Semisupervised Feature Selection via Generalized Uncorrelated Constraint and Manifold Embedding

Xuelong Li, *Fellow, IEEE*, Yunxing Zhang, and Rui Zhang, *Member, IEEE*

Abstract—**Ridge regression is frequently utilized by both supervised learning and semisupervised learning. However, the results cannot obtain the closed-form solution and perform manifold structure when ridge regression is directly applied to semisupervised learning. To address this issue, we propose a novel semisupervised feature selection method under generalized uncorrelated constraint, namely SFS. The generalized uncorrelated constraint equips the framework with the elegant closed-form solution and is introduced to the ridge regression with embedding the manifold structure. The manifold structure and closed-form solution can better save data's topology information compared to the deep network with gradient descent. Furthermore, the full rank constraint of the projection matrix also avoids the occurrence of excessive row sparsity. The scale factor of the constraint that can be adaptively obtained also provides the subspace constraint more flexibility. Experimental results on data sets validate the superiority of our method to the state-of-the-art semisupervised feature selection methods.**

Index Terms—**Feature selection, manifold embedding, semisupervised learning, uncorrelated constraint.**

I. Introduction

IN MANY applications such as computer vision and face recognition, the dimension of data is getting higher and higher, while the obtaining of labeled data is becoming very limited [1], [2]. Directly processing such high-dimensional data not only degrades its performance but also is time-consuming. By selecting the most discriminative feature subset from the original data, feature selection improves the interpretability of the data, which is becoming one of the most important methods to deal with high-dimensional data [3]–[5]. By retaining the original features, feature selection can improve the interpretability of the data, which is preferred in many real applications, such as face recognition [6], object recognition [7], and video semantic recognition [8].

By selecting the most discriminative feature subset from the original data, feature selection improves the interpretability of the data, which is becoming one of the most important methods to deal with high-dimensional data [9], [10]. Generally speaking, the existing feature selection methods can roughly fall into three categories: filter-based methods [11], wrapper-based methods [12], and embedding-based methods [13]. The filter-based methods score the features with a ranking, and the feature selection process is independent of the classifier. Wrapper methods take the performance of the classifier to be used as the evaluation criterion of the feature subset. Embedded methods integrate the feature selection process and the classifier training process into one optimization process.

As described in [14], filter-based feature selection methods may discard important features that are less informative but feature-rich when combined with other features. There is also some wrapper-type forward feature selection framework, which is usually time-consuming for high-dimensional data because it involves iterative feature subset searching. Making feature selection as part of the training process, embedded feature selection methods are better than other methods in many respects.

Depending on the availability of labels, feature selection methods can be roughly divided into three categories: supervised [13], semisupervised [9], [15], and unsupervised feature selection [16], [17]. Supervised feature selection method determines feature importance by evaluating the feature's correlation with the class labels, semisupervised feature selection uses both (small) labeled data and (large) unlabeled data, and unsupervised feature selection exploits the most discriminating features without any class labels. By utilizing the whole label of data, some supervised feature selection methods have gained excellent performance. However, the use of whole labeled data makes those methods very expensive and time-consuming. Benefitted from requiring less human effort and giving higher accuracy, semisupervised feature selection methods have been greatly developed recently.

Ridge regression is a very important technique in supervised and semisupervised learning and has been widely used in the deep network field [18]. In addition, by sharing a similar least-square form with k-means, the traditional ridge regression is not suitable for unsupervised learning due to the trivial solution it will trigger. Although the traditional ridge regression can be used in semisupervised learning, the framework cannot obtain closed-form solutions and perform manifold structure. Furthermore, most existing literature solves the ridge regression on semisupervised learning by gradient descent. However, those methods cannot better save data's topology information because the framework cannot perform manifold structure.

Manuscript received October 3, 2019; revised July 17, 2020 and November 13, 2020; accepted March 23, 2021. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0102201 and in part by the National Natural Science Foundation of China under Grant 61871470, Grant U1801262, and Grant 61761130079. *(Corresponding author: Rui Zhang.)*

The authors are with the School of Computer Science and the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: xuelong_li@nwpu.edu.cn; zhangyunxing423@outlook.com; ruizhang8633@gmail.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3069038.

Digital Object Identifier 10.1109/TNNLS.2021.3069038

To address this issue, we propose an effective method (SFS) that extends ridge regression to semisupervised feature selection under the generalized uncorrelated constraint. The generalized uncorrelated constraint not only equips the model with an elegant closed-form solution but also avoids excessive row sparsity of the projection matrix. Besides that, the side information that contains the prior links information makes the Laplacian structure more accurate, which will be introduced in detail in Section II. The main contributions of our work are given in the following.

1) The generalized uncorrelated constraint is introduced to the ridge regression with embedding the manifold structure, where the topology structure of data is preserved during the optimization.
2) The generalized uncorrelated constraint equips the framework with an elegant closed-form solution when the ridge regression is applied to semisupervised learning.
3) The transformation matrix $\mathbf{Z} \in \mathbb{R}^{d\times c}$ in the generalized uncorrelated constraint is row sparse and column full rank, which guarantees that there are at least c nonzero rows and avoids the occurrence of excessive row sparsity.
4) The scale factor α that can be adaptively obtained is added to the framework, which makes the problem more general, and the subspace scale more flexible. Besides that, the side information that contains the prior pairwise connections makes the Laplacian structure more accurate.

II. Related Work

In this section, we will introduce some literature similar to our work and explain the differences between these methods and our method. We will not only explain in the aspect of semisupervised features selection but also from the perspective of the uncorrelated constraint.

Many semisupervised feature selection algorithms have been proposed to exploit both labeled and unlabeled data in the past decade. To conduct the semisupervised feature selection on large-scale data sets, Chang *et al.* [19] proposed a novel convex semisupervised feature selection (CSFS) algorithm, which can be applied to large-scale data sets. Chen *et al.* [15] proposed a semisupervised feature selection method (RLSR) that can learn both global and sparse solutions of the projection matrix. In the video processing field, inspired by abundant unlabeled videos, Han *et al.* [20] proposed a framework of video semantic recognition by semisupervised feature selection via spline regression (SFSR). Yu *et al.* [21] proposed the semisupervised model (ASFS) that updates the mapping matrices and the label matrix for unlabeled data simultaneously and iteratively. Luo *et al.* [22] proposed a novel semisupervised feature selection method under insensitive sparse regression (ISR). Chen *et al.* [23] proposed a novel semisupervised embedded feature selection method (SRLSR), which extends the least-square regression model by rescaling the regression coefficients in the least-square regression with a set of scale factors.

In these semisupervised feature selection methods, several methods, such as CSFS, SFSR, and SRLSR, also use ridge regression. However, these methods cannot perform the manifold structure where the structure of data remains during the optimization of the model. Equipped with the generalized uncorrelated constraint, our framework can perform the manifold structure while obtaining the elegant closed-form solution.

There is also some literature about the uncorrelated constraint. Li *et al.* [24] presented an improved sparse regression model (GURM) for seeking the uncorrelated yet discriminative features. Zhang *et al.* [25] investigated unsupervised feature selection by virtue of an uncorrelated and nonnegative ridge regression model and propose the method (UN-RFS), which imposes a nonnegative orthogonal constraint on the indicator matrix. Li *et al.* [26] proposed an unsupervised regularized regression model (DUCFS) that explores the low-redundant and discriminative features by the generalized uncorrelated constraint.

However, these frameworks are only restricted to unsupervised learning. Our method not only extends generalized uncorrelated constraint to the semisupervised learning but also equips the subspace with the scale factor, which makes the problem more general and the subspace scale more flexible. Besides that, the projection matrix $\mathbf{Z} \in \mathbb{R}^{d\times c}$ in our framework is row sparse and column full rank, which guarantees that there are at least c nonzero rows and avoids the occurrence of excessive row sparsity.

III. Methodology

In this section, we present a novel semisupervised feature selection method SFS. Also, we will expound on the ideas and principles of the method in detail.

A. Notations

In this article, matrices are written as boldface capital letters, and vectors are denoted as boldface lowercase letters. If $\mathbf{w}_i$ as the ith row of the matrix $\mathbf{W} = [\mathbf{w}_{ij}] \in \mathbb{R}^{d\times c}$, then the F-norm of $\mathbf{W}$ is defined as $\|\mathbf{W}\|_F = (\sum_{i=1}^{d} \sum_{j=1}^{c} \mathbf{w}_{ij}^2)^{1/2} = (\sum_{i=1}^{d} \|\mathbf{w}_i\|_2^2)^{1/2}$. The $\ell_{2,1}$ norm of $\mathbf{W}$ is defined as $\|\mathbf{W}\|_{2,1} = \sum_{i=1}^{d}(\sum_{j=1}^{c} \mathbf{w}_{ij}^2)^{1/2} = \sum_{i=1}^{d} \|\mathbf{w}_i\|_2$, where $\|\mathbf{w}_i\|_2$ denotes the ℓ_2-norm of the vector $\mathbf{w}_i$.

In the semisupervised learning, let us denote $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ as the centralized total data matrix with dimension d and data number n. $\mathbf{X}$ can be divided into labeled data $\mathbf{X}_l \in \mathbb{R}^{d\times n_l}$ and unlabeled data $\mathbf{X}_u \in \mathbb{R}^{d\times n_u}$, i.e., $\mathbf{X} = [\mathbf{X}_l, \mathbf{X}_u] \in \mathbb{R}^{d\times n}$ where $n = n_l + n_u$. The global indication matrix $\mathbf{F}$ can be divided as the labeled part $\mathbf{F}_u$ and unlabeled part $\mathbf{F}_l$. $\mathbf{F}_l \in \mathbb{R}^{n_l\times c}$ denotes the indication matrix of the labeled data, $\mathbf{F}_u \in \mathbb{R}^{n_u\times c}$ denotes the pseudo indication matrix of the unlabeled data, i.e., $\mathbf{F} = [\mathbf{F}_l; \mathbf{F}_u] \in \mathbb{R}^{n\times c}$. $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}^{\mathrm{T}} \in \{0, 1\}^{n\times c}$ is the pre-given ground truth with satisfying $\mathbf{Y}\mathbf{1}_c = \mathbf{1}_n$, where $\mathbf{1}_c$ and $\mathbf{1}_n$ are unit column vector with dimension c and n. $\mathbf{C} = \{1, 2, \ldots, c\}$ is the number of categories.

B. When Ridge Regression Is Directly Applied to Semisupervised Learning

As a basic technology, ridge regression is widely used in machine learning and deep learning. To classify the original sample $\mathbf{X}$ into c cluster, classical supervised ridge regression can be expressed as

$$\min_{\mathbf{W}} \left\| \mathbf{X}^{\mathrm{T}}\mathbf{W} - \mathbf{Y} \right\|_F^2 + \lambda \|\mathbf{W}\|_F^2 \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the projection matrix and $\lambda \in \mathbb{R}$ is the regularization parameter. Auxiliary with label information, classical ridge regression has achieved good performance in such supervised problem. Also, this framework has been used widely in the deep network field.

However, when the classical ridge regression is directly applied to semisupervised learning, the label information $\mathbf{Y}$ becomes an incompletely known variable. In that case, many existing methods solve this problem with gradient descent. Nevertheless, these frameworks cannot obtain the closed-form solution and perform manifold structure under gradient descent. To address this issue, we propose a novel method, namely *SFS*. We will explain in detail how our method can perform manifold structure and obtain the closed-form solution.

C. Ridge Regression and Uncorrelated Constraint

To equip the result with the closed-form solution, it is natural to constrain subspace $\mathbf{W}$ to be full rank, i.e., $\mathrm{rank}(\mathbf{W}) = c$. To achieve this full rank constraint, there are usually two general options: orthogonal constraint $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ and uncorrelated constraint $\mathbf{W}^{\mathrm{T}}\mathbf{S}_t\mathbf{W} = \mathbf{I}$, where $\mathbf{S}_t = \mathbf{X}\mathbf{X}^{\mathrm{T}}$ is the total scatter matrix.

Both of them are equipped with the manifold structure where the structure of data remains during the optimization of the model. In this article, we choose the uncorrelated constraint over the orthogonal constraint due to the fact that the uncorrelated constraint can explore the most uncorrelated data in the subspace. In addition, we embed scale factor α into the subspace constraint as $\mathbf{W}^{\mathrm{T}}\mathbf{S}_t\mathbf{W} = (1/\alpha^2)\mathbf{I}$ to make the problem more general and provide more flexibility to the subspace constraint. Then, problem (1) can be extended to

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{F}} \left\| \mathbf{X}^{\mathrm{T}}\mathbf{W} - \mathbf{F} \right\|_F^2 + \lambda \|\mathbf{W}\|_F^2 \\ &\text{s.t. } \mathbf{W}^{\mathrm{T}}\mathbf{S}_t\mathbf{W} = \frac{1}{\alpha^2}\mathbf{I}, \quad \mathbf{F} \geq \mathbf{0}_{\mathrm{nc}} \end{aligned} \tag{2}$$

where $\mathbf{F}$ is the indicator matrix that needs to be solved.

Note that the uncorrelated constraint $\mathbf{W}^{\mathrm{T}}\mathbf{S}_t\mathbf{W} = (1/\alpha^2)\mathbf{I}$ in problem (2) can be rewritten as $((1/\alpha)\mathbf{Z}^{\mathrm{T}})\mathbf{S_t}((1/\alpha)\mathbf{Z}) = (1/\alpha^2)\mathbf{I} \Rightarrow \mathbf{Z}^{\mathrm{T}}\mathbf{S_t}\mathbf{Z} = \mathbf{I}$ under $\mathbf{Z} = \alpha\mathbf{W}$. Then, problem (2) can be rescaled into the following equivalent counterpart:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{F},\alpha} \left\| \mathbf{X}^{\mathrm{T}}\mathbf{Z} - \alpha\mathbf{F} \right\|_F^2 + \lambda \|\mathbf{Z}\|_F^2 \\ &\text{s.t. } \mathbf{Z}^{\mathrm{T}}\mathbf{S}_t\mathbf{Z} = \mathbf{I}, \quad \mathbf{F} \geq \mathbf{0}_{nc}. \end{aligned} \tag{3}$$

Problem (3) is the rescaled dual problem of problem (2) and is the base of Section III-D.

D. Proposed Framework With Generalized Uncorrelated Constraint

As mentioned above, we apply the ridge regression extended by the uncorrelated constraint (3) to the semisupervised learning as follows:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{F}_u,\alpha} \left\| \left[\mathbf{X}_l, \mathbf{X}_u\right]^{\mathrm{T}}\mathbf{Z} - \alpha\left[\mathbf{F}_l; \mathbf{F}_u\right] \right\|_F^2 + \lambda \|\mathbf{Z}\|_F^2 \\ &\text{s.t. } \mathbf{Z}^{\mathrm{T}}\mathbf{S}_t\mathbf{Z} = \mathbf{I}. \end{aligned} \tag{4}$$

In problem (4), λ is the regularization parameter, which can determine the degree of sparsity of $\mathbf{Z}$.

To connect similar samples with larger weight in subspace, the Laplacian construction is embedded in the problem (4). Also, for better performance on the semisupervised feature selection task, F-norm is replaced by a nonconcave $\ell_{2,1}$-norm to obtain a row-sparse matrix $\mathbf{Z}$. In the embedded feature selection methods, a row-sparse matrix $\mathbf{Z}$ can serve as the evaluation criterion for selecting features. Also, $\|\mathbf{z}_i\|_2$ is the score of the ith feature [24]. Then, problem (4) can be extended as

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{F}_u,\alpha} \left\| \left[\mathbf{X}_l, \mathbf{X}_u\right]^{\mathrm{T}}\mathbf{Z} - \alpha\left[\mathbf{F}_l; \mathbf{F}_u\right] \right\|_F^2 + \beta\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}\mathbf{F}\right) + \lambda \|\mathbf{Z}\|_{2,1} \\ &\text{s.t. } \mathbf{Z}^{\mathrm{T}}\mathbf{S}_t\mathbf{Z} = \mathbf{I} \end{aligned} \tag{5}$$

where β is a parameter and $\mathbf{L}$ is the Laplacian matrix.

Furthermore, we replace the uncorrelated constraint $\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t\mathbf{Z} = \mathbf{I}$ in problem (5) by the generalized uncorrelated constraint $\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z} = \mathbf{I}$. Also, $\mathbf{S}_t^{(p)}$ is defined as $(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P})$, where $\mathbf{P}$ is a $d \times d$ diagonal matrix whose diagonal element p_{ii} can be defined as

$$p_{\mathrm{i\,i}} = \frac{1}{2\sqrt{\|\mathbf{z}_i\|_2^2 + \varepsilon}} \quad (\varepsilon \to 0, i = 1, 2, \ldots, d). \tag{6}$$

In (6), $\varepsilon \to 0$ is a small value added for preventing the situation that denominator may become zero due to the row sparsity of $\mathbf{Z}$.

The final objective function to be optimized in our framework is

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{F}_u,\alpha} \left\| \left[\mathbf{X}_l, \mathbf{X}_u\right]^{\mathrm{T}}\mathbf{Z} - \alpha\left[\mathbf{F}_l; \mathbf{F}_u\right] \right\|_F^2 + \beta\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}\mathbf{F}\right) + \lambda \|\mathbf{Z}\|_{2,1} \\ &\text{s.t. } \mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z} = \mathbf{I}. \end{aligned} \tag{7}$$

There are three benefits to replace the uncorrelated constraint with the generalized uncorrelated constraint. First, the total scatter matrix $\mathbf{S}_t^{(p)}$ is also positive semidefinite when the number of samples is less than features ($n < d$) [27]. Second, by adding a diagonal matrix $\mathbf{P}$ into $\mathbf{S}_t$, the generalized uncorrelated constraint relaxes the rigid constraint aforementioned and avoids loss of the ability to discriminate against the samples. Third, the proposed generalized uncorrelated constraint $\mathbf{S}_t^{(p)}$ simplifies the optimization of the model (7) and equips an elegant closed-form solution for $\mathbf{Z}$, which is introduced to the ridge regression by embedding the manifold structure. This is also the most important contribution of the generalized uncorrelated constraint in this article.

In addition, when the rank of $\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z}$ is c, and according to theorem 1, the rank of $\mathbf{Z}$ is also c. This full rank constraint can avoid the occurrence of excessive sparseness in the solution process.

Theorem 1: **If rank**$(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z}) = c$, then **rank**$(\mathbf{Z}) = c$.

Proof:

$$\begin{aligned}\mathbf{rank}\Big(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z}\Big) &= \mathbf{rank}\bigg(\Big((\mathbf{S}_t^{(p)})^{\frac{1}{2}}\mathbf{Z}\Big)^{\mathrm{T}}(\mathbf{S}_t^{(p)})^{\frac{1}{2}}\mathbf{Z}\bigg)\\ &= \mathbf{rank}\Big((\mathbf{S}_t^{(p)})^{\frac{1}{2}}\mathbf{Z}\Big)\\ &= \mathbf{rank}(\mathbf{Z})\\ &= c. \end{aligned}\tag{8}$$

□

E. ML, CL, and Side Information

The second term of the objective function (7) is based on the Laplacian construction $\mathbf{L}$, which is calculated as $\mathbf{L} = \mathbf{D} - \mathbf{S}$, where $\mathbf{S}$ is the similarity matrix and $\mathbf{D}$ is the degree matrix. The performance of the graph-based approach is affected by the similarity matrix $\mathbf{S}$. There are several ways to construct the similarity matrix, such as LTSA [28], LSE [29], KNN [30], CAN [31], and Gaussian kernel. In this article, we use a more general method KNN to construct the similarity matrix, and the ablation experiment in Fig. 5 shows that our method can perform better if we use a more refined similarity matrix construction way like CAN.

The real-world data are usually generated along with prior information, i.e., side information, which are categorized into must-link (ML) and cannot-link (CL) constraints. ML constraint represents that two data points x_i, x_j are in the same cluster, whereas the CL constraint means that two data points x_i, x_j are not in the same cluster. By incorporating the side information, similarity matrix $\mathbf{S}$ can be further extended to $\widetilde{\mathbf{S}}$ whose element is defined as

$$\tilde{s}_{\mathrm{i\,j}} = \begin{cases} 1, & \text{must–link}\big(\mathbf{x}_i, \mathbf{x}_j\big)\\ 0, & \text{cannot–link}\big(\mathbf{x}_i, \mathbf{x}_j\big)\\ s_{\mathrm{i\,j}}, & \text{otherwise}\ . \end{cases}\tag{9}$$

The side information added in (9) makes the similarity matrix more accurate since it carries correct prior links information. Then, the Laplacian matrix can be calculated as

$$\mathbf{L} = \mathbf{D} - \big(\widetilde{\mathbf{S}}^{\mathrm{T}} + \widetilde{\mathbf{S}}\big)/2 \tag{10}$$

where the degree matrix $\mathbf{D}$ is the diagonal matrix whose ith diagonal element is $\sum_j (\tilde{s}_{\mathrm{j\,i}} + \tilde{s}_{\mathrm{i\,j}})/2$.

IV. Optimization Procedure

The third item of problem (7) is the $\ell_{2,1}$ norm of $\mathbf{Z}$. To facilitate the calculation, we make the $\ell_{2,1}$ norm of $\mathbf{Z}$ into a matrix form by the diagonal matrix $\mathbf{P}$ in (6). Hence, solving SFS in (7) is equivalent to solving the following question:

$$\begin{aligned}\min_{\mathbf{Z},\mathbf{F}_u,\alpha} &\Big\|\big[\mathbf{X}_l, \mathbf{X}_u\big]^{\mathrm{T}}\mathbf{Z} - \alpha\big[\mathbf{F}_l; \mathbf{F}_u\big]\Big\|_F^2 + \beta\mathrm{Tr}\big(\mathbf{F}^{\mathrm{T}}\mathbf{L}\mathbf{F}\big)\\ &\quad + \lambda\mathrm{Tr}\big(\mathbf{Z}^{\mathrm{T}}\mathbf{P}\mathbf{Z}\big)\\ \text{s.t. } &\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z} = \mathbf{I}. \end{aligned}\tag{11}$$

There are three variables $\mathbf{Z}$, $\mathbf{F}_u$, and α in the optimization of problem (11). Hereinafter, the coordinate blocking method, i.e., alternating method, is employed.

A. Optimize $\mathbf{Z}$ With Fixing $\mathbf{F}_u$ and α

When $\mathbf{F}_u$ and α are fixed, problem (11) satisfies the following deduction:

$$\min_{\mathbf{Z}}\ \mathrm{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{Z} - \alpha\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F} - \alpha\mathbf{F}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{Z}) + \lambda\mathrm{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{P}\mathbf{Z}). \tag{12}$$

In problem (12), as the variables $\mathbf{F}$ and α are fixed and $\mathbf{Z}$ satisfies the constraint $\mathbf{Z}^{\mathrm{T}}(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P})\mathbf{Z} = \mathbf{I}$, therefore the solution to $\mathbf{Z}$ can be simplified as

$$\mathbf{Z} = \underset{\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z}=\mathbf{I}}{\mathrm{argmin}}\ -2\alpha\ \mathrm{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F}) \Leftrightarrow \underset{\mathbf{Z}^{\mathrm{T}}\mathbf{S}_t^{(p)}\mathbf{Z}=\mathbf{I}}{\mathrm{argmax}}\ \mathrm{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F}). \tag{13}$$

One contribution of the generalized uncorrelated constraint is to equip the problem with an elegant closed-form solution. Equation (13) can be converted into the following form:

$$\max_{\mathbf{M}^{\mathrm{T}}\mathbf{M}=\mathbf{I}} \mathrm{Tr}\big(\mathbf{M}^{\mathrm{T}}\mathbf{N}\big) \tag{14}$$

where

$$\mathbf{M} = \big(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P}\big)^{\frac{1}{2}}\mathbf{Z}, \quad \mathbf{N} = \big(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P}\big)^{-\frac{1}{2}}\mathbf{X}\mathbf{F}. \tag{15}$$

According to (15), the optimal solution of $\mathbf{Z}$ can be obtained by solving $\mathbf{M}$ as

$$\mathbf{Z} = \big(\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P}\big)^{-\frac{1}{2}}\mathbf{M} \tag{16}$$

where $\mathbf{M}$ can be obtained by performing singular value decomposition (SVD) on $\mathbf{N}$ according to Lemma (1). Huang *et al.* [32] provided a clear and powerful proof to Lemma (1). In (16), the matrix $\mathbf{P}$ also depends on $\mathbf{Z}$. Therefore, an iterative algorithm to obtain the optimal solution $\mathbf{Z}$ for problem (11) is presented in Algorithm 1.

Lemma 1: The optimal solution $\mathbf{M}$ to problem (14) is defined as

$$\mathbf{M} = \mathbf{U}\mathbf{V}^{\mathrm{T}} \tag{17}$$

where $\mathbf{U}$, $\mathbf{V}$ are left and right singular matrices of compact SVD decomposition of $\mathbf{N}$ defined in (15), respectively.

Algorithm 1 Algorithm to Solve Problem (16)

```
Input: The labeled data $\mathbf{X}_l \in \mathbb{R}^{d\times n_l}$, the unlabeled data
    $\mathbf{X}_u \in \mathbb{R}^{d\times n_u}$, the indicator matrix $\mathbf{F}_l \in \mathbb{R}^{n_l\times c}$ of the labeled
    samples, the indicator matrix $\mathbf{F}_u \in \mathbb{R}^{n_u\times c}$ of the unlabeled
    samples, the Laplacian matrix $\mathbf{L}$, and the parameter $\lambda$.
    Initialize Unit diagonal matrix $\mathbf{P} = \mathbf{I}$.
    Repeat:
    With current $\mathbf{X}$, $\mathbf{P}$ and $\mathbf{F} = [\mathbf{F}_l; \mathbf{F}_u]$, calculate $\mathbf{N}$ with
    definition in Eq. (15).
    Calculate $\mathbf{M}$ and $\mathbf{N}$ via the compact SVD decomposition
    of $\mathbf{N}$.
    Update $\mathbf{Z} \leftarrow (\mathbf{X}\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{P})^{-\frac{1}{2}}\mathbf{M}$.
    Update
    $\mathbf{P} = \mathrm{diag}(\frac{1}{2\sqrt{\|\mathbf{z}_1\|_2^2+\varepsilon}}, \frac{1}{2\sqrt{\|\mathbf{z}_2\|_2^2+\varepsilon}}, \ldots, \frac{1}{2\sqrt{\|\mathbf{z}_d\|_2^2+\varepsilon}})$.
    Until convergence
Output: The projection matrix $\mathbf{Z} \in \mathbb{R}^{d\times c}$.
```

B. Optimize $\mathbf{F}_u$ *With Fixing* $\mathbf{Z}$ *and* α

The total sample $\mathbf{X}$ can be divided into labeled data $\mathbf{X}_l$ and unlabeled data $\mathbf{X}_u$, i.e., $\mathbf{X} = [\mathbf{X}_l, \mathbf{X}_u] \in \mathbb{R}^{d\times n}$. Similarly, we can rewrite the Laplacian matrix $\mathbf{L}$ as a partitioned matrix: $\mathbf{L} = \begin{bmatrix} \mathbf{L}_{ll} & \mathbf{L}_{lu} \\ \mathbf{L}_{ul} & \mathbf{L}_{uu} \end{bmatrix}$, where $\mathbf{L}_{ll} \in \mathbb{R}^{n_l\times n_l}$, $\mathbf{L}_{lu} \in \mathbb{R}^{n_l\times n_u}$, $\mathbf{L}_{ul} \in \mathbb{R}^{n_u\times n_l}$, and $\mathbf{L}_{uu} \in \mathbb{R}^{n_u\times n_u}$ are block matrix representations for $\mathbf{L}$.

Then, the solution of $\mathbf{F}_u$ in problem (11) can be rewritten as

$$\min_{\mathbf{F}_u} \mathrm{Tr}\Big(\alpha^2\mathbf{F}_u^{\mathrm{T}}\mathbf{F}_u - 2\alpha\mathbf{F}_u^{\mathrm{T}}\mathbf{X}_u^{\mathrm{T}}\mathbf{Z} + \beta\mathbf{F}_u^{\mathrm{T}}\mathbf{L}_{uu}\mathbf{F}_u + 2\beta\mathbf{F}_u^{\mathrm{T}}\mathbf{L}_{ul}\mathbf{F}_l\Big). \tag{18}$$

Since there is no constraint on $\mathbf{F}_u$ in problem (18), the optimization of $\mathbf{F}_u$ is equal to solving the following problem:

$$\begin{aligned} &\frac{\partial}{\partial\mathbf{F}_u}\mathrm{Tr}\Big(\alpha^2\mathbf{F}_u^{\mathrm{T}}\mathbf{F}_u - 2\alpha\mathbf{F}_u^{\mathrm{T}}\mathbf{X}_u^{\mathrm{T}}\mathbf{Z} + \beta\mathbf{F}_u^{\mathrm{T}}\mathbf{L}_{uu}\mathbf{F}_u + 2\beta\mathbf{F}_u^{\mathrm{T}}\mathbf{L}_{ul}\mathbf{F}_l\Big) = \mathbf{0} \\ &\Rightarrow \mathbf{F}_u = \big(\alpha^2\mathbf{I} + \beta\mathbf{L}_{uu}\big)^{-1}\big(\alpha\mathbf{X}_u^{\mathrm{T}}\mathbf{Z} - \beta\mathbf{L}_{ul}\mathbf{F}_l\big). \end{aligned} \tag{19}$$

C. Optimize α *With Fixing* $\mathbf{Z}$ *and* $\mathbf{F}_u$

When $\mathbf{Z}$ and $\mathbf{F}_u$ are fixed, problem (11) can be rewritten as

$$\min_{\alpha} \ \alpha^2\mathrm{Tr}\big(\mathbf{F}^{\mathrm{T}}\mathbf{F}\big) - 2\alpha\mathrm{Tr}\big(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F}\big) \tag{20}$$

which leads to the solution

$$\frac{\partial\big(\alpha^2\mathrm{Tr}\big(\mathbf{F}^{\mathrm{T}}\mathbf{F}\big) - 2\alpha\mathrm{Tr}\big(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F}\big)\big)}{\partial\alpha} = 0 \Rightarrow \alpha = \frac{\mathrm{Tr}\big(\mathbf{Z}^{\mathrm{T}}\mathbf{X}\mathbf{F}\big)}{\mathrm{Tr}\big(\mathbf{F}^{\mathrm{T}}\mathbf{F}\big)}. \tag{21}$$

With the alternating optimization of $\mathbf{W}$, $\mathbf{F}_u$, and α. Algorithm 2 summarizes the proposed SFS (7).

```
Algorithm 2 Algorithm to SFS (11)
Input: The labeled data $\mathbf{X}_l \in \mathbb{R}^{d\times n_l}$, the unlabeled data
    $\mathbf{X}_u \in \mathbb{R}^{d\times n_u}$, the indicator matrix $\mathbf{F}_l \in \mathbb{R}^{n_l\times c}$ of the labeled
    samples, the Laplacian matrix $\mathbf{L}$, the number of selected
    feature $f$, the coefficients $\beta$ and $\lambda$.
    Initialize Random matrix $\mathbf{F}_u \in \mathbb{R}^{n_u\times c}$, parameter $\alpha = 1$.
    Repeat:
    Update $\mathbf{Z}$ by Algorithm 1.
    Update $\mathbf{F}_u$ by Eq. (19).
    Update $\alpha$ by Eq. (21).
    until convergence
Output: $\mathbf{F}_u$; calculate and sort $\|\mathbf{z}_i\|_2 (i = 1, 2, \ldots, d)$ in the
    descending order, then select the top $f$ ranked features as
    the results of feature selection.
```

D. Convergence of Algorithm

Algorithm 1 solves problem (11) with locally optimal solution of $\mathbf{Z}$. In order to demonstrate this convergence, Lemma 2 is utilized subsequently, which is proposed and proved in [33].

Lemma 2: For any nonzero vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^c$, it holds that

$$\|\mathbf{u}\|_2 - \frac{\|\mathbf{u}\|_2^2}{2\|\mathbf{v}\|_2} \le \|\mathbf{v}\|_2 - \frac{\|\mathbf{v}\|_2^2}{2\|\mathbf{v}\|_2}. \tag{22}$$

Theorem 2: Algorithm 1 decreases problem (11) by iteratively updating Z with its optimal solution to problem (7) until convergence.

Proof: Denote the objective value in the tth iteration of problem (11) as $\mathcal{Q}(\mathbf{Z}^{(t)}, \mathbf{P}^{(t)})$, i.e.,

$$\mathcal{Q}(\mathbf{Z}^{(t)}, \mathbf{P}^{(t)}) = \|(\mathbf{X}^{\mathrm{T}}\mathbf{Z}^{(t)} - \alpha\mathbf{F})\|_F^2 + \lambda\mathrm{Tr}((\mathbf{Z}^{(t)})^{\mathrm{T}}\mathbf{P}^{(t)}\mathbf{Z}^{(t)}). \tag{23}$$

Since Algorithm 1 updates $\mathbf{Z}$ and $\mathbf{P}$ with the optimal solution to problem (11) in each iteration, for the $(t+1)$th iteration, it must hold

$$\mathcal{Q}(\mathbf{Z}^{(t+1)}, \mathbf{P}^{(t)}) \le \mathcal{Q}(\mathbf{Z}^{(t)}, \mathbf{P}^{(t)}). \tag{24}$$

According to Lemma 1, we have

$$\begin{aligned} &\Big\|\mathbf{z}_i^{(t+1)}\Big\|_2 - \frac{1}{2\Big\|\mathbf{z}_i^{(t)}\Big\|_2}\Big\|\mathbf{z}_i^{(t+1)}\Big\|_2^2 \\ &\le \Big\|\mathbf{z}_i^{(t)}\Big\|_2 - \frac{1}{2\Big\|\mathbf{z}_i^{(t)}\Big\|_2}\Big\|\mathbf{z}_i^{(t)}\Big\|_2^2 \\ &\Rightarrow \ \lambda\sum_{i=1}^{d}\Big\|\mathbf{z}_i^{(t+1)}\Big\|_2 - \lambda\sum_{i=1}^{d}\frac{1}{2\Big\|\mathbf{z}_i^{(t)}\Big\|_2}\Big\|\mathbf{z}_i^{(t+1)}\Big\|_2^2 \\ &\le \lambda\sum_{i=1}^{d}\Big\|\mathbf{z}_i^{(t)}\Big\|_2 - \lambda\sum_{i=1}^{d}\frac{1}{2\Big\|\mathbf{z}_i^{(t)}\Big\|_2}\Big\|\mathbf{z}_i^{(t)}\Big\|_2^2 \\ &\Rightarrow \ \lambda\Big\|\mathbf{Z}^{(t+1)}\Big\|_{2,1} - \lambda\mathrm{Tr}((\mathbf{Z}^{(t+1)})^{\mathrm{T}}\mathbf{P}^{(t)}\mathbf{Z}^{(t+1)}) \\ &\le \lambda\Big\|\mathbf{Z}^{(t)}\Big\|_{2,1} - \lambda\mathrm{Tr}((\mathbf{Z}^{(t)})^{\mathrm{T}}\mathbf{P}^{(t)}\mathbf{Z}^{(t)}). \end{aligned} \tag{25}$$

Combining the result of deduction with (24), it can be inferred that

$$\begin{aligned} &\Big\|(\mathbf{X}^{\mathrm{T}}\mathbf{Z}^{(t+1)} - \alpha\mathbf{F})\Big\|_F^2 + \lambda\Big\|\mathbf{Z}^{(t+1)}\Big\|_{2,1} \\ &\quad\le \ \Big\|(\mathbf{X}^{\mathrm{T}}\mathbf{Z}^{(t)} - \alpha\mathbf{F})\Big\|_F^2 + \lambda\Big\|\mathbf{Z}^{(t)}\Big\|_{2,1}. \end{aligned} \tag{26}$$

Then, it is obvious that the objective value of problem (7) is decreased by Algorithm 1 in each iteration. □

V. Experiments

In this section, extensive experiments are conducted to verify the superiority and effectiveness of the proposed SFS method. We apply our method and other competitors on four data sets with the same experimental setup. The experimental results are presented in the form of tables and figures to verify the validity of the proposed method.

TABLE I
DETAILED INTRODUCTION TO DATA SETS

Datasets	Size(n)	Number of All Features(d)	Class (c)	The Percentage of Labeled Data (p)	Number of Selected Features (f)
COIL20	1440	1024	20	5%, 10%, 20%, 30%, 40%, 50%	{20,40,60,80,100,120,140,160,180,200}
UMIST	575	1024	20	5%, 10%, 20%, 30%, 40%, 50%	{20,40,60,80,100,120,140,160,180,200}
USPS	2007	256	10	5%, 10%, 20%, 30%, 40%, 50%	{20,40,60,80,100,120,140,160,180,200}
YALEB	16128	1024	28	5%, 10%, 20%, 30%, 40%, 50%	{20,40,60,80,100,120,140,160,180,200}

A. Data Sets

We conduct experiments on four different public data sets, including one object image data set, i.e., COIL20, two face image data sets, i.e., UMIST and YALEB, and a handwritten digital data set USPS. A detailed introduction to these data sets is shown in Table I. Note that the training data in semisupervised learning contains both labeled samples and unlabeled samples, which is different from supervised learning.

B. Compared Methods

To validate the advantage of the **SFS** method, we compare it with several state-of-the-art semisupervised feature selection methods, and all the methods (include ours) use the $\ell_{2,1}$ norm for sparsity. The other competitor can be summarized as follows.

1) The **CSFS** method is a feature selection method, in which joint feature selection with sparsity and semisupervised learning is combined into a single framework [19].
2) The **RRPC** method is a feature selection method, which can achieve a good balance between relevance and redundancy in semisupervised feature selection [34].
3) The **SFSR** method is a features selection method that combines two scatter matrices to capture both the discriminative information and the local geometry structure of labeled and unlabeled training videos [20].
4) The **SFSS** method aims to jointly select the most relevant features from all the data points by using a sparsity-based model and exploiting both labeled and unlabeled data to learn the manifold structure [3].
5) The **SRLSR** method extends the least-square regression by rescaling the regression coefficients in the least-square regression with a set of scale factors, which are used to rank the features [23].

C. Experiment Setup and Evaluation Metrics

To verify the effectiveness of our semisupervised feature selection method, we design comparative experiments in three aspects.

In the first comparative experiment part, we aim to verify the label prediction performance of semisupervised learning. Therefore, we compare the $\mathbf{F}_u$ (pseudo labels for unlabeled data $\mathbf{X}_u$) solved by each semisupervised method with the ground truth, the more accurate the prediction, the more effective the semisupervised learning. When the model is running, the percentage of labeled data is set to {5%, 10%, 20%, 30%, 40%, and 50%}. The *F1-score* is the evaluation indicator, which is the harmonic mean of precision and recall, and the experiment results are shown in Table II.

The second part of the comparative experiment is to verify the effectiveness of feature selection on the classification task. The first step is to select top f features with the highest scores of each data set by the semisupervised models where 30% of samples are set to the labeled samples. The top f features are selected and make up a feature subset matrix (i.e., downsampling of $\mathbf{X}$) $\overline{\mathbf{X}} \in \mathbb{R}^{f \times n}$ $(f < d)$. In the second step, we aim to verify the quality of the selected feature, so the data matrix $\overline{\mathbf{X}}$ is randomly divided into a training set (70%) and a test set (30%). For the selected features, we first perform tenfold cross validation selecting the best SVM model, and then, we test the selected SVM model on the test set. The baseline is implemented by training SVM on raw data matrix $\mathbf{X}$. The evaluation indicator in the classification task is accuracy (ACC), and the experiment results are shown in Fig. 1.

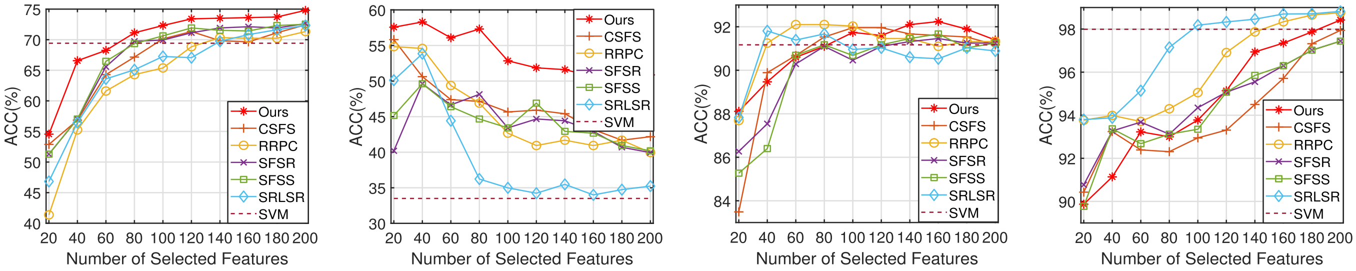
The third part of the comparative experiment is to verify the effectiveness of feature selection on the clustering task. Similar to the previous part, the first step is to select top f features with the highest scores of each data set by the semisupervised models where 30% of samples are set to the labeled samples. However, in the clustering task, we need not divide the data matrix $\overline{\mathbf{X}}$ into the train set and test set. For the selected features, we run the k-means algorithm on the whole samples $\overline{\mathbf{X}}$ for the final clustering results, while the k-means algorithm is also run on all features $\mathbf{X}$ as a baseline. In order to eliminate the error caused by the randomness of k-means, we run k-means ten times, and the experimental results take the mean. The evaluation indicator in the clustering task is normalized mutual information (NMI), and the experiment results are shown in Fig. 2.

D. Convergence Analysis and Running Time

The condition for iteration stop is set as $|\mathcal{J}_t - \mathcal{J}_{t+1}| \leq \varepsilon$, where $\mathcal{J}$ is the objective value of the proposed **SFS** in the model (7) and the threshold $\varepsilon = 10^{-4}$ and t represents the number of iterations. Fig. 6 shows the convergence of objective function value $\mathcal{J}$ on all data sets and the value of parameters λ and β is 1. The time complexity of our algorithm is $\mathcal{O}((d^3 + dn^2)t)$. The time complexity for other methods such as **CSFS** is $\mathcal{O}((d^3 + d^2n)t)$ and **SFSS** is $\mathcal{O}((n^3 + n^2)t)$. In addition, the time cost of our method

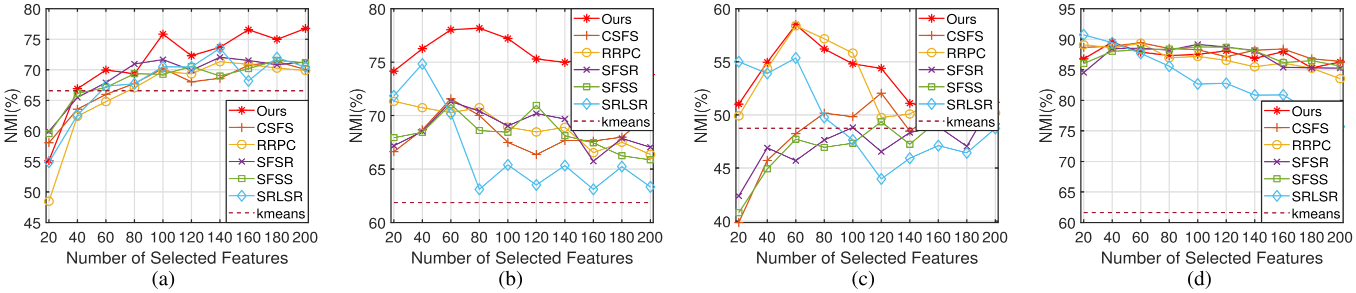
TABLE II
PERFORMANCE COMPARISON ON LABEL PREDICTION

Datasets	Percentage	Ours	CSFS	RRPC	SFSR	SFSS	SRLSR
COIL20	5%	**68.72**	43.81	44.82	39.27	41.73	43.61
	10%	**71.28**	50.55	47.41	47.91	48.19	47.20
	20%	**71.93**	61.75	59.81	62.16	62.08	60.20
	30%	**74.28**	66.97	66.73	68.69	68.70	66.94
	40%	**79.34**	71.56	75.81	75.20	75.33	76.12
	50%	**80.96**	74.96	76.14	74.97	75.30	76.39
UMIST	5%	**57.49**	44.09	41.15	37.50	37.44	41.22
	10%	**58.41**	46.50	49.48	48.47	47.86	49.50
	20%	**58.54**	50.13	52.30	47.70	48.00	52.14
	30%	**58.72**	47.42	52.12	43.14	43.39	51.75
	40%	**59.24**	47.66	54.32	46.81	46.84	53.18
	50%	**64.43**	43.98	54.49	43.81	44.91	52.90
USPS	5%	**87.33**	68.07	70.53	66.51	66.47	68.87
	10%	**89.47**	76.85	79.03	76.65	76.69	78.16
	20%	**89.72**	80.89	81.85	78.40	78.48	81.82
	30%	**91.30**	84.26	85.35	83.78	83.77	85.15
	40%	**91.73**	84.48	84.03	84.33	84.33	84.09
	50%	**92.06**	84.53	84.07	84.89	84.77	83.99
YALEB	5%	**94.11**	90.27	88.80	93.52	93.51	87.65
	10%	**96.63**	92.17	92.29	94.82	94.80	91.29
	20%	**98.88**	94.07	96.29	96.05	96.05	96.24
	30%	**99.59**	95.38	96.42	96.87	96.67	96.53
	40%	**99.74**	96.43	98.67	97.37	97.38	98.71
	50%	99.81	98.70	99.13	**99.88**	99.38	99.15



(a) (b) (c) (d)

Fig. 1. Performance comparison for feature selection on the classification task. (a) COIL20. (b) UMIST. (c) USPS. (d) YALEB.



(a) (b) (c) (d)

Fig. 2. Performance comparison for feature selection on the clustering task. (a) COIL20. (b) UMIST. (c) USPS. (d) YALEB.

on each data set is shown in Table III. Noting that all experiments are conducted in MATLAB R2018b, the codes are run on a Windows 10 machine with 1.60-GHz i5-8250U CPU, 8-GB main memory.

E. Parameter Sensitivity and Ablation Experiments

Parameter Settings: In this article, the optimal value of parameters β and λ is both set via the traversal search method in the range of $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$. The parameters of other methods are set as the value for best performance.

To show the parameter sensitivity on different label percentages, we randomly select different percentages of labeled data on each data set to perform experiments (30% for COIL20, 10% for UMIST, 20% for USPS, and 40% for YALEB). Fig. 3 shows the influence of parameters on our method and we can

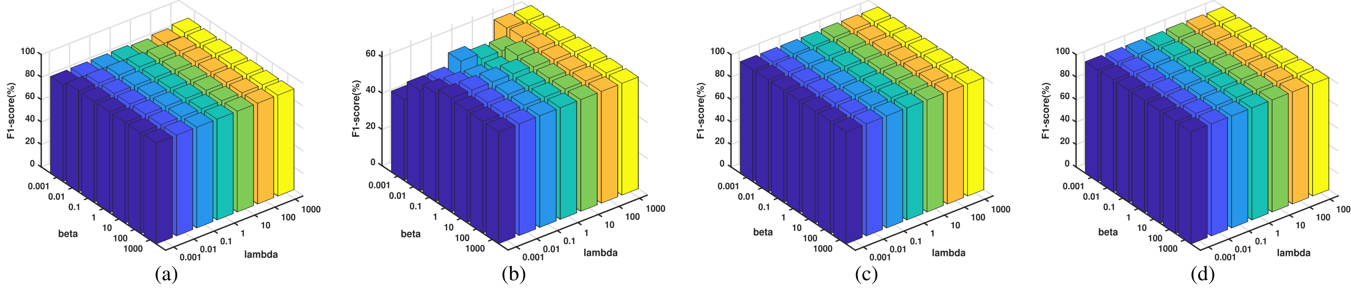


Fig. 3. Effect of parameters to performance on each data set. (a) COIL20 (30%). (b) UMIST (10%). (c) USPS (20%). (d) YALEB (40%).

TABLE III
TIME COST ON EACH DATA SET

Datasets	COIL20	UMIST	USPS	YALEB
Time (s)	9.15	18.96	2.49	967.78

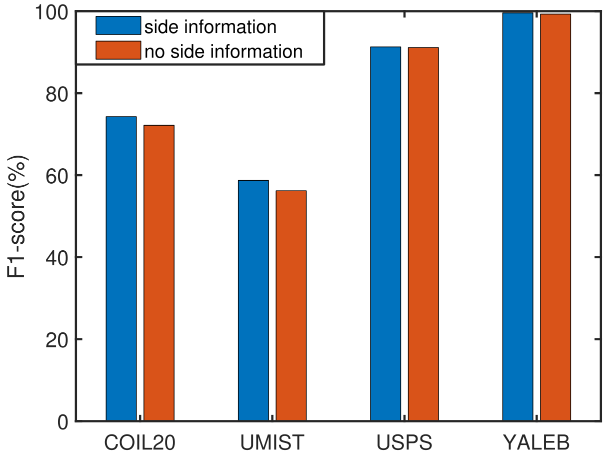


Fig. 4. Effect of adding side information on each data set.

conclude that our method SFS is not particularly sensitive to parameters when the percentage of labeled data is more than 10%. In Fig. 3(b), the percentage of labeled data is 10% and the parameters λ and β have a slight impact on our method when the value of them is [0.001, 0.1].

As for the side information, we do not add the side information in the previous comparative experiment for fairness. However, to show the effect of adding side information, we randomly select ten (very little) pairs of points as ML or CL to verify the validity of adding side information. Note that we only select very little samples to simulate the ML and CL in this section, and you can determine ML and CL according to the actual situation. Fig. 4 shows the effect of adding side information and the results show that the performance is improved by adding the side information.

To verify the influence of the construction way of the similarity matrix to our method, we compared the performance of our method in two construction ways. In Fig. 5, there are two ways to construct the similarity matrix: one is CAN [31] and another is the KNN. The result in Fig. 5 shows that the way constructing the similarity matrix does not have much impact on our method. In this article, we use the KNN that is a more general way to construct the similarity matrix and the performance of our method can be better if we use a more refined similarity matrix construction way.

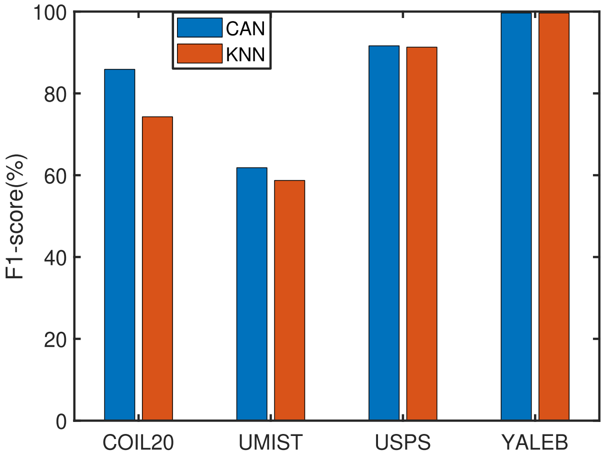


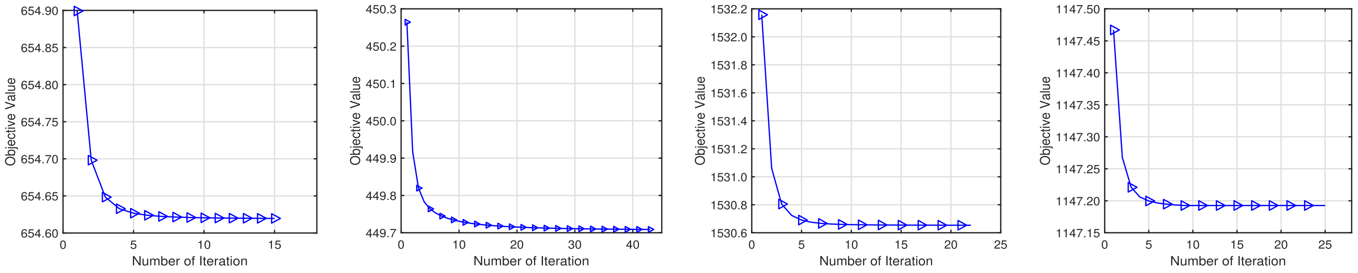
Fig. 5. Effect of the similarity matrix.

F. Experimental Results and Analysis

In this section, we will explain our experimental results from three aspects. The first aspect is the label prediction performance of semisupervised learning in Table II, the second aspect is the performance of feature selection on the classification task in Fig. 1, and the third aspect is the performance of feature selection on the clustering task in Fig. 2.

In the label prediction performance of semisupervised learning in Table II, our method distinctly outperforms other competitors. Especially when the percentage of labeled data is less than 10%, our method is 10%–20% higher than other methods on COIL20, UMIST, and USPS. This is because the similarity matrix and side information equip our framework adequate prior information, which enables our method to perform well even if the labeled data are little. Besides that, the competitors CSFS, SFSS, and SRLSR all use the ridge regression in their framework. Comparing to these methods, it is clear that the generalized uncorrelated constraint produces a significant effect in our framework.

Fig. 1 shows the performance of feature selection on the classification task. Since all methods perform better than the baseline SVM, the feature selection is effective. In the COIL20 and UMIST data sets, our method is 5%–10% higher than the other methods. Also, in COIL20, USPS, and YALEB,



(a) (b) (c) (d)

Fig. 6. Convergence of the algorithm on each data set. (a) COIL20. (b) UMIST. (c) USPS. (d) YALEB.

the performance increases as the number of selected features increases, which is because the increase of selected features brings more information. In the UMIST, the classification performance first rises and then falls, and this is because too many selected features bring unnecessary noise.

Fig. 2 shows the performance of feature selection on the clustering task. Since all methods perform better than the baseline k-means, the feature selection is effective. In COIL20, UMIST, and USPS, our method outperforms other competitors. However, there is only a little difference in the performance between each method on YALEB due to the number of total samples in this data set is very large, i.e., over 10^4. In addition, the clustering performance first rises and then falls in the UMIST and USPS, and it is because too many selected features bring unnecessary noise.

VI. Conclusion

In this article, we propose a novel semisupervised feature selection method (SFS) via the general uncorrelated constraint and manifold embedding for seeking the uncorrelated and discriminative features. The generalized uncorrelated constraint equips the framework with the closed-form solution when the ridge regression is applied to semisupervised learning. In addition, the generalized uncorrelated constraint not only enables the method to perform manifold structure but also avoids excessive row sparsity of the projection matrix. Furthermore, the side information makes the Laplacian structure more accurate since it brings correct prior pairwise connections. Experimental results support the effectiveness of our method by comparing it to other state-of-the-art approaches.

VII. Future Research

Since our method can perform the manifold structure, we try to combine our method with the graph convolution network from the perspective of graph embedding. In future work, we will equip the graph construction with adaptive learning and embed the updating of the similarity matrix into the framework. Also, the similarity matrix constructed in this article will be tried as the adjacency matrix in the graph convolution network for more experiments. Besides that, we will consider creating a new graph embedding method by graph convolution that is different from graph autoencoder. In the feature work, we will make more attempts based on this idea.

References

[1] X. Li, M. Chen, F. Nie, and Q. Wang, “Locality adaptive discriminant analysis,” in *Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI)*, Aug. 2017, pp. 2201–2207.

[2] X. Li, M. Chen, F. Nie, and Q. Wang, “A multiview-based parameter free framework for group detection,” in *Proc. 21st AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 1–7.

[3] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, N. Sebe, and A. G. Hauptmann, “Discriminating joint feature analysis for multimedia data understanding,” *IEEE Trans. Multimedia*, vol. 14, no. 6, pp. 1662–1672, Dec. 2012.

[4] R. Sheikhpour, M. A. Sarram, S. Gharaghani, and M. A. Z. Chahooki, “A survey on semi-supervised feature selection methods,” *Pattern Recognit.*, vol. 64, pp. 141–158, Apr. 2017.

[5] X.-D. Wang, R.-C. Chen, F. Yan, Z.-Q. Zeng, and C.-Q. Hong, “Fast adaptive K-means subspace clustering for high-dimensional data,” *IEEE Access*, vol. 7, pp. 42639–42651, 2019.

[6] S. S. Danraka, S. M. Yahaya, A. D. Usman, A. Umar, and A. M. Abubakar, “Discrete firefly algorithm based feature selection scheme for improved face recognition,” *Comput. Inf. Syst.*, vol. 23, no. 2, pp. 23–34, May 2019.

[7] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, “Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking,” *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5596–5609, Nov. 2019.

[8] R. Zhang, F. Nie, X. Li, and X. Wei, “Feature selection with multi-view data: A survey,” *Inf. Fusion*, vol. 50, pp. 158–167, Oct. 2019.

[9] R. Sheikhpour, M. A. Sarram, and E. Sheikhpour, “Semi-supervised sparse feature selection via graph Laplacian based scatter matrix for regression problems,” *Inf. Sci.*, vol. 468, pp. 14–28, Nov. 2018.

[10] X. Chang and Y. Yang, “Semisupervised feature analysis by mining correlations among multiple tasks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2294–2305, Oct. 2017.

[11] Y. Zhang, H.-G. Li, Q. Wang, and C. Peng, “A filter-based barebone particle swarm optimization algorithm for unsupervised feature selection,” *Int. J. Speech Technol.*, vol. 49, no. 8, pp. 2889–2898, Aug. 2019.

[12] T. Thaher, M. Mafarja, B. Abdalhaq, and H. Chantar, “Wrapper-based feature selection for imbalanced data using binary queuing search algorithm,” in *Proc. 2nd Int. Conf. Trends Comput. Sci. (ICTCS)*, Oct. 2019, pp. 1–6.

[13] R. Zhang, F. Nie, and X. Li, “Self-weighted supervised discriminative feature selection,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3913–3918, Aug. 2018.

[14] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003.

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

[16] M. Luo, F. Nie, X. Chang, Y. Yang, A. G. Hauptmann, and Q. Zheng, “Adaptive unsupervised feature selection with structure regularization,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 944–956, Apr. 2018.

[17] X.-D. Wang, R.-C. Chen, C.-Q. Hong, and Z.-Q. Zeng, “Unsupervised feature analysis with sparse adaptive learning,” *Pattern Recognit. Lett.*, vol. 102, pp. 89–94, Jan. 2018.

[18] A. M. E. Saleh, M. Arashi, and B. G. Kibria, *Theory of Ridge Regression Estimation with Applications*, vol. 285. Hoboken, NJ, USA: Wiley, 2019.

[19] X. Chang, F. Nie, Y. Yang, and H. Huang, “A convex formulation for semi-supervised multi-label feature selection,” in *Proc. 28th AAAI Conf. Artif. Intell.*, Jun. 2014, pp. 1–7.

[20] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe, and X. Zhou, “Semisupervised feature selection via spline regression for video semantic recognition,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 252–264, Feb. 2015.

[21] E. Yu, J. Sun, J. Li, X. Chang, X.-H. Han, and A. G. Hauptmann, "Adaptive semi-supervised feature selection for cross-modal retrieval," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1276–1288, May 2019.

[22] T. Luo, C. Hou, F. Nie, H. Tao, and D. Yi, "Semi-supervised feature selection via insensitive sparse regression with application to video semantic recognition," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 10, pp. 1943–1956, Oct. 2018.

[23] X. Chen, G. Yuan, F. Nie, and Z. Ming, "Semi-supervised feature selection via sparse rescaled linear square regression," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 1, pp. 165–176, Jan. 2020.

[24] X. Li, H. Zhang, R. Zhang, Y. Liu, and F. Nie, "Generalized uncorrelated regression with adaptive graph for unsupervised feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1587–1595, May 2019.

[25] H. Zhang, R. Zhang, F. Nie, and X. Li, "A generalized uncorrelated ridge regression with nonnegative labels for unsupervised feature selection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2781–2785.

[26] X. Li, H. Zhang, R. Zhang, and F. Nie, "Discriminative and uncorrelated feature selection with constrained spectral analysis in unsupervised learning," *IEEE Trans. Image Process.*, vol. 29, pp. 2139–2149, 2020.

[27] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners and open problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 3, pp. 252–264, Mar. 1991.

[28] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, Jan. 2004.

[29] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Nonlinear dimensionality reduction with local spline embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1285–1298, Sep. 2009.

[30] L. Kozma, *K Nearest Neighbors Algorithm (kNN)*. Espoo, Finland: Helsinki Univ. of Technol. Press, 2008.

[31] F. Nie, X. Wang, and H. Huang, "Clustering and projected clustering with adaptive neighbors," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 977–986.

[32] J. Huang, F. Nie, H. Huang, and C. Ding, "Robust manifold nonnegative matrix factorization," *ACM Trans. Knowl. Discovery Data*, vol. 8, no. 3, p. 11, 2014.

[33] 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.

[34] J. Xu, B. Tang, H. He, and H. Man, "Semisupervised feature selection based on relevance and redundancy criteria," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 9, pp. 1974–1984, Sep. 2017.

Xuelong Li (Fellow, IEEE) is currently a Full Professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China.

Yunxing Zhang received the B.E. degree in mechatronics engineering from Northwestern Polytechnical University, Xi'an, China, in 2019, where he is currently pursuing the master's degree with the School of Computer Science and the School of Artificial Intelligence, Optics and Electronics (iOPEN).

Rui Zhang (Member, IEEE) received the Ph.D. degree in computer science from Northwestern Polytechnical University, Xi'an, China, in 2018.

He is currently an Associate Professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University.