Semisupervised Feature Selection With Sparse Discriminative Least Squares Regression

Chen Wang, Xiaojun Chen, *Member, IEEE*, Guowen Yuan, Feiping Nie, *Member, IEEE*, and Min Yang

***Abstract*—In big data time, selecting informative features has become an urgent need. However, due to the huge cost of obtaining enough labeled data for supervised tasks, researchers have turned their attention to semisupervised learning, which exploits both labeled and unlabeled data. In this article, we propose a sparse discriminative semisupervised feature selection (SDSSFS) method. In this method, the ϵ-dragging technique for the supervised task is extended to the semisupervised task, which is used to enlarge the distance between classes in order to obtain a discriminative solution. The flexible $\ell_{2,p}$ norm is implicitly used as regularization in the new model. Therefore, we can obtain a more sparse solution by setting smaller p. An iterative method is proposed to simultaneously learn the regression coefficients and ϵ-dragging matrix and predicting the unknown class labels. Experimental results on ten real-world datasets show the superiority of our proposed method.**

***Index Terms*—Discriminative feature selection, feature selection, least square regression, semisupervised feature selection.**

I. Introduction

DURING the past decades, many feature selection methods have been proposed for selecting the important features [1]–[3]. According to whether the label information is available, feature selection can be classified into supervised or unsupervised methods. The supervised feature selection methods, for example, the Fisher score [4]; Relief-F [5], [6]; RFS [7]; and GRM [8], evaluate the feature relevance according to the correlation between the feature and the class labels. Unsupervised feature selection, such as the Laplacian score (LS) [9], RSFS [10], and SOGFS [11], evaluate the feature relevance by feature dependency or similarity. However, in big data time, it is costly to obtain labeled data [12]. Therefore, to liberate us from laborious and trivial data labeling work, people expect to only mark a small set of data samples with ground truth. At the same time, they would like to exploit unlabeled samples during the training procedure to ensure the effectiveness of the learned models. The task of "semisupervised feature selection" that conducts feature selection by exploring both labeled and unlabeled data is gaining more and more attention.

The early semisupervised feature selection methods are mainly filter based [13]–[16], which could discard important features that are informative when combined with other features or wrapper type which is usually time consuming [17]. Least squares regression is a very important method [18], and researchers have proposed some embedded semisupervised methods based on regression recently. Xu *et al.* [19] proposed a discriminative least squares regression (DLSR) based on manifold regularization, but their method has high computational complexity. Chen *et al.* proposed a semisupervised feature selection method RLSR [20], in which a rescaled linear square regression is proposed to replace the conventional least square regression for feature selection. The new method provides a good theoretical explanation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to rank the features for the $\ell_{2,1}$-norm-regularized least square regression.

In order to select features with discriminative power, it is often desired that the distances between data points in different classes are as large as possible after they are transformed. Also, compared with the ℓ_2 norm, the $\ell_{2,p}$ norm can reduce the impact of outliers to the models [21]. Moreover, it has been shown that the $\ell_{2,p}$ norm ($p \in (0, 1)$) is able to obtain a more feasible solution than the $\ell_{2,1}$ norm [22] and improve the supervised feature selection performance [23]. In order to improve the robustness and discriminative power of RLSR, in this article, we propose a novel embedded semisupervised feature selection method, called the sparse discriminative semisupervised feature selection (SDSSFS). In the new method, we extend the ϵ-dragging technique for the supervised task to the semisupervised task in order to obtain a discriminative solution and implicitly used the $\ell_{2,p}$ norm for a more feasible solution. The experimental results on ten real-world datasets show the superiority of the new method. Note that part of our early results has been reported in our abstract paper [24]. The main contributions of our work include the following.

1) We have extended the ϵ-dragging technique used in the supervised task for the semisupervised task, in order

Manuscript received 26 October 2018; revised 17 October 2020 and 22 January 2021; accepted 17 February 2021. Date of publication 19 April 2021; date of current version 19 July 2022. This work was supported in part by the Major Project of the New Generation of Artificial Intelligence under Grant 2018AAA0102900; in part by NSFC under Grant 61773268; in part by the Natural Science Foundation of SZU under Grant 000346; and in part by the Shenzhen Research Foundation for Basic Research, China, under Grant JCYJ20180305124149387. This article was recommended by Associate Editor Y. Jin. *(Corresponding author: Xiaojun Chen.)*

Chen Wang, Xiaojun Chen, and Guowen Yuan are with the College of Computer Science and Software, Shenzhen University, Shenzhen 518060, China (e-mail: chenwang417@163.com; xjchen@szu.edu.cn; gwyuan93@qq.com).

Feiping Nie is with the School of Computer Science and Center for Optical Imagery Analysis and Learning, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: feipingnie@gmail.com).

Min Yang is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: min.yang1129@gmail.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2021.3060804.

Digital Object Identifier 10.1109/TCYB.2021.3060804

to enlarge the distances between classes to obtain a discriminative solution.

2) We have proved that the $\ell_{2,p}$ norm is implicitly used as a sparse regularization in our proposed method. The smaller p, the more sparse in rows the learned rescaled regression coefficients are. Therefore, we can obtain a more sparse solution by setting smaller p.
3) An effective iterative algorithm is proposed to solve the new model, with a computational complexity of $O(d^3 + nd^2 + dnc)$, where n is the number of samples, d is the number of features, and c is the number of classes.
4) Experimental results on ten benchmark datasets demonstrate the superior performance of SDSSFS over the state-of-the-art methods.

The remainder of this article is organized as follows. Section II presents the notations and the definition of norms used in this article and gives a brief review of the related work. In Section III, the new method SDSSFS is proposed. The experimental results are reported in Section IV. The conclusion and future work are given in Section V.

II. Notations and Related Work

In this section, we summarize the notations and the definition of norms used in this article and give a brief review of related work.

A. Notations and Definitions

We write matrices as boldface uppercase letters and vectors as boldface lowercase letters. Given a matrix $\mathbf{M} = (m_{ij})$, $\mathbf{m}^i$ denotes its ith row and $\mathbf{m}_j$ denotes its jth column. The Frobenius norm of $\mathbf{M} \in \mathbb{R}^{n\times m}$ is defined as

$$\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} m_{ij}^2}. \tag{1}$$

The $\ell_{2,p}$-norm of $M \in \mathbb{R}^{n\times m}$ is defined as

$$\|\mathbf{M}\|_{2,p} = \left(\sum_{i=1}^{n}\left(\sum_{j=1}^{m} m_{ij}^2\right)^{\frac{p}{2}}\right)^{\frac{1}{p}}. \tag{2}$$

B. Semisupervised Feature Selection

Given a dataset $\mathbf{X} \in \mathbb{R}^{d\times n}$ with C classes, which consists of two subsets: 1) a set of l labeled objects $\mathbf{X}_L = (\mathbf{x}_1, \ldots, \mathbf{x}_l)$ that are associated with class labels $\mathbf{Y}_L = \{\mathbf{y}_1, \ldots, \mathbf{y}_l\}^T \in \mathbb{R}^{l\times c}$ and 2) a set of $u = n - l$ unlabeled objects $\mathbf{X}_U = (\mathbf{x}_{l+1}, \ldots, \mathbf{x}_{l+u})^T$, whose labels $\mathbf{Y}_U = \{\mathbf{y}_{l+1}, \ldots, \mathbf{y}_{l+u}\}^T \in \mathbb{R}^{u\times c}$ are unknown. Here, $\mathbf{y}_i \in \mathbb{R}^c (1 \le i \le l)$ is a binary vector defined as

$$\mathbf{y}_i^j = \begin{cases} 1, & \text{if } \mathbf{x}_i \text{ belongs to the } j\text{th class} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Most of the existing semisupervised feature selection methods are filter based, which score the features with a ranking criterion regardless of the model [13]–[16]. For example, Zhao and Liu [13] proposed sSelect, which is based on spectral analysis. In the feature view, $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_d\}$, where $\mathbf{f}_i$ denotes the ith feature vectors in $\mathbf{X}$. A similarity matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ can be computed with the RBF kernel function, in which a_{ij} is defined as follows:

$$a_{ij} = \begin{cases} e^{\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\delta^2}}, & \text{if } \mathbf{x}_i \in \mathbf{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathbf{N}_k(i) \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $\mathbf{N}_k(\mathbf{x}_j)$ is a set containing the k-nearest neighbors of $\mathbf{x}_j$, and δ is a parameter. Select computes a score s_j for the jth feature as follows:

$$s_j = \lambda \frac{\sum_{i,h=l+1}^{l+u} (g_i - g_h)^2 \times a_{ih}}{2\sum_{i=l+1}^{l+u} g_i^2 \times \mathbf{d}_i} + (1-\lambda)\big(1 - NMI\big(\hat{g}, \mathbf{Y}_L\big)\big) \tag{5}$$

where λ is a parameter, $\mathbf{g} \in \mathbb{R}^{n\times 1}$ is the cluster indicator generated from the jth feature vector $\mathbf{f}_j \in \mathbb{R}^{n\times 1}$ with the F-C transformation, and $\hat{g} \in \mathbb{R}^{l\times c}$ is the cluster labels obtained from $\mathbf{g}$ and $\mathbf{d}_i = \sum_{j=1}^{n} a_{ij}$. The overall computational complexity of sSelect is $O(dn^2)$.

Based on the Laplacian criteria, Zhao *et al.* [14] proposed a locality-sensitive method LSDF. The new method first constructs a within-class affinity matrix $\mathbf{W}$ and a between-class affinity matrix $\mathbf{S}$. Then, the importance score for the jth feature can be computed as

$$\mathbf{L}_j = \frac{\mathbf{f}_j^T \mathbf{L}_b \mathbf{f}_j}{\mathbf{f}_j^T \mathbf{L}_w \mathbf{f}_j} \tag{6}$$

in which b_{ij} is defined as

$$b_{ij} = \begin{cases} 1, & \text{if if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ belong to different classes} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where $\mathbf{f}_j \in \mathbb{R}^{n\times 1}$ is the jth feature vector, $\mathbf{L}_w = \mathbf{D}_w - \mathbf{W}$ ($\mathbf{D}_w = \text{diag}(\mathbf{W1})$) and $\mathbf{L}_b = \mathbf{D}_b - \mathbf{S}$ ($\mathbf{D}_b = \text{diag}(\mathbf{S1})$) are graph Laplacians. The computational complexity of LSDF is $O(dn^2)$.

Doquire and Verleysen [15] proposed a semisupervised LS (SSLS). In their method, they first construct two affinity matrices $\mathbf{S}^{\text{sup}}$ and $\mathbf{S}^{\text{semi}}$. Then, SSLS measures the importance of the jth feature as follows:

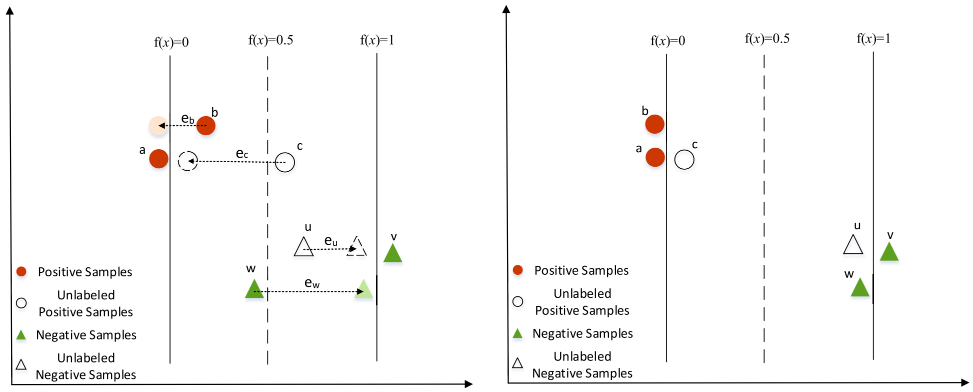
$$\text{SSLS}_j = \frac{\tilde{\mathbf{f}}_j^T \mathbf{L}^{\textbf{semi}} \tilde{\mathbf{f}}_j}{\tilde{\mathbf{f}}_j^T \mathbf{D}^{\textbf{semi}} \tilde{\mathbf{f}}_j} \times \frac{\tilde{\mathbf{f}}_j^T \mathbf{L}^{\textbf{sup}} \tilde{\mathbf{f}}_j}{\tilde{\mathbf{f}}_j^T \mathbf{D}^{\textbf{sup}} \tilde{\mathbf{f}}_j} \tag{8}$$

where $\tilde{\mathbf{f}}_j = \mathbf{f}_j - [(\mathbf{f}_j^T \mathbf{D}^{\textbf{semi}} \mathbf{1})/(\mathbf{1}_j^T \mathbf{D}^{\textbf{semi}} \mathbf{1})]\mathbf{1}$ and $\mathbf{L}^{\text{semi}} = \mathbf{D}^{\textbf{semi}} - \mathbf{S}^{\text{semi}}$. It can be verified that SSLS also has a computational complexity of $O(dn^2)$.

Based on the relevance and redundancy criteria, Xu *et al.* [16] proposed RRPC. Assume that we have obtained feature subsets $\mathbf{F}_{k-1}$, which consists of $k-1$ selected features from $\mathbf{F}$. The kth feature is selected from $\{\mathbf{F} - \mathbf{F}_{k-1}\}$ as follows:

$$\mathbf{F}_k = \mathop{\arg\min}_{\mathbf{F}_j \in \mathbf{F} - \mathbf{F}_{k-1}} \left[\mathbf{P}\big(\mathbf{F}_j, \mathbf{Y}_L\big) - \frac{1}{k-1} \sum_{\mathbf{F}_i \in\, \mathbf{F}_{k-1}} \mathbf{P}\big(\mathbf{F}_j, \mathbf{F}_i\big)\right] \tag{9}$$

where $\mathbf{P}(\mathbf{F}_j, \mathbf{Y}_L)$ is Pearson's correlation coefficient between two vectors $\mathbf{F}_j$ and the label matrix $\mathbf{Y}_L$. The computational complexity of RRPC is $O(nd^2)$.



(a) (b)

Fig. 1. Illustration of ϵ-dragging in the semisupervised task. (a) ϵ-dragging. (b) After ϵ-dragging.

Ren *et al.* [17] extended the supervised sequential forward feature selection (SFFS) to an iterative "wrapper-type" forward semisupervised feature selection framework. In each iteration, a classifier is trained from the training data (labeled data at first) with the currently selected feature subset and used to predict unlabeled samples. Then, a number of "labeled" unlabeled samples are added to the labeled data to form new training data. But such a method is usually time consuming because it involves iterative feature subset searching.

Recently, many embedded methods have been proposed. Xu *et al.* [19] proposed FS-Manifold, which is based on the idea of manifold regularization. In their method, an affinity matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is initially constructed in which a_{ij} is computed according to (4). Then, the projection matrix $\mathbf{W}$ is learned by optimizing the following objective function:

$$\min_{\mathbf{W},\mathbf{b},\xi} \left(\frac{\|\mathbf{w}\|_2^2}{2} + C\sum_{i=1}^{l}\xi_i + \frac{\rho}{2}\mathbf{w}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{w}\right)$$
$$\text{s.t.} \quad \mathbf{y}_i\left(\mathbf{w}^T\mathbf{x}_i - b\right) \geq 1 - \xi_i \ (1 \leq i \leq l)$$
$$\xi_i \geq 0 \ (1 \leq i \leq l) \tag{10}$$

where C and ρ are two parameters and $\mathbf{L} = \text{diag}(\mathbf{A}\mathbf{1}) - \mathbf{A}$ is the graph Laplacian. They proposed to use the level method to optimize the above objective function, but it has high computational complexity bounded by $O(n^{2.5}/\varepsilon^2)$, where ε is a small stopping criterion.

Chen *et al.* [20] proposed **RLSR**, in which a rescaled linear square regression is proposed to extend the least square regression for feature selection. To rank the d features of $\mathbf{X}$ by consuming both $\mathbf{X}_L$ and $\mathbf{X}_U$, RLSR introduces d scale factors θ in which $\theta_j > 0$ $(1 \leq j \leq d)$ measures the importance of the jth feature. To learn θ and $\mathbf{Y}_U$ simultaneously, RLSR rescales the regression coefficients $\mathbf{W} \in \mathbb{R}^{d\times c}$ with a rescale matrix $\Theta \in \mathbb{R}^{d\times d}$, which is a diagonal matrix and $\mathbf{\Theta}_{jj} = \theta_j^{1/2}$. The objective function of RLSR is defined as

$$\min_{\mathbf{W},\mathbf{b},\mathbf{\Theta},\mathbf{Y}_U} \left(\left\|\mathbf{X}^T\Theta\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\|\mathbf{W}\|_F^2\right)$$
$$\text{s.t.} \quad \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1} \tag{11}$$

which is proved to be equivalent to a sparse feature selection model

$$\min_{\mathbf{W},\mathbf{b},\mathbf{M}\geq 0} \left(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\circ\mathbf{M}\right\|_F^2 + \gamma\|\mathbf{W}\|_{2,1}^2\right) \tag{12}$$

where $\mathbf{Y}_U$ are relaxed as continuous values in [0, 1]. $\mathbf{b} \in \mathbb{R}^c$ is the bias and $\gamma > 0$ is the regularization parameter.

III. Proposed Method

In this article, we extend the RLSR in (11) as follows:

$$\min_{\mathbf{W},\mathbf{b},\mathbf{\Theta},\mathbf{Y}_U,\mathbf{M}} \left(\left\|\mathbf{X}^T\Theta\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y}-1)\circ\mathbf{M}\right\|_F^2 + \gamma\|\mathbf{W}\|_F^2\right)$$
$$\text{s.t.} \quad \theta > 0, \ \mathbf{1}^T\theta = 1, \ \mathbf{Y}_U \geq 0, \ \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \ \mathbf{M} \geq \mathbf{0} \tag{13}$$

where $\mathbf{\Theta} \in \mathbb{R}^{d\times d}$ is a diagonal matrix and $\Theta_{jj} = \theta_j^{q/2} (q \geq 1)$, $q \in (0, 1]$ is a parameter, and $\mathbf{M}$ is a positive ϵ-dragging matrix in which m_{ij} is the slack variable for the ith sample on the jth class.

The ϵ-dragging technique was originally used in [25] for the supervised task. In our method, we extend the ϵ-dragging technique was extended for semisupervised task, as shown in Fig. 1(a). Let $f(x)$ be the logistic regression function to learn, where $f(x) = 1$ indicates that $\mathbf{x}$ is a positive sample and $f(x) = 0$ indicates that $\mathbf{x}$ is a negative sample. Denote $\mathbf{E} = (2\mathbf{Y}-1)\mathbf{M}$. If the label of an object $\mathbf{x}$ is known, we hope to learn $f(x) = 1 + m_{ij}$ if $\mathbf{x}$ belongs to the jth class, or $f(x) = 1 - m_{ij}$ if $\mathbf{x}$ belongs to the jth class.

Let $\widetilde{\mathbf{W}} = \mathbf{\Theta}\mathbf{W}$, then $\mathbf{W} = \mathbf{\Theta}^{-1}\widetilde{\mathbf{W}}$. Problem (13) can be rewritten as

$$\min_{\widetilde{\mathbf{W}},\mathbf{b},\mathbf{\Theta},\mathbf{Y}_U,\mathbf{M}} \left(\left\|\mathbf{X}^T\widetilde{\mathbf{W}} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y}-1)\circ\mathbf{M}\right\|_F^2 + \gamma\left\|\mathbf{\Theta}^{-1}\widetilde{\mathbf{W}}\right\|_F^2\right)$$
$$\text{s.t.} \quad \widetilde{\mathbf{W}}, \mathbf{b}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \ \mathbf{M} \geq \mathbf{0} \tag{14}$$

which leads to a new problem of SDSSFS

$$\min_{\mathbf{W},\mathbf{b},\mathbf{\Theta},\mathbf{Y}_U,\mathbf{M}} \left(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y}-1)\circ\mathbf{M}\right\|_F^2 + \gamma\left\|\mathbf{\Theta}^{-1}\mathbf{W}\right\|_F^2\right)$$
$$\text{s.t.} \quad \mathbf{W}, \mathbf{b}, \mathbf{\Theta}_{jj} = \theta_j^{q/2}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \ \mathbf{M} \geq \mathbf{0}. \tag{15}$$

Problem (15) can be solved with an alternate optimization approach. In the following, we show how to alternatively update each variable in this problem.

A. Update $\mathbf{b}$ *With* $\mathbf{W}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, *and* $\mathbf{Y_U}$ *Fixed*

When $\mathbf{W}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, and $\mathbf{Y_U}$ are fixed, problem (15) becomes

$$\min_{\mathbf{b}} \ \left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y} - 1) \circ \mathbf{M}\right\|_F^2. \tag{16}$$

Setting the partial derivative of problem (16) with respect to $\mathbf{b}$ to 0 gives

$$\mathbf{b} = \frac{1}{n}\left[(\mathbf{Y} + (2\mathbf{Y} - 1) \circ \mathbf{M})^T\mathbf{1} - \mathbf{W}^T\mathbf{X}\mathbf{1}\right]. \tag{17}$$

B. Update $\mathbf{W}$ *With* $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, *and* $\mathbf{Y_U}$ *Fixed*

When $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, and $\mathbf{Y_U}$ are fixed, problem (15) becomes

$$\min_{\mathbf{W}} \ \left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y} - 1) \circ \mathbf{M}\right\|_F^2 + \gamma\left\|\Theta^{-1}\mathbf{W}\right\|_F^2. \tag{18}$$

Substituting $\mathbf{b}$ in (17) into (18), we get

$$\min_{\mathbf{W}} \ \left\|\mathbf{H}\mathbf{X}^T\mathbf{W} - \mathbf{H}(\mathbf{Y} + (2\mathbf{Y} - 1) \circ \mathbf{M})\right\|_F^2 + \gamma Tr\left(\mathbf{W}^T\Theta^{-2}\mathbf{W}\right) \tag{19}$$

where $\mathbf{H} = \mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T$.

The Lagrangian function of problem (19) is

$$\mathcal{L}(\mathbf{W}) = \left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - (2\mathbf{Y} - 1) \circ \mathbf{M}\right\|_F^2 + \gamma Tr\left(\mathbf{W}^T\Theta^{-2}\mathbf{W}\right). \tag{20}$$

Taking the derivative of $\mathcal{L}(\mathbf{W})$ with respect to $\mathbf{W}$ and setting the derivative to 0 gives

$$2\mathbf{X}\mathbf{H}^T\left(\mathbf{H}\mathbf{X}^T\mathbf{W} - \mathbf{H}(\mathbf{Y} + (2\mathbf{Y} - 1) \circ \mathbf{M})\right) + 2\gamma\Theta^{-2}\mathbf{W} = 0. \tag{21}$$

Then, we obtain the optimal solution of $\mathbf{W}$ as

$$\mathbf{W} = \left(\mathbf{X}\mathbf{H}^T\mathbf{H}\mathbf{X}^T + \gamma\Theta^{-2}\right)^{-1}\mathbf{X}\mathbf{H}^T\mathbf{H}\mathbf{Y}. \tag{22}$$

It can be verified that $\mathbf{H}^T\mathbf{H} = \mathbf{H}$, so the optimal solution of $\mathbf{W}$ can be rewritten as

$$\mathbf{W} = \left(\mathbf{X}\mathbf{H}\mathbf{X}^T + \gamma\mathbf{Q}\right)^{-1}\mathbf{X}\mathbf{H}(\mathbf{Y} + (2\mathbf{Y} - 1) \circ \mathbf{M}). \tag{23}$$

C. Update $\boldsymbol{\Theta}$ *With* $\mathbf{Y}_U$, $\mathbf{W}$, *and* $\mathbf{b}$ *Fixed*

When $\mathbf{W}$, $\mathbf{Y}_U$, and $\mathbf{b}$ are fixed, we can obtain the optimal solution to $\boldsymbol{\Theta}$ by solving the following problem:

$$\min_{\theta > 0, \mathbf{1}^T\theta = 1} \ \sum_{j=1}^{d} \frac{\left\|\mathbf{w}^j\right\|_2^2}{\theta_j^q} \tag{24}$$

where θ_j is the jth diagonal element of Θ.

The Lagrangian function of problem (24) is

$$\mathcal{L}(\theta, \chi, \tau) = \sum_{j=1}^{d} \frac{\left\|\mathbf{w}^j\right\|_2^2}{\theta_j^q} + \chi\left(\sum_{j=1}^{d}\theta_j - 1\right) - \theta^T\tau \tag{25}$$

where χ and positive vector τ are Lagrangian multipliers.

Setting the derivative of $\mathcal{L}(\theta, \chi, \tau)$ with respect to θ to 0 gives

$$\frac{\partial\mathcal{L}(\theta, \chi, \tau)}{\partial\theta_j} = -\ q\frac{\left\|\mathbf{w}^j\right\|_2^2}{\theta_j^{q+1}} + \chi = 0 \tag{26}$$

which leads to

$$\theta_j = \left(q\frac{\left\|\mathbf{w}^j\right\|_2^2}{\chi}\right)^{\frac{1}{q+1}}. \tag{27}$$

Setting the derivative of $\mathcal{L}(\theta, \chi, \tau)$ with respect to χ to 0 gives

$$\sum_{j=1}^{d}\theta_j - 1 = 0. \tag{28}$$

Substituting θ in (27) into (28), we get the optimal solution of θ_j as

$$\theta_j = \frac{\left\|\mathbf{w}^j\right\|_2^{\frac{2}{q+1}}}{\sum_{h=1}^{d}\left\|\mathbf{w}^h\right\|_2^{\frac{2}{q+1}}}. \tag{29}$$

Letting $p = (2/q + 1)$, (29) can be rewritten as

$$\theta_j = \frac{\left\|\mathbf{w}^j\right\|_2^p}{\sum_{h=1}^{d}\left\|\mathbf{w}^h\right\|_2^p}. \tag{30}$$

Note that $q \geq 1$, we have $0 < p \leq 1$.

D. Update $\mathbf{Y_U}$ *With* $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, *and* $\mathbf{W}$ *Fixed*

Note that problem (15) is independent between different $\mathbf{y}_i$ $(l + 1 \leq i \leq l + u)$, so we can obtain $\mathbf{y}_i \in \mathbf{Y_U}$ individually from the following problem with fixed $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{M}$, and $\mathbf{W}$:

$$\min_{\mathbf{y}_i \geq 0, \mathbf{y}_i^T\mathbf{1} = 1} \ \left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} + \mathbf{m}^i - (2\mathbf{m}^i + \mathbf{1}) \circ \mathbf{y}_i\right\|_2^2 \tag{31}$$

which can be solved with an efficient method in the Appendix.

E. Update $\mathbf{M}$ *With* $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{W}$, *and* $\mathbf{Y}_U$ *Fixed*

If $\mathbf{b}$, $\boldsymbol{\Theta}$, $\mathbf{W}$, and $\mathbf{Y}_U$ are fixed, it can be verified that the optimal solution of $\mathbf{M}$ is

$$\mathbf{M} = \max\left((2\mathbf{Y} - 1) \circ \left(\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right), \mathbf{0}\right). \tag{32}$$

F. Optimization Algorithm

The detailed algorithm to solve problem (15), namely, SDSSFS, is summarized in Algorithm 1. In the new algorithm, $\mathbf{b}$, Θ, $\mathbf{W}$, $\mathbf{M}$, and $\mathbf{Y_U}$ are alternately updated in each iteration until convergence. Finally, θ is computed according to $\mathbf{W}$ and the top k ranked features are selected according to θ. Since we obtain the optimal solution of $\mathbf{b}$, Θ, $\mathbf{W}$, $\mathbf{M}$, and $\mathbf{Y_U}$ in each iteration, Algorithm 1 will monotonously decrease the objective function of problem (15) until the algorithm converges.

Now, we analyze the computational complexity of SDSSFS. Suppose that we are given a data matrix $\mathbf{X} \in \mathbb{R}^{d \times n}$, which consists of l labeled objects and $n - l$ unlabeled objects. n is the number of samples, d is the number of features, and c is the

Algorithm 1 Algorithm to Solve Problem (15): SDSSFS

```
1: Input: Data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$, labels $\mathbf{Y}_L \in \mathbb{R}^{l\times c}$, number of selected features $k$, norm parameter $p \in (0, 1]$, regularization parameter $\gamma$.
2: Output: $k$ selected features.
3: Initialize $\mathbf{\Theta}_0 \in \mathbb{R}^{d\times d}$ as an identity matrix.
4: $t := 0$.
5: repeat
6:     Update $\mathbf{W}_{t+1}$ according to Eq. (23).
7:     Update $\mathbf{b} = \frac{1}{n}\left[(\mathbf{Y} + (2\mathbf{Y} - 1) \circ \mathbf{M})^T\mathbf{1} - \mathbf{W}^T\mathbf{X1}\right]$.
8:     Update $\mathbf{Y}_U$, in which each $\mathbf{y}_i \in \mathbf{Y}_U$ is individually obtained by solving problem (31).
9:     Update $\mathbf{\Theta}_{t+1}$ in which the $j$-th diagonal element $\mathbf{\Theta}_{jj}$ is defined as $\left(\frac{\|\mathbf{w}^j\|_2^p}{\sum_{h=1}^d \|\mathbf{w}^h\|_2^p}\right)^{\frac{1}{p}-0.5}$.
10:    Update $\mathbf{M}$ according to Eq. (32).
11:    $t := t + 1$.
12: until Converges
13: Compute $\theta = \frac{\|\mathbf{w}^j\|_2^p}{\sum_{h=1}^d \|\mathbf{w}^h\|_2^p}$.
14: Sort $\theta$ in descending order, and select top $k$ ranked features as ultimate result.
```

number of classes. In step 6, we need $O(nd^2)$ time to compute $\mathbf{XHX}^T$ since $\mathbf{XHX}^T = \mathbf{XX}^T - 1/n(\mathbf{X1})(\mathbf{X1})^T$, and $O(d^3+dnc)$ time to compute $(\mathbf{XHX}^T + \gamma\mathbf{\Theta}_t^{-2})^{-1}\mathbf{XHY}$. Thus, the total complexity in step 6 is $O(d^3+nd^2)$. In step 7, we need $O(ncd)$ time to update $\mathbf{b}$. In step 8, for each $i \in [l + 1, n]$, we can compute $\mathbf{u}$ in $O(c)$ time and solve β_l^* in $O(c)$ times. Therefore, we need less than $O(nc)$ time to obtain $\mathbf{Y}_U$. Finally, we need $O(dc)$ time to update $\mathbf{\Theta}$ in step 9. Finally, the computational complexity of SDSSFS is $O(d^3 + nd^2 + dnc)$.

Substituting the optimal solution of θ_j defined in (30) into (15), we get the following equivalent problem:

$$\min_{\mathbf{W},\mathbf{b},\mathbf{\Theta},\mathbf{Y}_U,\mathbf{M}} \left(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1b}^T - \mathbf{Y} - (2\mathbf{Y} - 1) \circ \mathbf{M}\right\|_F^2 + \gamma\|\mathbf{W}\|_{2,p}^2\right)$$
$$\text{s.t.} \quad \mathbf{W}, \mathbf{b}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{M} \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1} \tag{33}$$

where $p = (2/q + 1)$. Note that $q \geq 1$, we have $0 < p \leq 1$. In such a case, minimizing $\|\mathbf{W}\|_{2,p}^2$ makes $\mathbf{W}$ sparse in rows. Therefore, problem (15) implicitly imposes a sparse $\ell_{2,p}$ norm regularizer on $\mathbf{W}$. If $p = 1$, the commonly used $\ell_{2,1}$ norm is implicitly used as a sparse regularizer. The smaller p, the sparser the learned regression coefficients $\mathbf{W}$ are.

Moreover, if $p = 1$, problem (33) is a convex problem and Algorithm 1 will converge to its global optima. If $0 < p < 1$, problem (33) is not a convex problem and Algorithm 1 will converge to its local optima.

IV. Experimental Results and Analysis

In this section, we show the performance of the proposed method on real-world datasets.

A. Benchmark Datasets

The following ten benchmark datasets were used in our experiment.

1) The *Binalpha dataset* is one of the datasets for digits and characters recognition, it includes 1404 images with 320 features. These images are from 36 classes, including ten digits (integers in ranges 0–9) and 26 characters (a–z).
2) The *CNAE-9 dataset* is free text business descriptions of Brazilian companies, these data include 1080 documents represented with 856 frequency binary features, and these documents are categorized into nine categories.
3) The *Colon dataset* is used in different research papers on gene expression, it contains 62 samples represented with expression levels of 2000 genes, for each sample, it is indicated whether it came from a tumor biopsy or not, and these samples are categorized into two categories, which includes 40 tumor biopsies and 22 normal biopsies.
4) The Caltech Silhouettes (CSs) dataset is a new dataset based on the CalTech 101 image annotations, which consists of 8641 binary images, CSs data contain silhouettes of objects belonging to 101 categories and it has two versions with sizes of 16 × 16 and 28 × 28, for our experiments, we selected the data with a sample size of 16 × 16.
5) The *Isolet dataset* is used to predict which letter-name was spoken—a simple classification task, these data consist of 7797 voice samples for the name of each letter of the 26 alphabets. Each voice sample has 617 features that include spectral coefficients, contour features, sonorant features, presonorant features, and post-sonorant features.
6) The *USPS dataset* is one of the standard datasets for handwritten digit recognition, it consists of 9298 grayscale images with size 16 × 16, which are normalized to $[-1, 1]$, and these images are categorized into 10 digits (integers in ranges 0–9).
7) The *Breast dataset* is used in different research papers on gene expression, it includes 77 samples represented with expression levels of 4869 genes, and these samples are categorized into two categories, where each sample is indicated whether it came from a tumor biopsy or not.
8) The *Prostate dataset* is used in different research papers on gene expression, it consists of 102 samples, which was represented by 6033 genes, and these samples are categorized into two categories, where each sample is indicated whether it came from a tumor biopsy or not.
9) The *ORL dataset* was taken at different times, varying the lighting, facial expressions, and facial details. All the images were taken at a dark homogeneous background with the subjects in an upright, frontal position. It contains ten different images of size 32×32 from 40 distinct subjects.
10) The *Segment dataset* is an image segmentation database similar to a database already present in the repository (Image segmentation database), which consists of 2310 images that were drawn randomly from a database of seven outdoor images, and these images were hand segmented to create a classification for every pixel, each sample is a 3 × 3 region.

The characteristics of these 10 datasets are summarized in Table I.

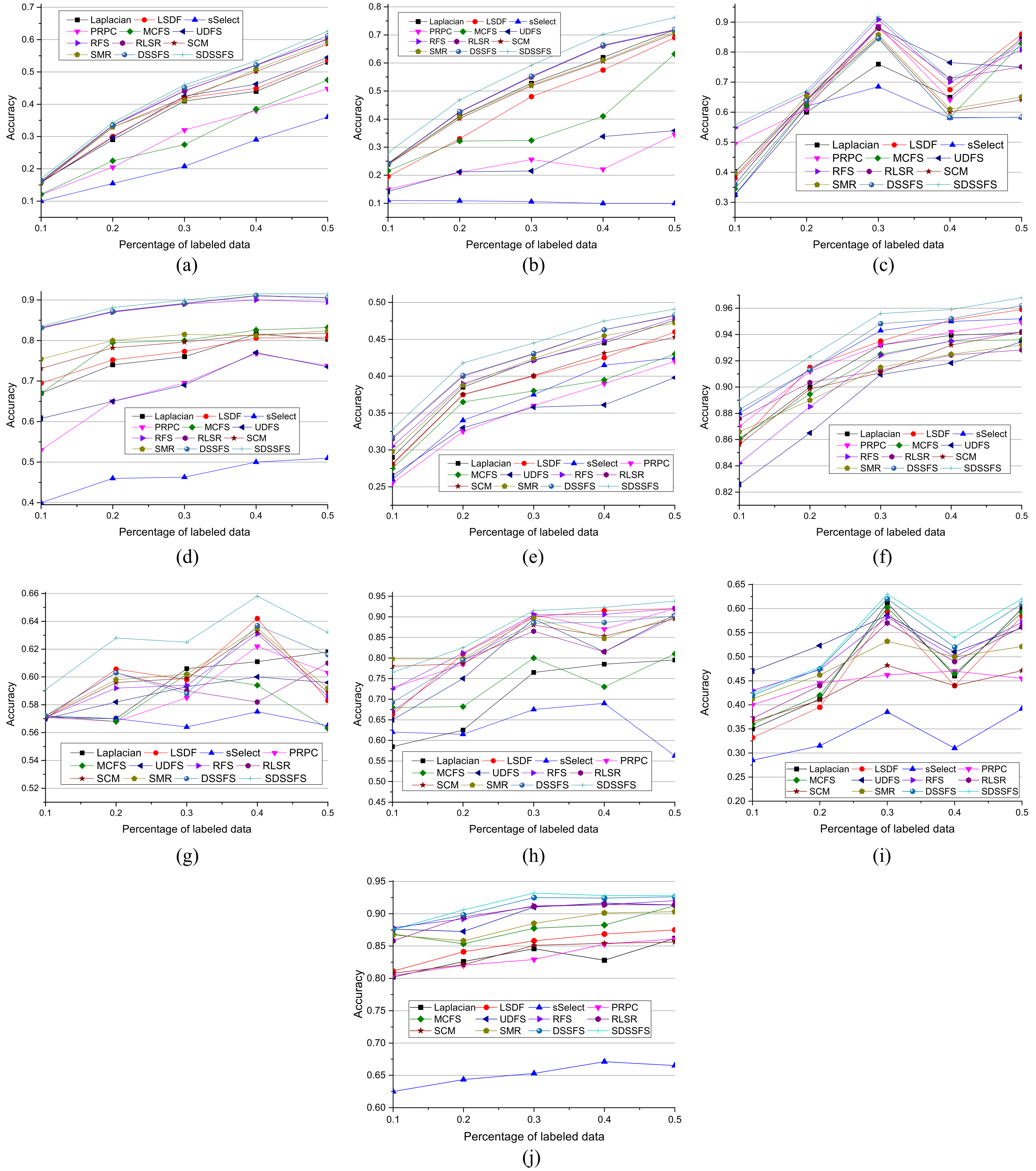
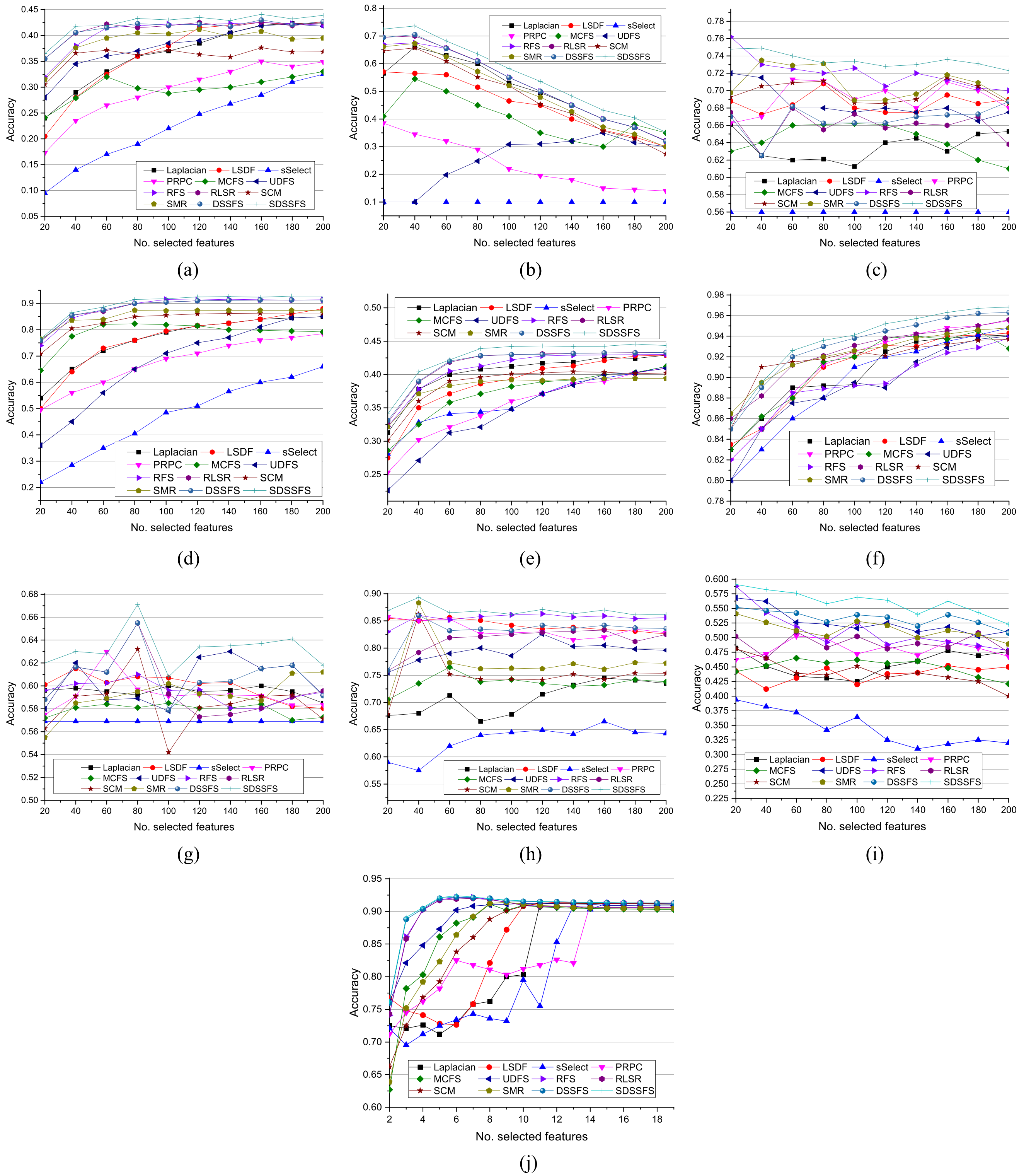


Fig. 2. Comparison results of the average accuracies versus the percentage of labeled records by 12 feature selection methods on ten datasets. (a) Results on the Binalpha dataset. (b) Results on the CNAE-9 dataset. (c) Results on the Colon dataset. (d) Results on the Isolet dataset. (e) Results on the CS dataset. (f) Results on the Usps dataset. (g) Results on the Breast dataset. (h) Results on the Prostate dataset. (i) Results on the ORL dataset. (j) Results on the Segment dataset.

B. Comparison Scheme

We compared SDSSFS with 12 state-of-the-art feature selection methods, including five semisupervised feature selection methods: 1) sSelect [13]; 2) LSDF [14]; 3) SCM [26]; 4) SMR [27]; and 5) RRPC [16], three unsupervised feature selection methods: 1) LS [9]; 2) UDFS [28]; and 3) MCFS [29], and two supervised feature selection methods: 1) RFS [7] and 2) DSSFS [24].[1] Baseline is implemented by training SVM with all features. We set the regularization parameter γ of LS, LSDF, RFS, UDFS, sSelect, RLSR,

[1] The early version in our AAAI abstract paper, which is a special case of SDSSFS with $p = 1$.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

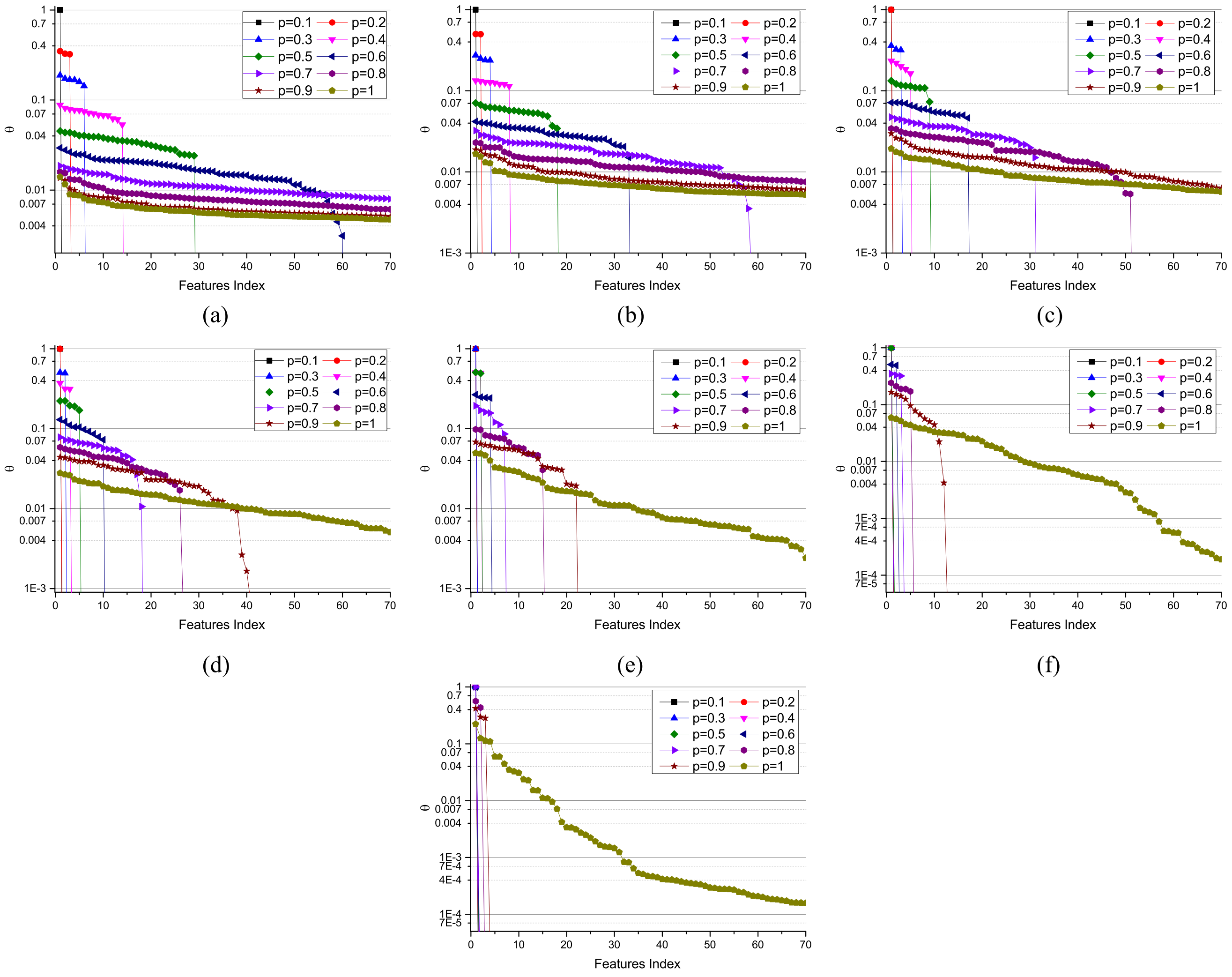
Fig. 3. Comparison results of the average accuracies versus the number of selected features by 12 methods on Ten benchmark datasets, in which only 40% of labels in each dataset are used for training. (a) Results on the Binalpha dataset. (b) Results on the CNAE-9 dataset. (c) Results on the Colon dataset. (d) Results on the Isolet dataset. (e) Results on the CS dataset. (f) Results on the Usps dataset. (g) Results on the Breast dataset. (h) Results on the Prostate dataset. (i) Results on the ORL dataset. (j) Results on the Segment dataset.

DSSFS, and SDSSFS as $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^2, 10^3\}$, λ of sSelect to $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$. The sparsity parameter p in SDSSFS was set as ten values $\{0.1, 0.2, \ldots, 1.0\}$. For LS, LSDF, sSelect, and UDFS, the projection dimensions of these datasets are shown in Table I.

For each of the eight benchmark datasets, we randomly selected training samples with the given ratio $\{10\%, 20\%, 30\%, 40\%, 50\%\}$ and then used the remaining samples as the test data. The test data were also used as the unlabeled data for the semisupervised methods. For

TABLE I
CHARACTERISTICS OF TEN BENCHMARK DATASETS

Name	#Samples	#Features	#Classes	#Projection dimensions	#Selected features
Binalpha	1404	320	36	[40,80,...,200]	[20,40,...,200]
CNAE-9	1080	856	9	[40,80,...,200]	[20,40,...,200]
Colon	62	2000	2	[40,80,...,200]	[20,40,...,200]
CS	8641	256	101	[40,80,...,200]	[20,40,...,200]
Isolet	7797	617	26	[40,80,...,200]	[20,40,...,200]
USPS	9298	256	10	[40,80,...,200]	[20,40,...,200]
Breast	77	4869	5	[40,80,...,200]	[20,40,...,200]
Prostate	102	6033	2	[40,80,...,200]	[20,40,...,200]
ORL	400	1024	10	[40,80,...,200]	[20,40,...,200]
Segment	2310	19	7	[2,3,...,19]	[2,3,...,19]



(a)

(b)

(c)

(d)

(e)

(f)

(g)

Fig. 4. Relationship between θ (100 highest values) and p, γ on the USPS dataset. (a) $\gamma = 0.001$. (b) $\gamma = 0.01$. (c) $\gamma = 0.1$. (d) $\gamma = 1$. (e) $\gamma = 10$. (f) $\gamma = 100$. (g) $\gamma = 1000$.

unsupervised methods, we used the training examples without labels and the test data as input. For the selected features, we trained SVM to predict the unlabeled data, and the average accuracies were computed.

C. Results and Analysis

The average accuracies of 12 methods versus the percentage of labeled samples are shown in Fig. 2, and the average accuracies are summarized in Table II. In summary, the more labeled data we have, the higher accuracy we can achieve. SDSSFS outperformed the other methods excluding DSSFS in accuracy on most datasets, especially on the *CNAE-9* and *Breast* datasets. To be specific, SDSSFS achieves a nearly 10% average improvement on the *CNAE-9* dataset, compared to the second-best methods RFS and RLSR. On the *Breast* dataset, SDSSFS achieves a nearly 5% average improvement compared to the second-best method SMR. SDSSFS also achieves good performance on the rest datasets on average. We also

TABLE II

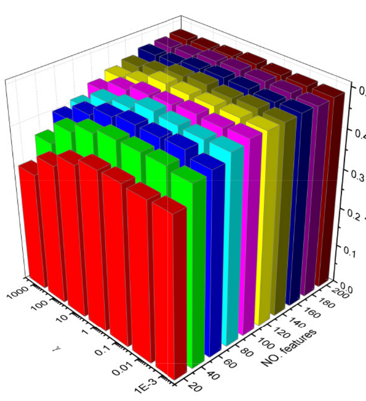
AVERAGE CLASSIFICATION ACCURACIES OF 12 FEATURE SELECTION METHODS ON TEN BENCHMARK DATASETS (THE BEST TWO RESULTS ON EACH DATASET ARE HIGHLIGHTED IN BOLD)

Algorithm	Binalpha	CNAE-9	Colon	Isolet	CS	USPS	Breast	Prostate	ORL	Segment
LS	0.37 ± 0.14	0.50 ± 0.18	0.64 ± 0.19	0.76 ± 0.05	0.41 ± 0.07	0.92 ± 0.01	0.60 ± 0.02	0.71 ± 0.10	0.39 ± 0.02	0.79 ± 0.02
LSDF	0.37 ± 0.14	0.45 ± 0.19	0.69 ± 0.20	0.77 ± 0.04	0.39 ± 0.06	0.91 ± 0.01	0.60 ± 0.05	0.85 ± 0.11	0.40 ± 0.03	0.81 ± 0.02
sSelect	0.22 ± 0.10	0.10 ± 0.15	0.56 ± 0.13	0.47 ± 0.04	0.36 ± 0.06	0.91 ± 0.01	0.57 ± 0.01	0.64 ± 0.04	0.29 ± 0.02	0.62 ± 0.02
PRPC	0.29 ± 0.13	0.24 ± 0.07	0.70 ± 0.16	0.68 ± 0.09	0.35 ± 0.06	0.90 ± 0.01	0.59 ± 0.02	0.84 ± 0.08	0.41 ± 0.02	0.80 ± 0.02
MCFS	0.30 ± 0.14	0.38 ± 0.15	0.64 ± 0.20	0.79 ± 0.06	0.37 ± 0.06	0.93 ± 0.01	0.58 ± 0.02	0.74 ± 0.06	0.42 ± 0.01	0.83 ± 0.02
UDFS	0.38 ± 0.15	0.26 ± 0.09	0.69 ± 0.18	0.68 ± 0.05	0.34 ± 0.05	0.75 ± 0.01	0.59 ± 0.01	0.80 ± 0.10	0.42 ± 0.02	0.85 ± 0.02
RFS	0.41 ± 0.17	**0.52** ± 0.18	**0.72** ± 0.14	0.88 ± 0.03	0.41 ± 0.06	0.94 ± 0.01	0.59 ± 0.02	**0.86** ± 0.09	0.43 ± 0.02	0.86 ± 0.02
RLSR	**0.42** ± 0.17	**0.52** ± 0.19	0.66 ± 0.18	0.89 ± 0.03	0.42 ± 0.07	**0.95** ± 0.02	0.59 ± 0.02	0.82 ± 0.10	0.43 ± 0.02	0.86 ± 0.02
SCM	0.40 ± 0.14	0.49 ± 0.09	0.62 ± 0.16	0.79 ± 0.07	0.39 ± 0.06	0.90 ± 0.02	0.60 ± 0.02	0.84 ± 0.10	0.41 ± 0.02	0.81 ± 0.02
SMR	0.41 ± 0.13	0.50 ± 0.10	0.64 ± 0.19	0.80 ± 0.08	0.41 ± 0.06	0.91 ± 0.01	**0.61** ± 0.01	0.83 ± 0.02	0.44 ± 0.02	0.85 ± 0.02
DSSFS	**0.42** ± 0.11	**0.52** ± 0.19	0.67 ± 0.19	**0.90** ± 0.01	**0.43** ± 0.07	**0.94** ± 0.01	**0.61** ± 0.02	0.83 ± 0.04	**0.45** ± 0.02	**0.87** ± 0.02
SDSSFS	**0.43** ± 0.17	**0.57*** ± 0.19	**0.74** ± 0.12	**0.91*** ± 0.03	**0.44*** ± 0.07	**0.94** ± 0.01	**0.64*** ± 0.04	**0.87*** ± 0.07	**0.47*** ± 0.02	**0.88** ± 0.02

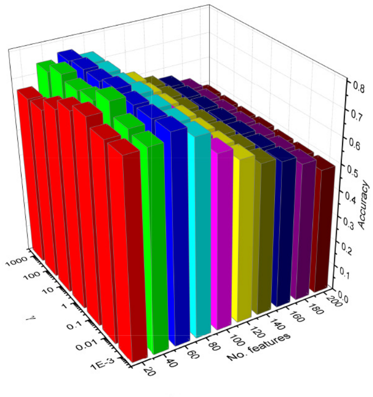
TABLE III

AVERAGE CLASSIFICATION ACCURACIES OF 12 FEATURE SELECTION METHODS ON TEN BENCHMARK DATASETS, IN WHICH ONLY 40% OF LABELS IN EACH DATASET ARE USED FOR TRAINING (THE BEST TWO RESULTS ON EACH DATASET ARE HIGHLIGHTED IN BOLD)

$Algorithm$	Binalpha	CNAE-9	colon	Isolet	CS	USPS	Breast	Prostate	ORL	Segment
LS	0.45 ± 0.14	0.62 ± 0.18	0.65 ± 0.16	0.81 ± 0.05	0.44 ± 0.07	0.94 ± 0.01	0.61 ± 0.01	0.79 ± 0.11	0.46 ± 0.02	0.83 ± 0.03
LSDF	0.45 ± 0.12	0.57 ± 0.16	0.68 ± 0.11	0.81 ± 0.04	0.42 ± 0.05	0.92 ± 0.01	**0.68** ± 0.02	**0.92** ± 0.10	0.44 ± 0.03	0.87 ± 0.03
sSelect	0.29 ± 0.11	0.09 ± 0.12	0.58 ± 0.11	0.50 ± 0.03	0.40 ± 0.04	0.92 ± 0.01	0.57 ± 0.01	0.70 ± 0.11	0.31 ± 0.02	0.67 ± 0.02
PRPC	0.38 ± 0.12	0.23 ± 0.19	0.63 ± 0.13	0.77 ± 0.03	0.40 ± 0.04	0.91 ± 0.01	0.62 ± 0.02	0.87 ± 0.12	0.47 ± 0.02	0.85 ± 0.03
MCFS	0.39 ± 0.13	0.41 ± 0.11	0.58 ± 0.12	0.83 ± 0.04	0.40 ± 0.05	0.94 ± 0.02	0.59 ± 0.01	0.73 ± 0.11	0.46 ± 0.02	0.88 ± 0.01
UDFS	0.47 ± 0.11	0.34 ± 0.13	**0.76** ± 0.11	0.69 ± 0.04	0.36 ± 0.05	0.74 ± 0.01	0.61 ± 0.01	0.81 ± 0.11	0.51 ± 0.02	0.91 ± 0.03
RFS	0.52 ± 0.11	0.65 ± 0.13	0.69 ± 0.16	0.90 ± 0.03	0.45 ± 0.05	0.95 ± 0.01	0.63 ± 0.02	0.91 ± 0.11	0.50 ± 0.02	0.91 ± 0.02
RLSR	0.52 ± 0.10	0.65 ± 0.16	0.71 ± 0.14	0.91 ± 0.05	**0.46** ± 0.05	0.95 ± 0.02	0.58 ± 0.01	0.81 ± 0.11	0.49 ± 0.02	0.91 ± 0.02
SCM	0.39 ± 0.12	0.48 ± 0.16	0.69 ± 0.12	0.83 ± 0.03	0.38 ± 0.05	0.88 ± 0.01	0.58 ± 0.02	0.75 ± 0.10	0.44 ± 0.03	0.85 ± 0.03
SMR	0.40 ± 0.10	0.50 ± 0.17	0.70 ± 0.11	0.85 ± 0.04	0.38 ± 0.03	0.92 ± 0.02	0.59 ± 0.01	0.77 ± 0.11	0.50 ± 0.02	0.90 ± 0.02
DSSFS	**0.53** ± 0.11	**0.66** ± 0.12	0.71 ± 0.12	**0.92** ± 0.02	0.46 ± 0.06	**0.96** ± 0.02	0.64 ± 0.02	0.91 ± 0.10	**0.52** ± 0.02	**0.92** ± 0.02
SDSSFS	**0.54** ± 0.12	**0.69*** ± 0.13	**0.77** ± 0.11	**0.93*** ± 0.02	**0.47** ± 0.04	**0.97** ± 0.01	**0.70*** ± 0.01	**0.93*** ± 0.11	**0.53** ± 0.02	**0.93*** ± 0.02



(a)



(b)

Fig. 5. Accuracy versus γ and the number of selected features. (a) Results of the CS dataset. (b) Results on the CNAE-9 dataset.

notice that SDSSFS defeats RLSR on most datasets. Since SDSSFS is an extension to RLSR, these results indicate that the introduction of the ϵ-dragging technique indeed improves the performance of feature selection.

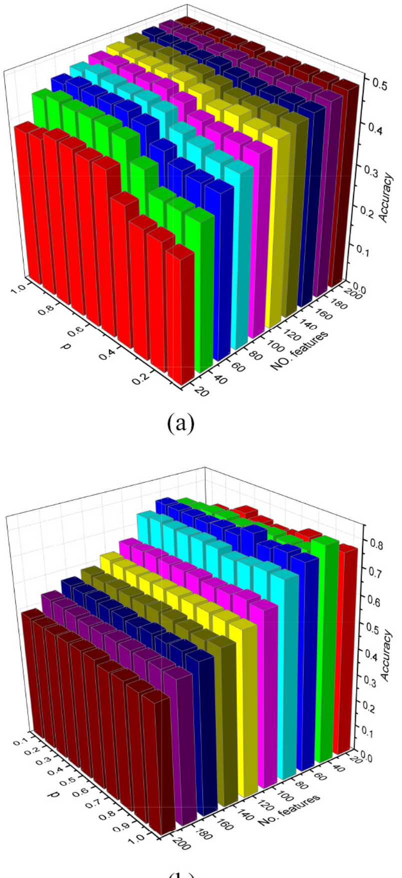
Moreover, SDSSFS outperforms DSSFS on all datasets, which is a special case of SDSSFS. For example, SDSSFS achieves a greater than 10% average improvement on the *Colon* dataset. These results indicate that the introduction of the feasible $\ell_{2,p}$ norm indeed improves the performance of feature selection.

We show the average accuracies of 12 methods versus the percentage of labeled samples are shown in Fig. 3, and the average accuracies are summarized in Table III. We can observe from Fig. 3 that with the increase of the number of selected features, most feature selection methods show performance increasing tendency on the Binalpha, Isolet, CS, and Usps datasets and performance decreasing tendency on the CNAE-9 dataset. On the other datasets, the performances of most feature selection methods do not change too much with the increase of the number of selected features. Comparing the results in Table III with the results in Table II, we can draw the same conclusions that the introduction of the ϵ-dragging technique and the feasible $\ell_{2,p}$ norm indeed improves the performance of our proposed feature selection method.

D. Parameter Sensitivity Study

In SDSSFS, both γ and p affect the row sparsity of the projection matrix $\mathbf{W}$. From (33), we can see that minimizing problem (33) with larger γ or smaller p prefers to make the learned rows of $\mathbf{W}$ sparser. In this experiment, we investigate the two parameters γ and p in SDSSFS.

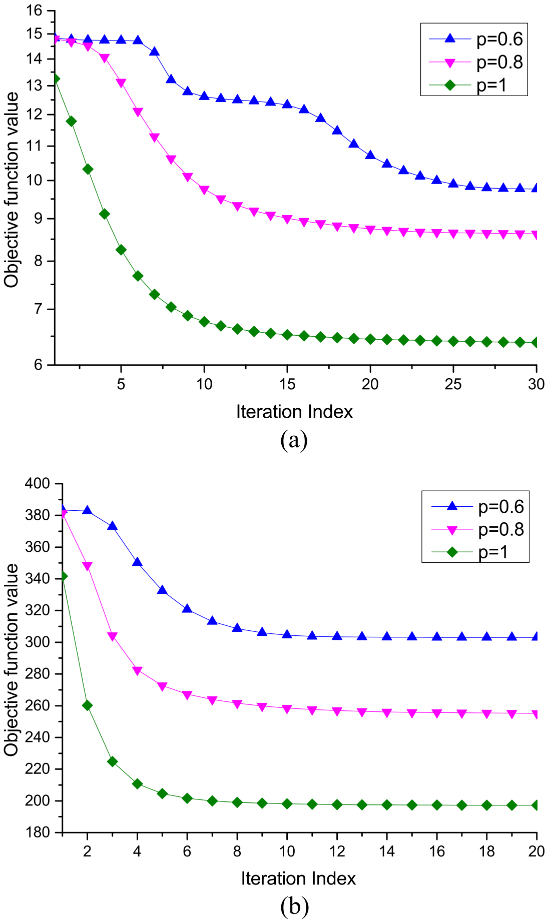
We first study the effect of γ and p on the performance of SDSSFS. The relationship between θ and p, γ is shown in Fig. 4, in which γ was set as seven values varying from 0.001 to 1000, and p was set as ten values varying from 0.1 to 1. We only show 70 highest θ values on the CNAE-9 dataset for brevity. From this figure, we can see that the high weights in



(a)

(b)

Fig. 6. Accuracy versus p and the number of selected features. (a) Results of the CS dataset. (b) Results on the CNAE-9 dataset.



(a)

(b)

Fig. 7. Objective function curves of SDSSFS. (a) Results on the Breast2 dataset. (b) Results on the CNAE-9 dataset.

θ are concentrated on fewer features with the increase of γ and the decrease of p. In real applications, we wish θ only consists of a specific number of features with high weights, thus we can select p and γ according to the distribution of θ.

In SDSSFS, γ and p can be used to control the row sparsity of $\mathbf{W}$. Varying the values of γ and the number of selected features, the average classification accuracies on the CS and CNAE-9 datasets are shown in Fig. 5. As the number of selected features increased, the accuracies of SDSSFS decreased on the CNAE-9 dataset but increased on the CS dataset. On the CNAE-9 dataset, when we selected a small number of features, SDSSFS produced better results with small γ in order to preserve enough features. As we selected more features, SDSSFS produced better results with high γ which forces only a few important features to be selected.

We also show the average classification accuracies versus p and the number of selected features on the CS and CNAE-9 datasets in Fig. 6. On the CS dataset, SDSSFS produced better results with high p when a large number of features were selected, but with low p when a small number of features were selected. In real-life applications, we can perform a hierarchy grid search to select the proper γ and p for better results.

E. Convergence Study

In this section, we study the convergence speed of SDSSFS. For simplicity, we only show the results on the Breast2 and CNAE-9 datasets. The objective function curves are shown in Fig. 7. From this figure, we can see that SDSSFS converges very fast.

V. Conclusion

We have proposed a novel embedded semisupervised feature selection method SDSSFS, in which the ϵ-dragging for the supervised task is extended to a semisupervised task to enlarge the distance between classes in order to obtain a discriminative solution. Moreover, we have proved that the sparse $\ell_{2,p}$ norm is implicitly used as regularization in the new model. Therefore, we can produce a sparser projection matrix by setting smaller p. Experimental results on ten datasets have demonstrated the superior performance of the new method. In future work, we will improve this method such that it can handle large-scale data.

Appendix
Efficient Algorithm to Solve Problem (31)

Consider the following general problem:

$$\min_{\mathbf{y}\in\mathbb{R}^{c\times 1},\mathbf{y}\geq 0,\mathbf{y}^T\mathbf{1}=1} \frac{1}{2}\|\mathbf{a}\circ\mathbf{y}-\mathbf{e}\|_2^2 \tag{34}$$

where $\mathbf{e} \in \mathbb{R}^{c\times 1} > 0$.

It is a quadratic program and is strictly convex, so there is a unique solution, which we denote by $\hat{\mathbf{y}}$. In the following, we propose a simple yet efficient algorithm to solve it.

The Lagrangian of the above problem is

$$\frac{1}{2}\|\mathbf{a} \circ \mathbf{y} - \mathbf{e}\|_2^2 - \chi(\mathbf{y}^T\mathbf{1} - 1) - \beta^T\mathbf{y} \tag{35}$$

where χ and $\beta = [\beta_1, \ldots, \beta_c]^T$ are the Lagrangian multipliers. At the optimal solution $\hat{\mathbf{y}}$, the following KKT conditions hold for each $j \in [1, c]$

$$\begin{cases} a_j^2\hat{y}_j - a_je_j - \hat{\chi} - \hat{\beta}_j = 0 & (36) \\ \sum_{j=1}^{c} \hat{y}_j = 1 & (37) \\ \hat{y}_j \geq 0 & (38) \\ \hat{\beta}_j \geq 0 & (39) \\ \hat{\beta}_j\hat{y}_j = 0 & (40) \end{cases}$$

where $\hat{\chi}$ and $\hat{\beta}$ are optimal solution of χ and β.

According to (36), we have

$$\hat{y}_j = \frac{a_je_j + \hat{\chi} + \hat{\beta}_j}{a_j^2}. \tag{41}$$

From (37), we have

$$\hat{\chi} = \frac{1 - \sum_{j=1}^{c} \frac{a_je_j + \hat{\beta}_j}{a_j^2}}{\sum_{j=1}^{c} \frac{1}{a_j^2}}. \tag{42}$$

Substituting (36) gives

$$\hat{y}_j = u_j + h_j\hat{\beta}_j - \bar{h}_j\bar{\beta} \tag{43}$$

where u_j is defined as

$$u_j = h_ja_je_j + \bar{h}_j\left(1 - \sum_{l=1}^{c} h_la_le_l\right) \tag{44}$$

and

$$\bar{\beta} = \sum_{j=1}^{c} h_j\hat{\beta}_j \tag{45}$$

and

$$h_j = \frac{1}{a_j^2} > 0 \tag{46}$$

and

$$\bar{h}_i = \frac{h_i}{\sum_{j=1}^{c} h_j}. \tag{47}$$

From (40) and (43), we know that $\hat{\beta}_j = 0$ if $\hat{y}_{jx} = u_j - \bar{h}_j\bar{\beta} \geq 0$ and $\hat{\beta}_j \geq 0$ if $\hat{y}_j = u_j + h_j\hat{\beta}_j - \bar{h}_j\bar{\beta} = 0$ indicating that $u_j - \bar{h}_j\bar{\beta} = -h_j\hat{\beta}_j \leq 0$. Therefore, the optimal solution of $\hat{y}_j$ is

$$\hat{y}_j = \left(u_j - \bar{h}_j\bar{\beta}\right)_+. \tag{48}$$

So we can obtain the optimal solution of $\hat{y}_j$ if we know $\bar{\beta}$. Since $\sum_{j=1}^{c} \hat{y}_j = 1$, we can obtain $\bar{\beta}$ by solving the following root finding problem:

$$f(\bar{\beta}) = \sum_{j=1}^{c}\left(u_j - \bar{h}_j\bar{\beta}\right)_+ - 1 = 0. \tag{49}$$

Note that $\bar{h}_j\bar{\beta} \geq 0, f'(\bar{\beta}) \leq 0$, and $f'(\bar{\beta})$ is a piecewise linear and convex function, we can use the Newton method to find the root of $f(\bar{\beta}) = 0$, that is

$$\bar{\beta}_{t+1} = \bar{\beta}_t - \frac{f(\bar{\beta})}{f'(\bar{\beta})} \tag{50}$$

where $f'(\bar{\beta}) = -\sum_{j=1}^{c} \bar{h}_j\delta(u_j - \bar{h}_j\bar{\beta})$, where $\delta = 1$ if $(u_j - \bar{h}_j\bar{\beta}) > 0$ and $\delta = 0$; otherwise, $(j \in [1, c])$.

References

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

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

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

[4] D. Richard, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York, NY, USA: Wiley-Intersci., 2010.

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

[6] I. Kononenko, "Estimating attributes: Analysis and extensions of RELIEF," in *Proc. ECML*, 1994, pp. 171–182.

[7] 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.

[8] 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.

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

[10] L. Shi, L. Du, and Y. D. Shen, "Robust spectral learning for unsupervised feature selection," in *Proc. IEEE Int. Conf. Data Min.*, Dec. 2014, pp. 977–982.

[11] F. Nie, W. Zhu, and X. Li, "Unsupervised feature selection with structured graph optimization," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1302–1308.

[12] Y. Luo, D. Tao, C. Xu, D. Li, and C. Xu, "Vector-valued multi-view semi-supervised learning for multi-label image classification," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 647–653.

[13] Z. Zhao and H. Liu, "Semi-supervised feature selection via spectral analysis," in *Proc. SIAM Int. Conf. Data Min.*, 2007, pp. 641–646. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9781611972771.75

[14] J. Zhao, K. Lu, and X. He, "Locality sensitive semi-supervised feature selection," *Neurocomputing*, vol. 71, nos. 10–12, pp. 1842–1849, 2008.

[15] G. Doquire and M. Verleysen, "A graph Laplacian based approach to semi-supervised feature selection for regression problems," *Neurocomputing*, vol. 121, pp. 5–13, Dec. 2013.

[16] 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.

[17] J. Ren, Z. Qiu, W. Fan, H. Cheng, and P. S. Yu, "Forward semi-supervised feature selection," in *Proc. 12th Pac. Asia Conf. Knowl. Disc. Data Min.*, 2008, pp. 970–976.

[18] X. Li, Y. Wang, Z. Zhang, R. Hong, Z. Li, and M. Wang, "RMoR-Aion: Robust multi-output regression by simultaneously alleviating input and output noises," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 3, pp. 1351–1364, Mar. 2021.

[19] Z. Xu, I. King, M. R.-T. Lyu, and R. Jin, "Discriminative semi-supervised feature selection via manifold regularization," *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1033–1047, Jul. 2010.

[20] 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.*, Aug. 2017, pp. 1525–1531.

[21] Z. Lai, D. Mo, W. K. Wong, Y. Xu, D. Miao, and D. Zhang, "Robust discriminant regression for feature extraction," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2472–2484, Aug. 2018.

[22] Z. Zhang, M. Zhao, F. Li, L. Zhang, and S. Yan, "Robust alternating low-rank representation by joint Lp-and L2,p-norm minimization," *Neural Netw.*, vol. 96, pp. 55–70, Dec. 2017.

[23] G. Lan, C. Hou, F. Nie, T. Luo, and D. Yi, "Robust feature selection via simultaneous sapped norm and sparse regularizer minimization," *Neurocomputing*, vol. 283, pp. 228–240, Mar. 2018.

[24] G. Yuan, X. Chen, C. Wang, F. Nie, and L. Jing, "Discriminative semi-supervised feature selection via rescaled least squares regression-supplement," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 8177–8178.

[25] S. Xiang, F. Nie, G. Meng, C. Pan, and C. Zhang, "Discriminative least squares regression for multiclass classification and feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 11, pp. 1738–1754, Nov. 2012.

[26] G. Lan, C. Hou, F. Nie, T. Luo, and D. Yi, "Robust feature selection via simultaneous capped norm and sparse regularizer minimization," *Neurocomputing*, vol. 283, pp. 228–240, Mar. 2018.

[27] C. Hou, Y. Jiao, F. Nie, T. Luo, and Z. H. Zhou, "Two dimensional feature selection by sparse matrix regression," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4255–4268, Sep. 2017.

[28] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "$\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 1589–1594.

[29] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2010, pp. 333–342.

Chen Wang is currently pursuing the master's degree with the College of Computer Science and Software, Shenzhen University, Shenzhen, China.

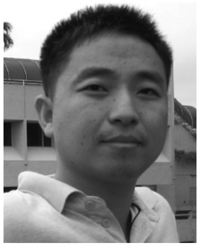
His research interests include clustering and feature selection.

Xiaojun Chen (Member, IEEE) received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2011.

He is currently an Associate Professor with the College of Computer Science and Software, Shenzhen University, Shenzhen, China. His research interests include subspace clustering, topic model, feature selection, and massive data mining.

Guowen Yuan is currently pursuing the master's degree with the College of Computer Science and Software, Shenzhen University, Shenzhen, China.

His research interests include clustering and feature selection.



Feiping Nie (Member, IEEE) received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2009.

He has published over 100 papers in the following top journals and conferences: the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the *International Journal of Computer Vision*, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON NEURAL NETWORKS, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, the *ACM Transactions on Knowledge Discovery From Data*, *Bioinformatics*, ICML, NIPS, KDD, IJCAI, AAAI, ICCV, and CVPR ACM MM. His papers have been cited more than 5000 times (Google scholar). His research interests are machine learning and its applications, such as pattern recognition, data mining, computer vision, image processing, and information retrieval.

Dr. Nie is currently serving as an associate editor or a PC member for several prestigious journals and conferences in the related fields.

Min Yang received the Ph.D. degree from the University of Hong Kong, Hong Kong, in 2017.

She is currently an Assistant Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. Her current research interests include machine learning and natural language processing.