) Original feature ranking result $\{\|\mathbf{c}\|_1, \cdots, \|\mathbf{c}\|_m\}$ on the 14T dataset ($l = 5$, $\eta = 1$). (d) Variance of percentages of selected features in feature clusters according to the feature weights in Figure 11(c).

For each data set in Table 1 and 2, we ran seven supervised feature selection methods to select different numbers of features and performed 10-fold SVM on the data with the selected features . The maximal accuracies versus the number of selected features of 7 methods on 12 datasets are reported in Figure 7, and their average accuracies are summarized in Table 3. Overall, our proposed method SFR outperformed all other methods in accuracy on 8 of 12 datasets, especially on the **BR3**, **BT2**, **14T** datasets. To be specific, SFR has 7.5% improvement on the **14T** dataset, compared to the second best method Relief-F. From Figure 7, we can see that SFR produced the best result with only 20 features on the **ST** dataset, 100 features on the **BR3** dataset, 140 features on the **14T** dataset, 60 features on the **YA3** dataset and 20 features on the **YA6** dataset. SFR also achieved good performance for the rest datasets in average.

C. PARAMETER SENSITIVITY ANALYSIS

In this experiment, we investigate the effect of three parameters l, η and λ on the performance of SFR.

We first study the effect of l on the performance of SFR. The relationships between the average accuracies and l on 12 datasets are shown in Figure 8. From this figure, we can see that the accuracies increased with the increase of l on most datasets. On all datasets, the lowest accuracies are obtained with only one feature cluster, which indicates that the introduction of feature clustering into feature selection indeed help to select better features for classification.

The relationships between the average accuracies and η on 12 datasets are shown in Figure 9. From this figure, we can see that the accuracies increased with the increase of l on most datasets. We can see that the accuracies do not change too much with the increase of η on the **YA3**, **YA6** and **MSRA** datasets, which indicates that only a small number of features in these datasets are useful. The accuracies are obtained with medium η on the **OR3** and **OR6** datasets, which have a relative small number of features. On two very high-dimensional datasets **ST** and **BT2**, the accuracies increased with the increase of η.

The relationships between the accuracy and λ on 12 datasets are shown in Figure 10. From this figure, we can see that the accuracies were stable when $0 \leq \lambda \leq 0.9$, which indicates that the classification results are insensitive to λ. We also notice that on all datasets, the lowest accuracies are obtained when $\lambda = 1$. According to Eq. (19), we know that the stratified weighting feature ranking with $\lambda = 1$ degenerates to the conventional ranking method. Therefore, these results show that stratified feature ranking indeed improved the feature selection.

In real applications, we can set the three parameters with domain knowledge, or to attain better result by choosing the combination of parameters in a means of grid search.

D. FEATURE RANKING

In this subsection, we selected two datasets, i.e., the **OR6** and **14T** datasets, to show how the stratified weighting feature ranking method improves the classification performance. We set $l = 5$, $\eta = 1$ and $\lambda = \{0.1, 0.2, \cdots, 1.0\}$ to run SFC on the two datasets and the original feature ranking results $\{\|\mathbf{c}\|_1, \cdots, \|\mathbf{c}\|_m\}$ on the two datasets are shown in Figure 11(a) and 11(c), in which features in a feature cluster are sorted in descending order according to their original feature ranking results. Then we conduct experiment to check whether the proposed method can select diverse features. Assume we have selected r important features, we first compute the percentages of selected features in each feature cluster and then the variances of the percentages are reported in Figure 11(b) and 11(d). From the two figures, we can see that as we selected more number of features, the variances become smaller indicating that the numbers of selected features from different feature clusters become equal. We also notice that the variances with $\lambda = 1$ are much bigger than those with $\lambda < 1$. Since the results with $\lambda = 1$ are equal to those without stratified feature ranking, we know that the proposed method indeed select more balanced number of features from all feature clusters. Since the features in different feature clusters are lowly correlated with each other, we know that the proposed method indeed select more diverse features than the original feature ranking method.

VI. CONCLUSIONS

This paper presents a Stratified Feature Ranking (SFR) method for ranking features in high-dimensional data. In this method, a subspace Feature Clustering (SFC) method was presented to cluster features into a set of feature clusters, and the features in different feature clusters were separately ranked according to the subspace weights in SFC. To select both informative and diverse features according to the subspace weights in SFC, we propose a stratified weighting feature ranking method to rank features such that high rank features come from as many feature clusters as possible. The effectiveness and scalability of SFC for feature clustering was verified by experiments on a set of synesthetic datasets. SFR was compared with other six feature ranking methods on a set of high dimensional datasets. The experimental results show that SFR outperformed other six feature ranking methods on most datasets. It is experimentally verified that the new method can select features which are both informative and diverse. Therefore, it is a new tool for high-dimensional data.

In the future work, we will introduce other techniques such as ensemble learning to boost SFC. The use of SFR in real applications is also our future work.

ACKNOWLEDGMENTS

Renjie Chen and Ning Sun contributed equally to this work.

REFERENCES

[1] S. Alzahrani, B. Ceran, S. Alashri, S. W. Ruston, S. R. Corman, and H. Davulcu, "Story forms detection in text through concept-based co-clustering," in *Proc. IEEE Int. Conf. Big Data Cloud Comput. (BDCloud), Social Comput. Netw. (SocialCom), Sustain. Comput. Commun. (SustainCom) (BDCloud-SocialCom-SustainCom)*, Oct. 2016, pp. 258–265.

[2] A. Banerjee, I. Dhillon, J. Ghosh, S. Merugu, and D. S. Modha, "A generalized maximum entropy approach to bregman co-clustering and matrix approximation," *J. Mach. Learn. Res.*, vol. 8, pp. 1919–1986, Aug. 2007.

[3] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.

[4] X. Chen, J. Z. Huang, Q. Wu, and M. Yang, "Subspace weighting co-clustering of gene expression data," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 1, no. 99, pp. 1–12, May 2017.

[5] X. Chen, F. Nie, G. Yuan, and J. Z. Huang, "Semi-supervised feature selection via rescaled linear regression," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 1525–1531.

[6] X. Chen, X. Xu, J. Z. Huang, and Y. Ye, "TW-k-means: Automated two-level variable weighting clustering algorithm for multiview data," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 4, pp. 932–944, Apr. 2013.

[7] X. Chen, M. Yang, J. Z. Huang, and Z. Ming, "TWCC: Automated two-way subspace weighting partitional co-clustering," *Pattern Recognit.*, vol. 76, pp. 404–415, Apr. 2018.

[8] X. Chen, Y. Ye, X. Xu, and J. Z. Huang, "A feature group weighting method for subspace clustering of high-dimensional data," *Pattern Recognit.*, vol. 45, no. 1, pp. 434–446, 2012.

[9] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proc. AAAI*, 2000, pp. 93–103.

[10] A. Das, A. Dasgupta, and R. Kumar, "Selecting diverse features via spectral regularization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1583–1591.

[11] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2003, pp. 89–98.

[12] D. L. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality," Dept. Statist., Stanford Univ., Stanford, CA, USA, Tech. Rep. 13, Aug. 2000.

[13] T. George and S. Merugu, "A scalable collaborative filtering framework based on co-clustering," in *Proc. 5th IEEE Int. Conf. Data Mining*, Nov. 2005, p. 4.

[14] G. Govaert and M. Nadif, *Co-Clustering: Models, Algorithms and Applications*. Hoboken, NJ, USA: Wiley, 2013.

[15] Q. Gu, Z. Li, and J. Han, "Joint feature selection and subspace learning," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1294–1299.

[16] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[17] J. A. Hartigan, "Direct clustering of a data matrix," *J. Amer. Statist. Assoc.*, vol. 67, no. 337, pp. 123–129, 1972.

[18] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 507–514.

[19] H. Hotelling, "The Selection of variates for use in prediction with some comments on the general problem of nuisance parameters," *Ann. Math. Statist.*, vol. 11, no. 3, pp. 271–283, 1940.

[20] S. H. Huang, "Supervised feature selection: A tutorial," *Artif. Intell. Res.*, vol. 4, no. 2, p. 22, 2015.

[21] L. Jing, M. K. Ng, and Z. Huang, "An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 8, pp. 1026–1041, Aug. 2007.

[22] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proc. 9th Int. Workshop Mach. Learn.*, 1992, pp. 249–256.

[23] D. Kong, J. Liu, B. Liu, and X. Bao, "Uncorrelated group lasso," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1765–1771.

[24] I. Kononenko, "Estimating attributes: Analysis and extensions of RELIEF," in *Proc. Eur. Conf. Mach. Learn.*, 1994, pp. 171–182.

[25] H. Liu and H. Motoda, *Computational Methods of Feature Selection* (Chapman & Hall/CRC Data Mining and Knowledge Discovery Series). London, U.K.: Chapman & Hall, 2007.

[26] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005.

[27] M. Masaeli, G. Fung, and J. G. Dy, "From transformation-based dimensionality reduction to feature selection," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 1–8.

[28] F. Nie, H. Huang, X. Cai, and C. H. Ding, "Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1813–1821.

[29] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.

[30] R. Pensa and J. Boulicaut, "Constrained co-clustering of gene expression data," in *Proc. SIAM Int. Conf. Data Mining*, 2008, pp. 25–36.

[31] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Hoboken, NJ, USA: Wiley, 2010.

[32] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[33] T. Sarazin, M. Lebbah, H. Azzag, and A. Chaibi, "Feature group weighting and topological biclustering," in *Neural Information Processing*. Kuching, Malaysia: Springer, 2014, pp. 369–376.

[34] M. Sugiyama, "Local Fisher discriminant analysis for supervised dimensionality reduction," in *Proc. 23rd Int. Conf. Mach. Learn.*, New York, NY, USA, 2006, pp. 905–912.

[35] D. Wang, F. Nie, and H. Huang, "Feature selection via global redundancy minimization," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 10, pp. 2743–2755, Oct. 2015.

[36] T. Wu, A. R. Benson, and D. F. Gleich, "General tensor spectral co-clustering for higher-order data," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2016, pp. 2559–2567.

[37] K. Yan and D. Zhang, "Feature selection and analysis on correlated gas sensor data with recursive feature elimination," *Sens. Actuators B, Chem.*, vol. 212, pp. 353–363, Jun. 2015.

[38] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

RENJIE CHEN received the B.S. degree in software engineering from the South China University of Technology, China, where he is currently pursuing the master's degree with the School of Software Engineering. His current research interests include clustering and feature selection.

NING SUN received the B.A. degree from the School of Information Science and Technology, University of Science and Technology of China, in 2004, the M.S. degree from the Engineering School, Columbia University, in 2005, and the M.B.A. degree from the Business School, The Hong Kong University of Science and Technology, in 2013. Her research interests include data mining and information systems.

XIAOJUN CHEN received the Ph.D. degree from the Harbin Institute of Technology in 2011. He is currently an Assistant Professor with the College of Computer Science and Software, Shenzhen University. His research interests include clustering, feature selection, and massive data mining.

MIN YANG received the Ph.D. degree from the Department of Computer Science, The University of Hong Kong. She is currently an Assistant Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, leading the Artificial Intelligence and Natural Language Processing team, Global Center for Big Mobile Intelligence. Her research interests include a variety of topics, including natural language processing, multimodal learning, deep learning, recommendation systems, and machine learning in finance.

QINGYAO WU received the B.S. degree in software engineering from the South China University of Technology, and the M.S. and Ph.D. degrees in computer science from the Harbin Institute of Technology, China, in 2007, 2009, and 2013, respectively. He is currently an Associate Professor with the School of Software Engineering, South China University of Technology, China. He was a Post-Doctoral Research Fellow with the School of Computer Engineering, Nanyang Technological University, Singapore, from 2014 to 2015. His current research interests include machine learning, data mining, and big data learning.