Semi-Supervised Feature Selection via Sparse Rescaled Linear Square Regression

Xiaojun Chen, *Member, IEEE*, Guowen Yuan, Feiping Nie, and Zhong Ming

Abstract—With the rapid increase of the data size, it has increasing demands for selecting features by exploiting both labeled and unlabeled data. In this paper, we propose a novel semi-supervised embedded feature selection method. The new method extends the least square regression model by rescaling the regression coefficients in the least square regression with a set of scale factors, which is used for evaluating the importance of features. An iterative algorithm is proposed to optimize the new model. It has been proved that solving the new model is equivalent to solving a sparse model with a flexible and adaptable $\ell_{2,p}$ norm regularization. Moreover, the optimal solution of scale factors provides a theoretical explanation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to evaluate the importance of features. Experimental results on eight benchmark data sets show the superior performance of the proposed method.

Index Terms—Feature selection, semi-supervised feature selection, sparse feature selection, least square regression

1 Introduction

High-dimensional data presents a big challenge to supervised learning due to the "curses of high-dimensionality" [1]. For example, a gene expression data which measures the expression levels of genes in experiments, often consists of thousands of genes. In classifying such data, learnt models tend to be overfitting and of poor generalization ability. In such high-dimensional data, it is often found that only a small portion of features are highly correlated to the classes, while most features are irrelevant. To deal with such problem, feature selection is one effective means to identify useful features from high-dimensional data [2]. During the past ten years, many feature selection methods have been proposed and various studies show that feature selection can help to remove irrelevant features without performance deterioration [3], [4], [5].

Feature selection can be conducted in a supervised or unsupervised manner, in terms of whether the label information is available. In supervised feature selection, feature relevance can be evaluated according to the correlations of the features with the class labels, e.g., Fisher score [6], Relief-F [7], [8], RFS [9] and GRM [10]. In unsupervised feature selection, without label information, feature relevance can be evaluated by feature dependency or similarity, e.g., Laplacian Score [11], RSFS [12] and SOGFS [13]. With the rapid increase of the data size, it is often costly to obtain labeled data [14]. 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 training to ensure the effectiveness of learnt models. Related researches are becoming hot spots in many machine learning fields, such as image annotations and categorizations. Under such circumstances, it is desirable to develop feature selection methods that are capable of exploiting both labeled and unlabeled data. The task of conducting feature selection from mixed labeled and unlabeled data is called "semi-supervised feature selection".

Various semi-supervised feature selection methods have been proposed during the past ten years. However, most of them are filter-based method that scores the features with a ranking criterion regardless of the model [15], [16], [17], [18]. As argued in [19], the filter-based feature selection method could discard important features that are less informative by themselves but are informative when combined with other features. Ren et al. proposed a wrapper-type forward semi-supervised feature selection framework [20], which performs supervised sequential forward feature selection on both labeled and unlabeled data. But such method is usually time consuming for high-dimensional data because it involves iterative feature subset searching. Embedded semi-supervised methods, which include feature selection as part of the training process, are superior to the other feature selection methods in many respects. Xu et al. proposed a discriminative semi-supervised feature selection method based on the idea of manifold regularization, but their method has high computational complexity of $O(\frac{n^{2.5}}{\epsilon})$ where n is the number of objects and ϵ is a small stopping criterion [21].

Least square regression is a widely-used statistical analysis technique. It has been used for many real-world applications due to its effectiveness for data analysis as well as its completeness in statistics theory. Many variants have been

- *X. Chen, G. Yuan, and Z. Ming are with the College of Computer Science and Software, Shenzhen University, Shenzhen, Guangdong 518060, P.R. China. E-mail: {xjchen, mingz}@szu.edu.cn, gwyuan93@qq.com.*
- *F. Nie is with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, Shanxi 710072, P. R. China. E-mail: feipingnie@gmail.com.*

Manuscript received 22 Feb. 2018; revised 23 Oct. 2018; accepted 1 Nov. 2018. Date of publication 9 Nov. 2018; date of current version 5 Dec. 2019. (Corresponding author: Xiaojun Chen.) Recommended for acceptance by D. Cai. Digital Object Identifier no. 10.1109/TKDE.2018.2879797

developed, including weighted LSR [22], partial LSR [23], ridge regression [24], discriminative LSR [25]. Least regression based feature selection methods usually learn a projection matrix $\mathbf{W}$ and select the important features in the descending order of $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$. However, it lacks of theoretical foundation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to evaluate the importance of features. Recently, Chen et al. proposed a semi-supervised feature selection method RLSR [26], in which a rescaled linear square regression is proposed to extend the 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. However, their method cannot work for more feasible $\ell_{2,p}$ norm where $p \in (0, 1)$.

In this paper, we extend RLSR to Sparse Rescaled Linear Square Regression (SRLSR) for semi-supervised feature selection. The main contributions of our work include:

1) We propose a general sparse rescaled least square regression to obtain more feasible solution. The rescaled linear square regression in [26] is a special case of our model. It has been proved that the new model is equivalent to a sparse model with the $\ell_{2,p}$ norm regularization ($p \in (0, 1]$).
2) To our knowledge, it is the first sparse semi-supervised feature selection method that uses $\ell_{2,p}$ norm as implicit regularization term. We can obtain more sparse regression coefficients by setting smaller p for the new method.
3) The optimal solution of the scale factors in the new model provides a theoretical explanation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to evaluate the importance of features for the $\ell_{2,p}$-norm regularized least square regression.
4) Comprehensive experiments on 8 benchmark data sets show the superior performance the proposed approach in comparison with semi-supervised, supervised and unsupervised feature selection methods.

The rest of this paper is organized as follows. Section 2 presents the notations and the definition of norms used in this paper, and gives a brief review of related work. In Section 3, the semi-supervised feature selection method SRLSR is proposed. The experimental results are reported in Sections 4. Conclusions and future work are given in Section 5.

2 Notations and Related Work

In this section, we summarize the notations and definitions used in this paper, and give a brief review of related work.

2.1 Notations and Definitions

Matrices are written as boldface uppercase letters. Vectors are written as boldface lowercase letters. For matrix $\mathbf{M} = (m_{ij})$, its ith row is denoted as $\mathbf{m}^i$, and its jth column is denoted by $\mathbf{m}_j$. The Frobenius norm of the matrix $\mathbf{M} \in \mathbb{R}^{n \times m}$ is defined as

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

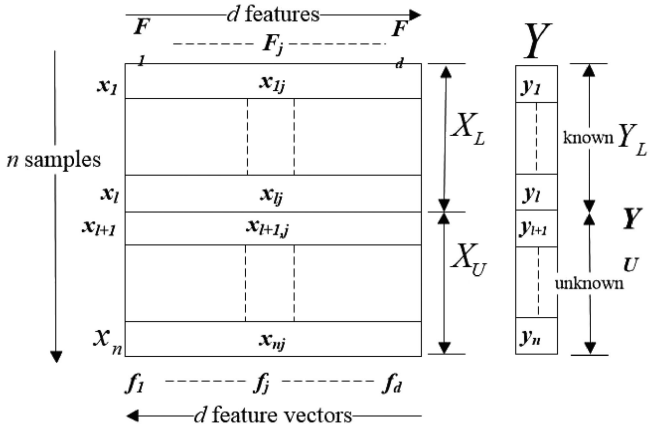


Fig. 1. The data used in semi-supervised feature selection task.

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

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

If $p = 1$, it becomes the commonly-used $\ell_{2,1}$ norm.

2.2 Semi-Supervised Feature Selection

In semi-supervised scenario, we assume that a data set $\mathbf{X} \in \mathbb{R}^{d \times n}$ consists of two subsets with c classes (see Fig. 1): a set of l labeled objects $\mathbf{X}_L = (\mathbf{x}_1, \ldots, \mathbf{x}_l)$ which are associated with class labels $\mathbf{Y}_L = \{\mathbf{y}_1, \ldots, \mathbf{y}_l\}^T \in \mathbb{R}^{l \times c}$, and 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. In the feature view, $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_d\}$, where $\mathbf{f}_i$ denotes the ith feature vectors in $\mathbf{X}$. $\mathbf{y}_i \in \mathbb{R}^c (1 \le i \le l)$ is usually a binary vector defined as

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

Although many semi-supervised feature selection methods have been proposed during the past ten years, most of them are filter-based which score the features with a ranking criterion regardless of the model [15], [16], [17], [18]. For example, Zhao et al. proposed a semi-supervised feature selection algorithm (sSelect) based on spectral analysis [15]. In their method, 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. sSelect computes a score s_j for the jth feature as follows

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

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, $\hat{g} \in \mathbb{R}^{l\times c}$ 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 $O(dn^2)$.

Zhao et al. proposed a locality sensitive semi-supervised feature selection (LSDF) based on Laplacian criteria [16]. The new method first construct a within-class affinity matrix $\mathbf{W}$, in which w_{ij} is defined as

$$w_{ij} = \begin{cases} \gamma & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ share the same label} \\ 1 & \text{if if } \mathbf{x}_i \text{ or } \mathbf{x}_j \text{ is unlabeled, but } \mathbf{x}_i \in \mathbf{N}_k(\mathbf{x}_j) \\ & \text{or } \mathbf{x}_j \in \mathbf{N}_k(\mathbf{x}_i) \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

and a between-class affinity matrix $\mathbf{B}$, in which b_{ij} is defined as

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

where γ is a suitable constant. Then the importance score for the jth feature can be computed as

$$\mathbf{L}_j = \frac{\mathbf{f}_j^T \mathbf{L}_b \mathbf{f}_j}{\mathbf{f}_j^T \mathbf{L}_w \mathbf{f}_j}, \tag{8}$$

where $\mathbf{f}_j \in \mathbb{R}^{n\times 1}$ is the jth feature vector, $\mathbf{L}_w = \mathbf{D}_w - \mathbf{W}$ $(\mathbf{D}_w = diag(\mathbf{W1}))$ and $\mathbf{L}_b = \mathbf{D}_b - \mathbf{B}$ $(\mathbf{D}_b = diag(\mathbf{B1}))$ are graph Laplacians. LSDF also has a computational complexity of $O(dn^2)$.

Doquire et al. proposed a semi-supervised Laplacian score (SSLS) for semi-supervised feature selection [17]. In their method, they first construct an affinity matrix $\mathbf{S}^{sup}$ in which s_{ij}^{sup} is defined as

$$s_{ij}^{sup} = \begin{cases} e^{\frac{-\|\mathbf{y}_i - \mathbf{y}_j\|^2}{t}} & \text{if } \mathbf{x}_i \in \mathbf{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathbf{N}_k(i) \\ 0 & \text{otherwise} \end{cases}, \tag{9}$$

where t is a suitable positive constant and an affinity matrix $\mathbf{S}^{semi}$ in which s_{ij}^{semi} is defined as follows

$$s_{ij}^{semi} = \begin{cases} e^{\frac{-d_{ij}^2}{t}} & \text{if } \mathbf{x}_i \in \mathbf{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathbf{N}_k(i) \\ & \text{and } \mathbf{y}_i \text{ or } \mathbf{y}_j \text{ is unknown} \\ C \times e^{\frac{-d_{ij}^2}{t}} & \text{if } \mathbf{x}_i \in \mathbf{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathbf{N}_k(i) \\ & \text{and } \mathbf{y}_i \text{ and } \mathbf{y}_j \text{ are known} \\ 0 & \text{otherwise,} \end{cases} \tag{10}$$

where C is a positive constant and d_{ij}^2 is defined as

$$d_{ij}^2 = \begin{cases} (y_i - y_j)^2 & \text{if } \mathbf{y}_i \text{ and } \mathbf{y}_j \text{ are known} \\ \frac{\sum_{l=1}^{n}(f_{li} - f_{ij})^2}{n} & \text{otherwise.} \end{cases} \tag{11}$$

Then SSLS measures the importance of the jth feature as follows

$$SSLS_j = \frac{\tilde{\mathbf{f}}_j^T \mathbf{L}^{\mathbf{semi}} \tilde{\mathbf{f}}_j}{\tilde{\mathbf{f}}_j^T \mathbf{D}^{\mathbf{semi}} \tilde{\mathbf{f}}_j} \times SLS_j, \tag{12}$$

where $\tilde{\mathbf{f}}_j = \mathbf{f}_j - \frac{\mathbf{f}_j^T \mathbf{D}^{\mathbf{semi}} \mathbf{1}}{\mathbf{1}^T \mathbf{D}^{\mathbf{semi}} \mathbf{1}} \mathbf{1}$, $\mathbf{L}^{semi} = \mathbf{D}^{\mathbf{semi}} - \mathbf{S}^{semi}$ and SLS_j is defined as

$$SLS_j = \frac{\tilde{\mathbf{f}}_j^T \mathbf{L}^{\mathbf{sup}} \tilde{\mathbf{f}}_j}{\tilde{\mathbf{f}}_j^T \mathbf{D}^{\mathbf{sup}} \tilde{\mathbf{f}}_j}. \tag{13}$$

It can be verified that SSLS also has a computational complexity of $O(dn^2)$.

Xu et al. proposed semi-supervised feature selection based on relevance and redundancy criteria (RRPC) [18]. Assume that we have obtained feature subsets $\mathbf{F}_{k-1}$, which consists of $k-1$ selected features from $\mathbf{F}$. In the next step, the kth feature is selected from feature subsets $\{\mathbf{F} - \mathbf{F}_{k-1}\}$ as follows

$$\mathbf{F}_k = \underset{\mathbf{F}_j \in \mathbf{F} - \mathbf{F}_{k-1}}{\arg\min} \left[\mathbf{P}(\mathbf{F}_j, \mathbf{y}_L) - \frac{1}{k-1} \sum_{\mathbf{F}_i \in \mathbf{F}_{k-1}} \mathbf{P}(\mathbf{F}_j, \mathbf{F}_i) \right], \tag{14}$$

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

However, as argued in [19], the filter-based feature selection method could discard important features that are less informative by themselves but are informative when combined with other features. Ren et al. extended the supervised sequential forward feature selection (SFFS) to take both unlabeled and labeled data into account and proposed an iterative "wrapper-type" forward semi-supervised feature selection framework [20]. 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 method is usually time-consuming because that it involves iterative feature subset searching.

Embedded semi-supervised methods, which include feature selection as part of the training process, are superior to others in many respects. Xu et al. proposed a discriminative semi-supervised feature selection method, namely FS-Manifold, based on the idea of manifold regularization [21]. 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 Eq. (4). Then the projection matrix $\mathbf{W}$ is learned by optimizing the following objective function

$$\begin{aligned} \min_{\mathbf{W},\mathbf{b},\xi} & \left(\frac{\|\mathbf{w}\|_2^2}{2} + C \sum_{i=1}^{l} \xi_i + \frac{\rho}{2} \mathbf{w}^T \mathbf{X}^T \mathbf{L} \mathbf{X} \mathbf{w} \right) \\ s.t.\ & \mathbf{y}_i (\mathbf{w}^T \mathbf{x}_i - b) \geq 1 - \xi_i \ (1 \leq i \leq l) \\ & \xi_i \geq 0 \ (1 \leq i \leq l), \end{aligned} \tag{15}$$

where C and ρ are two parameters, $\mathbf{L} = diag(\mathbf{A1}) - \mathbf{A}$ 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 $O(\frac{n^{2.5}}{\epsilon^2})$ where ϵ is a small stopping criterion.

Chen et al. proposed a semi-supervised feature selection method *RLSR* [26], in which a rescaled linear square regression is proposed to extend the least square regression for feature selection. To rank d features in $\mathbf{F}$ 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 $\Theta_{jj} = \theta_j^{1/2}$. The objective function of RLSR is defined as

$$\begin{aligned} &\min\Big(\left\|\mathbf{X}^T\Theta\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\|\mathbf{W}\|_F^2\Big) \\ &st.\ \mathbf{W}, \mathbf{b}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \end{aligned} \tag{16}$$

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 regularized parameter to control the trade-off between the bias and variance of the estimate.

In addition, some semi-supervised feature selection methods were proposed for specific data. For example, Kong et al. proposed a semi-supervised feature selection algorithm, which finds an optimal set of subgraph features based on branch-and-bound algorithm from labeled and unlabeled graphs [27]. However, it only works for graph data. Han et al. proposed a semi-supervised feature selection method which uses two scatter matrices to capture both the discriminative information and the local geometry structure of labeled and unlabeled training videos [28]. Specifically, a within-class scatter matrix is used to encode discriminative information of labeled training videos and a spline scatter matrix is used to encode data distribution from a local spline regression. An $\ell_{2,1}$-norm is imposed as a regularization term on the transformation matrix to ensure its sparsity.

2.3 Least Square Regression-Based Feature Selection

Least square regression is a widely-used statistical analysis technique. It has been adapted to many real-world situations due to its effectiveness for data analysis as well as its completeness in statistics theory. Many variants have been developed during the past decades, including weighted LSR [22], partial LSR [23], ridge regression [24], discriminative least squares regression [25] and so on. Meanwhile, LSR has been utilized in many machine learning problems, such as discriminative learning, manifold learning, clustering, and so on.

Let $\mathbf{X} \in \mathbb{R}^{d\times n}$ be a data set with n objects $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$. $\mathbf{X}$ is associated with class labels $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}^T$. Suppose $\mathbf{X}$ can be classified into c classes, we have $\mathbf{Y} \in R^{n\times c}$ and $\mathbf{y}_i \in R^{c\times 1}$. The commonly-used regularization for linear regression can be addressed as the following optimization problem

$$\min_{\mathbf{W}}\Big(\left\|X^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\|\mathbf{W}\|_F^2\Big), \tag{17}$$

where $\mathbf{W} \in \mathbb{R}^{d\times c}$ and $\mathbf{b} \in \mathbb{R}^{c\times 1}$ are to be estimated and $\gamma > 0$ is the regularized parameter to control the trade-off between the bias and variance of the estimated $\mathbf{W}$.

Nie et al. proposed a Robust Supervised Feature selection model (RSF) model [9], by minimization $\ell_{2,1}$-norms of both loss of least square regression and regularization term as

$$\min_{\mathbf{W},\mathbf{b}}\Big(\left\|X^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_{2,1} + \gamma\|\mathbf{W}\|_{2,1}\Big). \tag{18}$$

Xiang et al. proposed a discriminative least squares regression (DLSR) for classification and feature selection, which is to optimize the following problem [25]

$$\min_{\mathbf{W},\mathbf{M}}\Big(\left\|X^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y} - \mathbf{B}\odot\mathbf{M}\right\|_F^2 + \gamma\|\mathbf{W}\|_F^2\Big), \tag{19}$$

where $\mathbf{B}$ is a constant matrix generated from $\mathbf{Y}$ in which each element corresponds to a dragging direction, $\mathbf{M}$ is a matrix which performs ε-dragging on each element of $\mathbf{Y}$, $\odot$ is a Hadamard product operator of two matrices.

The existing least regression based feature selection methods learn a projection matrix $\mathbf{W}$ and rank the features according to their importance factors which are estimated as $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$. However, it lacks of theoretical foundation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to rank the features. Interestingly, Chen et al. have proved that problem (16) in RLSR [26] is equivalent to the following problem

$$\min_{\mathbf{W},\mathbf{b},\mathbf{Y}_U\geq 0,\mathbf{Y}_U\mathbf{1}=\mathbf{1}}\Big(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\|\mathbf{W}\|_{2,1}^2\Big), \tag{20}$$

where θ_j can be computed as

$$\theta_j = \frac{\|\mathbf{w}^j\|_2}{\sum_{j=1}^d \|\mathbf{w}^j\|_2}, \tag{21}$$

which provides a theoretical explanation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to rank the features in the $\ell_{2,1}$ norm regularized least square regression.

3 The Proposed Method

According to Eq. (20), we know that the sparse $\ell_{2,1}$ norm is implicitly used in problem (16) of RLSR [26]. However, this method cannot work for more feasible $\ell_{2,p}$ norm where $p \in (0, 1]$. In this paper, we extend problem (16) in order to obtain more sparse solution with feasible $\ell_{2,p}$ norm where $p \in (0, 1)$. We first introduce a parameter $q \in (0, 1]$ and replace the Θ in RLSR as $\Theta_{jj} = \theta_j^{q/2} (q \geq 1)$. Let $\widetilde{\mathbf{W}} = \Theta\mathbf{W}$, then $\mathbf{W} = \Theta^{-1}\widetilde{\mathbf{W}}$. Problem (16) can be rewritten as

$$\begin{aligned} &\min\Big(\left\|\mathbf{X}^T\widetilde{\mathbf{W}} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\left\|\Theta^{-1}\widetilde{\mathbf{W}}\right\|_F^2\Big) \\ &s.t.\ \widetilde{\mathbf{W}}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \end{aligned} \tag{22}$$

which leads to a new problem

$$\begin{aligned} &\min\Big(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\left\|\Theta^{-1}\mathbf{W}\right\|_F^2\Big) \\ &s.t.\ \mathbf{W}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1}, \end{aligned} \tag{23}$$

We can apply the alternative optimization approach to solve problem (23). In the following, we show how to update each of the three variables Θ, $\mathbf{Y}_U$, $\mathbf{W}$ and $\mathbf{b}$.

3.1 Update Θ with $\mathbf{Y}_U$, $\mathbf{W}$, and $\mathbf{b}$ Fixed

If $\mathbf{W}$, $\mathbf{Y}_U$ and $\mathbf{b}$ are fixed, we can obtain the optimal solution of Θ by solving the following problem

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

where θ_j is the jth diagonal element of Θ.

The Lagrangian function of problem (24) is

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

where χ and positive vector τ are Lagrangian multipliers.

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

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

which leads to

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

Substituting θ in Eq. (27) into $\sum_{j=1}^{d}\theta_j-1=0$, we obtain the optimal solution of θ_j as

$$\theta_j=\frac{\|\mathbf{w}^j\|_2^{\frac{2}{q+1}}}{\sum_{h=1}^{d}\|\mathbf{w}^h\|_2^{\frac{2}{q+1}}}. \tag{28}$$

Let $p=\frac{2}{q+1}$, Eq. (28) can be rewritten as

$$\theta_j=\frac{\|\mathbf{w}^j\|_2^{p}}{\sum_{h=1}^{d}\|\mathbf{w}^h\|_2^{p}}. \tag{29}$$

Then the optimal solution of the jth diagonal element in Θ is

$$\Theta_{jj}=\left(\frac{\|\mathbf{w}^j\|_2^{p}}{\sum_{h=1}^{d}\|\mathbf{w}^h\|_2^{p}}\right)^{\frac{1}{p}-0.5}. \tag{30}$$

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

3.2 Update $\mathbf{b}$ with $\mathbf{Y}$, Θ, and $\mathbf{W}$ Fixed

When $\mathbf{Y}$, Θ and $\mathbf{W}$ are fixed, problem (23) becomes

$$\min_{\mathbf{b}}\left\|\mathbf{X}^T\mathbf{W}+\mathbf{1}\mathbf{b}^T-\mathbf{Y}\right\|_F^2. \tag{31}$$

Setting the partial derivative of the above function with respect to $\mathbf{b}$ as 0, we obtain the optimal solution of $\mathbf{b}$ as

$$\mathbf{b}=\frac{1}{n}(\mathbf{Y}^T\mathbf{1}-\mathbf{W}^T\mathbf{X}\mathbf{1}). \tag{32}$$

3.3 Update $\mathbf{W}$ with $\mathbf{b}$, Θ, and $\mathbf{Y_U}$ Fixed

When $\mathbf{b}$, Θ and $\mathbf{Y_U}$ are fixed, problem (23) becomes

$$\min_{\mathbf{W}}\left(\left\|\mathbf{X}^T\mathbf{W}+\mathbf{1}\mathbf{b}^T-\mathbf{Y}\right\|_F^2+\gamma\left\|\Theta^{-1}\mathbf{W}\right\|_F^2\right). \tag{33}$$

Substituting $\mathbf{b}$ in Eq. (32) into Eq. (33), we obtain a new problem as follows

$$\min_{\mathbf{W}}\left[\left\|\mathbf{H}\mathbf{X}^T\mathbf{W}-\mathbf{H}\mathbf{Y}\right\|_F^2+\gamma\left\|\Theta^{-1}\mathbf{W}\right\|_F^2\right), \tag{34}$$

where $\mathbf{H}=\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^T$.

The Lagrangian function of problem (34) is

$$\mathcal{L}(\mathbf{W})=\left\|\mathbf{H}\mathbf{X}^T\mathbf{W}-\mathbf{H}\mathbf{Y}\right\|_F^2+\gamma\left\|\Theta^{-1}\mathbf{W}\right\|_F^2. \tag{35}$$

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

$$\frac{\partial\mathcal{L}(\mathbf{W})}{\partial\mathbf{W}}=2\mathbf{X}\mathbf{H}^T(\mathbf{H}\mathbf{X}^T\mathbf{W}-\mathbf{H}\mathbf{Y})+2\gamma\Theta^{-2}\mathbf{W}=0. \tag{36}$$

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

$$\mathbf{W}=(\mathbf{X}\mathbf{H}^T\mathbf{H}\mathbf{X}^T+\gamma\Theta^{-2})^{-1}\mathbf{X}\mathbf{H}^T\mathbf{H}\mathbf{Y}. \tag{37}$$

Since $\mathbf{H}$ is an idempotent matrix, the optimal solution of $\mathbf{W}$ can be rewritten as

$$\mathbf{W}=(\mathbf{X}\mathbf{H}\mathbf{X}^T+\gamma\Theta^{-2})^{-1}\mathbf{X}\mathbf{H}\mathbf{Y}. \tag{38}$$

3.4 Update $\mathbf{Y_U}$ with Θ, $\mathbf{W}$, and $\mathbf{b}$ Fixed

Note that the above problem is independent between different $\mathbf{y}_i$ $(l+1\leq i\leq l+u)$, so we can solve the following problem individually for each $\mathbf{y}_i\in\mathbf{Y_U}$ with fixed Θ, $\mathbf{W}$ and $\mathbf{b}$ by solving the following problem

$$\min_{\mathbf{y}_i\geq 0,\mathbf{y}_i^T\mathbf{1}=1}\left\|\mathbf{W}^T\mathbf{x}_i+\mathbf{b}-\mathbf{y}_i\right\|_2^2. \tag{39}$$

The Lagrangian function of the above problem is

$$\mathcal{L}(\mathbf{y_i})=\left\|\mathbf{W}^T\mathbf{x}_i+\mathbf{b}-\mathbf{y}_i\right\|_2^2+\eta(\mathbf{y}_i^T\mathbf{1}-1)-\mathbf{y}_i^T\beta, \tag{40}$$

where η and β are the Lagrangian multipliers.

Suppose the optimal solution to the proximal problem (40) is $\mathbf{y}_i^*$, the associate Lagrangian coefficients are η^* and β^*. According to the KKT condition [29], we have the following equations for every $l\in[1,c]$:

$$\begin{cases}-2(\mathbf{w}_l^T\mathbf{x}_i+\mathbf{b}_l-\mathbf{y}_{il}^*)+\eta^*-\beta_l^*=0 & (41)\\ \mathbf{y}_{il}^*\geq 0 & (42)\\ \beta_l^*\geq 0 & (43)\\ \beta_l^*\mathbf{y}_{il}^*=0 & (44)\end{cases}$$

According to Eq. (41), we have $\mathbf{y}_{il}^*=\mathbf{w}_l^T\mathbf{x}_i+\mathbf{b}_l+\frac{\beta_l^*-\eta^*}{2}$. Then we have $\eta^*=\frac{1}{c}(2\mathbf{1}^T\mathbf{W}^T\mathbf{x}_i+2\mathbf{1}^T\mathbf{b}+\mathbf{1}^T\beta^*-2)$ according to the constraint $\mathbf{y}_i^T\mathbf{1}=1$. So we have

$$\mathbf{y}_i^*=\left[\mathbf{W}^T\mathbf{x}_i-\frac{1}{c}\mathbf{1}\mathbf{1}^T\mathbf{W}^T\mathbf{x}_i+\mathbf{b}-\frac{1}{c}\mathbf{1}\mathbf{1}^T\mathbf{b}+\frac{1}{c}\mathbf{1}-\frac{\mathbf{1}\mathbf{1}^T\beta^*}{2c}\right]+\frac{\beta^*}{2}. \tag{45}$$

Denote $\bar{\beta}^*=\frac{\mathbf{1}\mathbf{1}^T\beta^*}{c}$ and $\mathbf{u}=\mathbf{W}^T\mathbf{x}_i-\frac{1}{c}\mathbf{1}\mathbf{1}^T\mathbf{W}^T\mathbf{x}_i+\mathbf{b}-\frac{1}{c}\mathbf{1}\mathbf{1}^T\mathbf{b}+\frac{1}{c}\mathbf{1}$, then we can rewrite Eq. (45) as

$$\mathbf{y}_i^*=\mathbf{u}+\frac{1}{2}(\beta^*-\bar{\beta}^*)_+. \tag{46}$$

According to Eq. (44), we know that at least one of $\mathbf{y}_{il}^*$ and β_l^* should be zero. If $(\mathbf{u}_l-\frac{1}{2}\bar{\beta}^*)\geq 0$, we know that β_l^* should be 0 such that Eq. (46) can be satisfied. If $(\mathbf{u}_l-\frac{1}{2}\bar{\beta}^*)<0$, we know that $\mathbf{y}_{il}^*$ should be 0 such that Eq. (46) can be satisfied. Therefore, the optimal solution of $\mathbf{y}_{il}^*$ should be

$$\mathbf{y}_{il}^*=(\mathbf{u}_l-\frac{1}{2}\bar{\beta}^*)_+, \tag{47}$$

where $a_+ = \max(a, 0)$. So we can obtain the optimal solution $\mathbf{y}_i^*$ if we know $\bar{\beta}^*$. Since $\mathbf{1}^T\mathbf{y}_i^* = 1$, we can obtain it by solve the following root finding problem

$$f(\bar{\beta}^*) = \sum_{l=1}^{c}\left(\mathbf{u}_l - \frac{1}{2}\bar{\beta}^*\right)_+ - 1 = 0. \tag{48}$$

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

$$(\bar{\beta}^*)_{t+1} = (\bar{\beta}^*)_t - \frac{f(\bar{\beta}^*)}{f'(\bar{\beta}^*)}, \tag{49}$$

where $f'(\bar{\beta}^*) = -b$ and b is the number of positive reals of $(\mathbf{u}_l - \frac{1}{2}\bar{\beta}^*)$ $(l \in [1, c])$.

3.5 The Optimization Algorithm

We summarize the detailed algorithm in Algorithm 1, which is denoted as Sparse Rescaled Linear Square Regression (SRLSR). In this algorithm, $\mathbf{W}$, Θ, $\mathbf{b}$ and $\mathbf{Y}_U$ are alternately updated until convergence. Finally, θ are computed from the learned $\mathbf{W}$ and the k most important features are selected according to θ. Since we obtain the optimal solution of $\mathbf{W}$, Θ, $\mathbf{b}$ and $\mathbf{Y}_U$ in each iteration, Algorithm 1 will monotonously decrease the objective function of problem (23) until the algorithm converges. Therefore, we can verify the following theorem.

Algorithm 1. Sparse Rescaled Linear Square Regression (SRLSR): Algorithm to Solve Problem (23)

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 $p \in (0, 1]$ and regularization parameter γ.
2: **Output:** k selected features.
3: Initialize $\Theta_0 \in \mathbb{R}^{d\times d}$ as an identity matrix.
4: $t := 0$.
5: **repeat**
6: Update $\mathbf{W}_{t+1} = (\mathbf{X}\mathbf{H}\mathbf{X}^T + \gamma\Theta_t^{-2})^{-1}\mathbf{X}\mathbf{H}\mathbf{Y}$.
7: Update $\mathbf{b} = \frac{1}{n}(\mathbf{Y}^T\mathbf{1} - \mathbf{W}^T\mathbf{X}\mathbf{1})$.
8: Update $\mathbf{Y}_U$, each $\mathbf{y}_i \in \mathbf{Y}_U$ is calculate from Eq. (47) individually.
9: Update Θ_{t+1} in which the jth diagonal element Θ_{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: $t := t + 1$.
11: **until** Problem (23) converges
12: Compute $\theta = \frac{\|\mathbf{w}^j\|_2^p}{\sum_{h=1}^{d}\|\mathbf{w}^h\|_2^p}$.
13: Sort θ in descending order and select top k ranked features as the ultimate result.

Theorem 1. *Algorithm 1 converges to the local optima of problem (23).*

Now we analyze the computational complexity of SRLSR. 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 number of classes. In step 6, we need $O(nd^2)$ time to compute $\mathbf{X}\mathbf{H}\mathbf{X}^T$ since $\mathbf{X}\mathbf{H}\mathbf{X}^T = \mathbf{X}\mathbf{X}^T - \frac{1}{n}(\mathbf{X}\mathbf{1})\ (\mathbf{X}\mathbf{1})^T$, and $O(nd^{2.3727} + dnc)$ time to compute $(\mathbf{X}\mathbf{H}\mathbf{X}^T + \gamma\Theta_t^{-2})^{-1}\mathbf{X}\mathbf{H}\mathbf{Y}$ if we use the Coppersmith-Winograd algorithm to compute the inverse. Thus, the total complexity in step 6 is $O(nd^{2.3727} + dnc)$. 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 $O(cd)$ time and solve β_l^* in $O(c)$ times. Therefore, we need $O(ncd)$ time to obtain $\mathbf{Y}_U$. Finally, we need $O(dn)$ time to update Θ in step 9. Finally, the computational complexity of SRLSR is $O(nd^{2.3727} + dnc)$.

Substituting the optimal solution of θ_j defined in Eq. (29) into Eq. (23), we get the following equivalent problem

$$\begin{aligned}&\min\left(\left\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{Y}\right\|_F^2 + \gamma\|\mathbf{W}\|_{2,p}^2\right)\\ &s.t.\ \mathbf{W}, \theta > 0, \mathbf{1}^T\theta = 1, \mathbf{Y}_U \geq 0, \mathbf{Y}_U\mathbf{1} = \mathbf{1},\end{aligned} \tag{50}$$

where $p = \frac{2}{q+1}$. Note that $q \geq 1$, we have $0 < p \leq 1$. In such case, minimizing $\|\mathbf{W}\|_{2,p}^2$ makes $\mathbf{W}$ sparse in rows. Therefore, problem (23) uses sparse regularizer $\|\mathbf{W}\|_{2,p}^2$ implicitly. If $p = 1$, the commonly-used $\ell_{2,1}$ norm is implicitly used as a sparse regularizer. The smaller p is, the sparser the learned regression coefficients $\mathbf{W}$ are.

Moreover, if $p = 1$, it can be verified that problem (50) is equivalent to the objective function of RLSR in [26]. In such case, problem (23) is a convex problem and Algorithm 1 will converge to its global optima. If $0 < p < 1$, problem (23) is not a convex problem and Algorithm 1 will converge to its local optima.

4 Experimental Results and Analysis

In this section, we show the performance of the proposed semi-supervised feature selection method SRLSR on real-world data sets.

4.1 Benchmark Data Sets

The following 8 benchmark data sets downloaded from Xiaojun Chen's page[1] were used in our experiment:

- *BinAlpha data set* contains 1404 images represented with 320 features. These images are from 36 classes, including 10 digits (0-9) and 26 characters (a-z).
- *CNAE-9 data set* includes 1080 documents represented with 856 word frequency features, and these documents are categorized into 9 categories.
- *Colon data set* contains 40 tumor biopsies and 22 normal biopsies, which are represented with 2000 genes.
- *Segment data set* consists of 2310 images which were drawn randomly sampled from a database of 7 outdoor images.
- *Isolet data set* consists of 7797 voice samples for the name of each letter of the 26 alphabets.
- *Caltech Silhouettes (CS) data set* consists of 8641 images contains silhouettes of objects belonging to 101 categories.
- *USPS data set* consists of 9298 images of ten digits. Each digit image is of size 16×16.
- *Srbct data set* consist of 63 samples in four classes as follows: 8 Burkitt Lymphoma (BL), 23 Ewing Sarcoma (EWS), 12 neuroblastoma (NB), and 20 rhabdomyosarcoma (RMS) and 2308 genes.

1. http://www.escience.cn/people/chenxiaojun/index.html

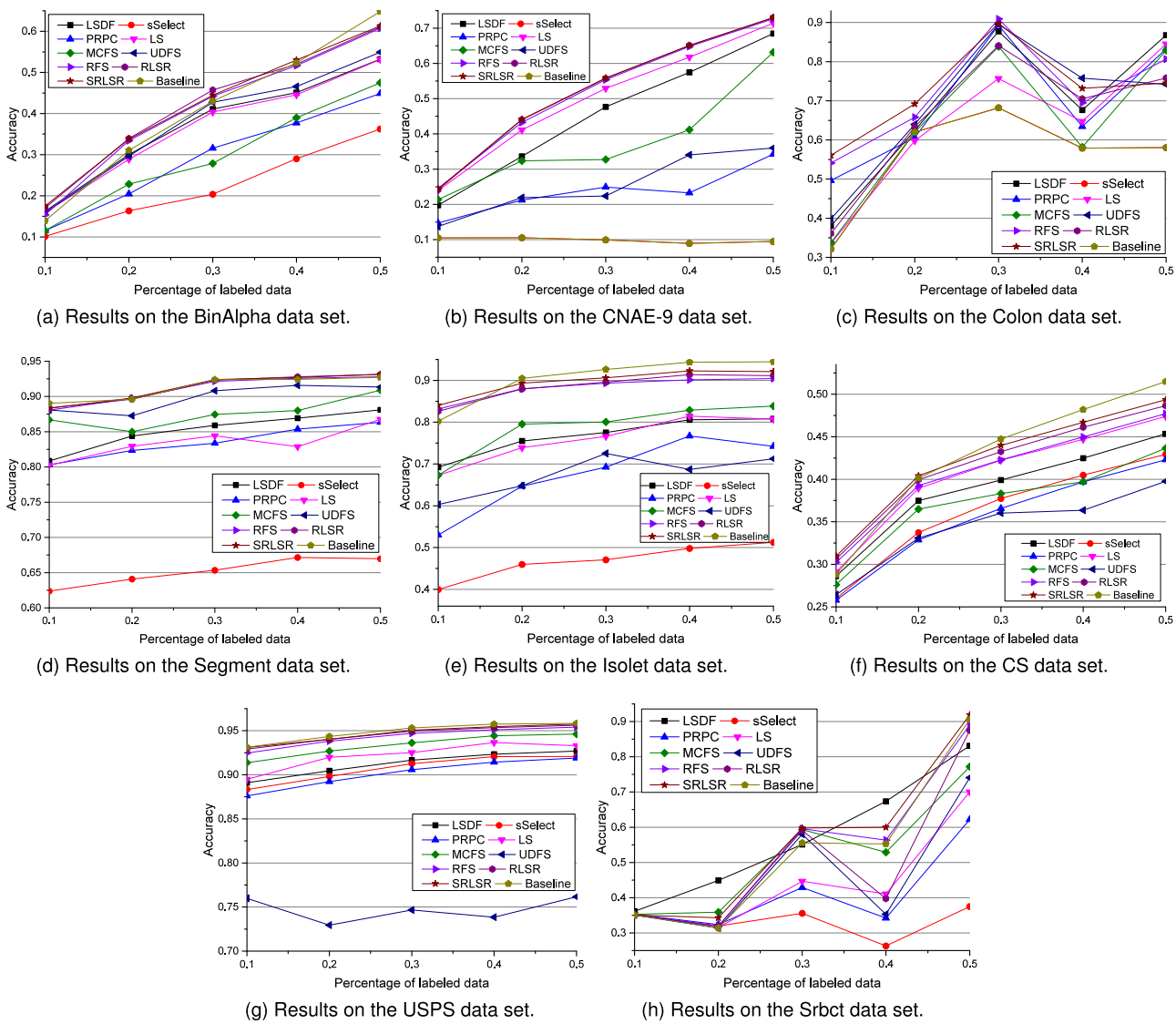
TABLE 1
Characteristics of Eight Benchmark Data Sets

Name	No. of samples	No. of features	No. of classes	No. of selected feratures
BinAlpha	1404	320	36	[20,40,...,200]
CNAE-9	1080	856	9	[20,40,...,200]
Colon	62	2000	2	[20,40,...,200]
Segment	2310	19	7	[2,3,...19]
Isolet	7797	617	26	[20,40,...,200]
CS	8641	256	101	[20,40,...,200]
USPS	9298	256	10	[20,40,...,200]
Srbct	63	2308	4	[20,40,...,200]

The characteristics of these 8 data sets are summarized in Table 1.

4.2 Comparison Scheme

To validate the effectiveness of *SRLSR*, we compared it with 8 state-of-the-art feature selection methods, including four semi-supervised feature selection methods *sSelect* [15], *LSDF* [16], *RRPC* [18] and *RLSR* [26], three unsupervised feature selection method Laplacian Score (*LS*) [11], *UDFS* [30] and *MCFS* [31], and a supervised feature selection method *RFS* [9]. Baseline is implemented by training SVM with all features. We set the regularization parameter γ of LS, LSDF, RFS, UDFS, sSelect and SRLSR as $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{2}, 10^{3}\}$, λ of sSelect as $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$. The sparsity parameter p in SRLSR was set as 10 values $\{0.1, 0.2, \ldots, 1.0\}$. For each method, we used each parameter setting to rank features, and selected a specific number of features for evaluating the performance, where the number of selected features for each data set are shown in Table 1.

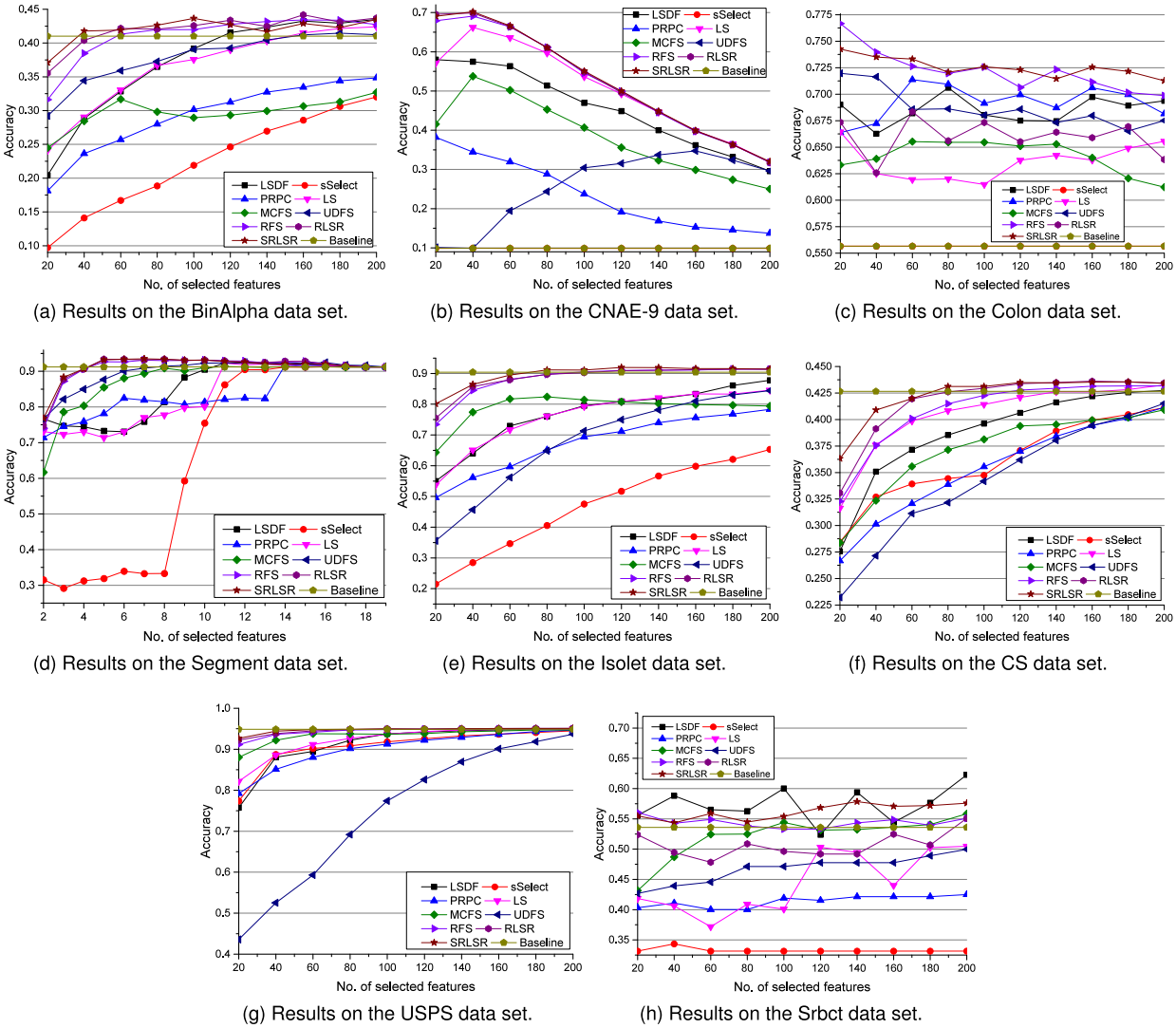


(a) Results on the BinAlpha data set. (b) Results on the CNAE-9 data set. (c) Results on the Colon data set.

(d) Results on the Segment data set. (e) Results on the Isolet data set. (f) Results on the CS data set.

(g) Results on the USPS data set. (h) Results on the Srbct data set.

Fig. 2. The average accuracy versus the percentage of labeled records by 9 feature selection methods on eight data sets.



(a) Results on the BinAlpha data set. (b) Results on the CNAE-9 data set. (c) Results on the Colon data set.

(d) Results on the Segment data set. (e) Results on the Isolet data set. (f) Results on the CS data set.

(g) Results on the USPS data set. (h) Results on the Srbct data set.

Fig. 3. The average accuracy versus the number of selected features by 9 feature selection methods on eight data sets.

For each of the 8 data sets, the training samples were randomly selected with the given sample ratio $\{10\%, 20\%, 30\%, 40\%, 50\%\}$. The remaining samples were then used as the test data. The test data were also used as the unlabeled data for the semi-supervised feature selection algorithm. For unsupervised feature selection methods, we only used the training samples without labels. For the selected features, we first performed 5-fold cross-validation on the labeled data to select the best SVM model and then we tested the selected SVM model on the unlabeled part.

4.3 Results and Analysis

Since each method produced multiple results with different parameters, we computed the average accuracies of 10 feature selection methods for comparison. The average accuracies of all methods versus the percentages of labeled records are shown in Fig. 2, and their average accuracies are summarized in Table 2. The average accuracies of all methods versus the number of selected features are reported in Fig. 3, in which only 40 percent of labels in each data set are used for training. In general, the more labeled data we used, the higher accuracy we can achieve. This indicates that we are able to select features with higher quality if more labeled data is available. Overall, our proposed method SRLSR outperformed other methods in accuracy on most data sets, especially significantly outperformed other methods on the *Isolet*, *CS* and *USPS* data sets. Specifically, SRLSR achieves a nearly 1.44 percent average improvement on the *CS* data set, compared to the second-best method RLSR. On the *Isolet* data set, SRLSR achieves a nearly 1.36 percent average improvement compared to the second-best method RLSR. SRLSR also achieved good performance for the rest data sets in average. We also notice that SRLSR outperformed RFS on most data sets. Especially on the *CS* dataset, SRLSR achieves a greater than 3.4 percent improvement compared to RFS. This indicates that the introduction of unlabeled data indeed improves the performance of feature selection.

The average time costs of 9 feature selection methods except RLSR are reported in Fig. 5, in which only 40 percent of labels in each data set are used for training. In this experiment, we set the regularization parameters in all methods to 1 in order to perform fair comparison. Note that RLSR is a special case of SRLSR with $p = 1$, the time costs of SRLSR in Fig. 5 are the same as those of RLSR. Among four semi-supervised feature selection methods, the time costs of

TABLE 2
The Average Accuracy±Standard Deviation of the Results of 9 Feature Selection Methods on Eight Benchmark Data Sets

Algorithms	BinAlpha	CNAE-9	Colon	Segment	Isolet	CS	USPS	Srbct
LSDF	0.371 ± 0.128	0.454 ± 0.172	0.685 ± 0.183	0.852 ± 0.025	0.767 ± 0.042	0.388 ± 0.057	0.912 ± 0.013	$\mathbf{0.573 \pm 0.166}$
sSelect	0.224 ± 0.092	0.099 ± 0.006	0.557 ± 0.123	0.652 ± 0.018	0.468 ± 0.039	0.362 ± 0.059	0.907 ± 0.014	0.333 ± 0.039
PRPC	0.292 ± 0.119	0.237 ± 0.063	0.692 ± 0.148	0.835 ± 0.021	0.676 ± 0.084	0.354 ± 0.058	0.901 ± 0.016	0.414 ± 0.110
LS	0.366 ± 0.129	0.502 ± 0.165	0.637 ± 0.173	0.834 ± 0.021	0.76 ± 0.052	0.405 ± 0.063	0.922 ± 0.015	0.445 ± 0.135
MCFS	0.297 ± 0.125	0.382 ± 0.14	0.641 ± 0.185	0.876 ± 0.019	0.787 ± 0.059	0.372 ± 0.053	0.933 ± 0.012	0.521 ± 0.157
UDFS	0.380 ± 0.137	0.256 ± 0.083	0.687 ± 0.166	0.898 ± 0.018	0.675 ± 0.045	0.343 ± 0.045	0.747 ± 0.012	0.468 ± 0.166
RFS	$\mathbf{0.411 \pm 0.157}$	0.521 ± 0.169	$\mathbf{0.722 \pm 0.126}$	0.911 ± 0.019	0.882 ± 0.027	0.409 ± 0.06	0.943 ± 0.011	0.544 ± 0.205
RLSR	$\mathbf{0.420 \pm 0.153}$	$\mathbf{0.524 \pm 0.171}$	0.660 ± 0.164	$\mathbf{0.912 \pm 0.019}$	$\mathbf{0.885 \pm 0.032}$	$\mathbf{0.417 \pm 0.063}$	$\mathbf{0.946 \pm 0.010}$	0.507 ± 0.207
SRLSR	$\mathbf{0.420 \pm 0.153}$	$\mathbf{0.525 \pm 0.170}$	$\mathbf{0.725 \pm 0.108}$	$\mathbf{0.913 \pm 0.019}$	$\mathbf{0.897^{*} \pm 0.030}$	$\mathbf{0.423^{*} \pm 0.064}$	$\mathbf{0.947^{*} \pm 0.010}$	$\mathbf{0.562 \pm 0.211}$

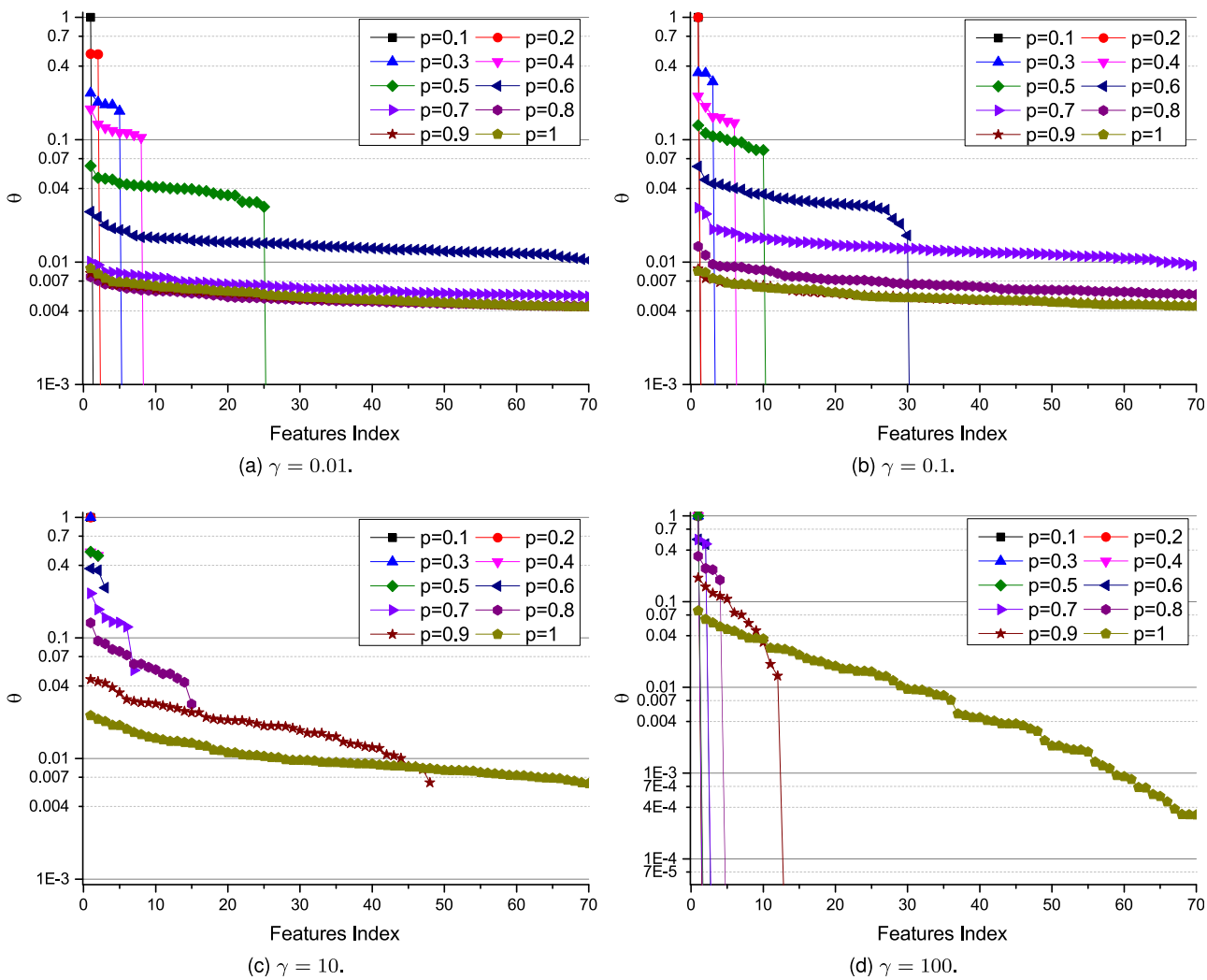
Here, the best two results on each data set are highlighted in bold and the "" also indicates the difference between the results of the corresponding algorithm (excluding SRLSR) and those of SRLSR is significant by t-test, i.e., the p-value of t-test is less than 0.05.*

SRLSR were much less than the time costs of sSelect and PRPC, which indicates the efficiency of the new method.

4.4 Parameter Sensitivity Study

In SRLSR, both γ and p affect the row sparsity of the projection matrix $\mathbf{W}$. From Eq. (50), we can observe that minimizing problem (50) with larger γ or smaller p prefers to make the learnt rows of $\mathbf{W}$ sparser. In this experiment, we investigate the two parameters γ and p in SRLSR.

We first study the effect of γ and p on the performance of SRLSR. The relationship between θ and p, γ is shown in Fig. 4, in which γ was set as 4 values as 0.01, 0.1, 10, 100, and p was set as 10 values varying from 0.1 to 1. We only show 70 largest θ values on the CS data set for brevity. From this figure, we can observe that the high weights in θ are set to 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



(a) $\gamma = 0.01$. (b) $\gamma = 0.1$.

(c) $\gamma = 10$. (d) $\gamma = 100$.

Fig. 4. θ (70 largest values) versus p and γ in SRLSR on the CS data set.

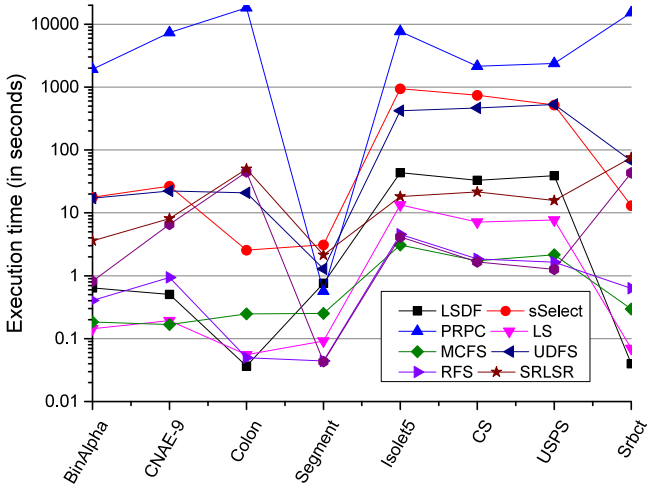


Fig. 5. The average time costs (s) of 8 feature selection methods on eight benchmark data sets.

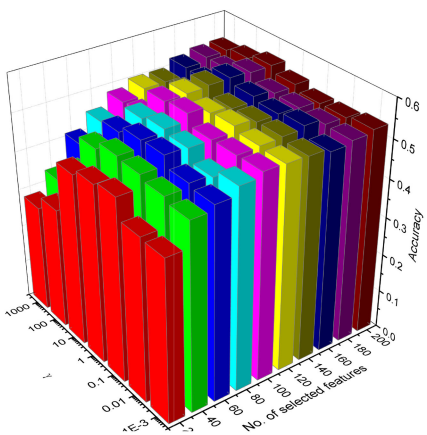
weights, thus we can select p and γ according to the distribution of θ.

In SRLSR, γ and p are used to control the row sparsity of $\mathbf{W}$, and their values seriously influence the final performance. Varying the values of γ and the number of selected features, the average classified accuracies on the BinAlpha and CNAE-9 data set are shown in Fig. 6. As the number of selected features increased, the accuracies of SRLSR decreased on the CNAE-9 data set but increased on the BinAlpha data set. On the BinAlpha data set, when we selected a small number of features, SRLSR produced better results with small γ in order to preserve enough features. As we selected more features, SRLSR produced better results with high γ which forces only a few of important features to be selected. On the CNAE-9 data set, the best results were produced by SRLSR with the largest γ on most experiments.

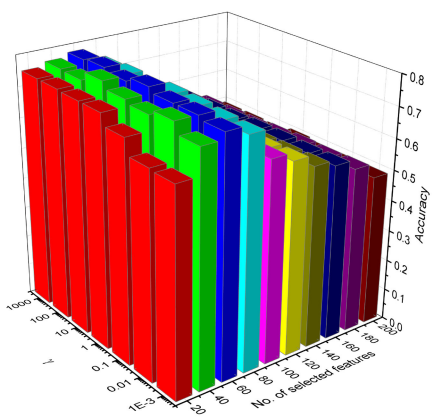
We also show the the average clustering accuracies versus p and the number of selected features on the BinAlpha and CNAE-9 data sets in Fig. 7. On the BinAlpha data set, SRLSR produced better results with high p when only a small number of features were selected, but with low p when a large number of features were selected. On the CNAE-9 data set, the best results were produced by SRLSR with the smallest p in most experiments. In summary, SRLSR produced good results with large γ and small p, which demonstrates the significance of learning sparse projection weights. In real life applications, we can perform hierarchy grid search to select the proper γ and p for better result.

4.5 Convergence Study

In this section, we experimentally study the convergence speed of SRLSR. For simplicity, we only show results on

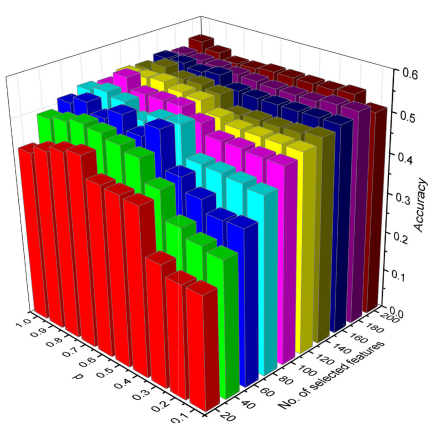


(a) Results of the BinAlpha data set.

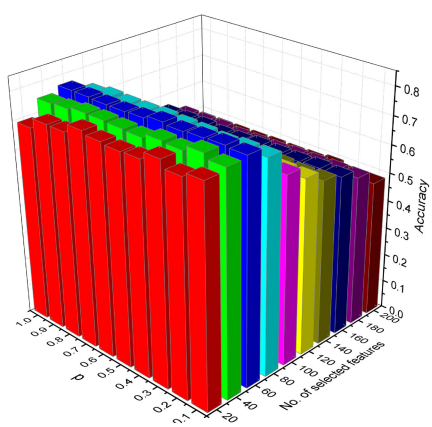


(b) Results on the CNAE-9 data set.

Fig. 6. Accuracy versus γ and the number of selected features.

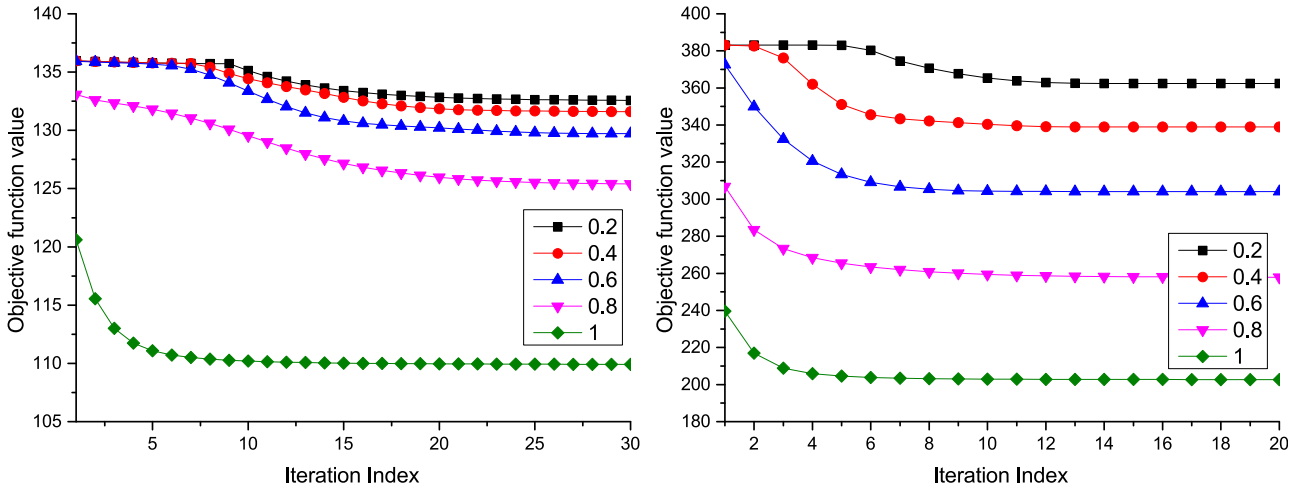


(a) Results of the BinAlpha data set.



(b) Results on the CNAE-9 data set.

Fig. 7. Accuracy versus p and the number of selected features.



(a) Results on the BinAlpha dataset. (b) Results on the CNAE-9 dataset.

Fig. 8. The objective function values of SRLSR versus iterations.

two datasets (i.e., BinAlpha and CNAE-9). The variation curves of the objective value are shown in Fig. 8 which shows that SRLSR converges rapidly.

5 Conclusions

In this paper, we propose a novel embedded semi-supervised feature selection approach named SRLSR, in which a new Sparse Rescaled Linear Regression model is proposed to explicitly evaluate the importance of features. The new method is proved to be equivalent to a sparse model, in which the sparse $\ell_{2,p}$ norm regularization is explicitly used. Therefore, we can produce more sparser projection matrix by setting smaller p. The optimal solution of scale factors also provides a theoretical explanation for why we can use $\{\|\mathbf{w}^1\|_2, \ldots, \|\mathbf{w}^d\|_2\}$ to evaluate the importance of features. Empirical studies have been performed on 8 real-world datasets and the experimental results show the superior performance of our method.

In future work, we will improve this method for large-scale data.

Acknowledgments

This research was supported by NSFC under Grant no. 61773268, U1636202, 61836005 and Tencent Rhinoceros Birds - Scientific Research Foundation for Young Teachers of Shenzhen University.

References

[1] D. L. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality," in *Proc. AMS Conf. Math Challenges 21st Century*, 2000, pp. 178–183.
[2] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005.
[3] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinf.*, vol. 23, no. 19, pp. 2507–2517, 2007.
[4] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 16–28, 2014.
[5] S. H. Huang, "Supervised feature selection: A tutorial," *Artif. Intell. Res.*, vol. 4, no. 2, 2015, Art. no. 22.
[6] D. Richard, P. E. Hart, and D. G. Stork, *Pattern Classification*. Hoboken, NJ, USA: Wiley, 2010.
[7] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proc. 9th Int. Workshop Mach. Learn.*, 1992, pp. 249–256.
[8] I. Kononenko, "Estimating attributes: Analysis and extensions of relief," in *Proc. Eur. Conf. Mach. Learn.*, 1994, pp. 171–182.
[9] F. Nie, H. Huang, X. Cai, and C. H. Ding, "Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization," in *Proc. Advances Neural Inf. Process. Syst.*, 2010, pp. 1813–1821.
[10] 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.
[11] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Advances Neural Inf. Process. Syst.*, 2005, pp. 507–514.
[12] L. Shi, L. Du, and Y. D. Shen, "Robust spectral learning for unsupervised feature selection," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2014, pp. 977–982.
[13] 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.
[14] 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. [Online]. Available: http://dl.acm.org/citation.cfm?id=2891460.2891550
[15] Z. Zhao and H. Liu, "Semi-supervised feature selection via spectral analysis," in *Proc. SIAM Int. Conf. Data Mining*, 2007, pp. 641–646. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9781611972771.75
[16] J. Zhao, K. Lu, and X. He, "Locality sensitive semi-supervised feature selection," *Neurocomputing*, vol. 71, pp. 1842–1849, 2008.
[17] G. Doquire and M. Verleysen, "A graph laplacian based approach to semi-supervised feature selection for regression problems," *Neurocomputing*, vol. 121, pp. 5–13, 2013.
[18] 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.
[19] I. Guyon and A. Elisseef, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.
[20] J. Ren, Z. Qiu, W. Fan, H. Cheng, and P. S. Yu, "Forward semi-supervised feature selection," in *Proc. 12th Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2008, pp. 970–976.
[21] 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.
[22] T. Strutz, *Data Fitting and Uncertainty: A Practical Introduction to Weighted Least Squares and Beyond*. Berlin, Germany: Vieweg and Teubner, 2010.
[23] S. Wold, A. Ruhe, H. Wold, and W. Dunn, III,, "The collinearity problem in linear regression. the partial least squares (pls) approach to generalized inverses," *SIAM J. Sci. Statistical Comput.*, vol. 5, no. 3, pp. 735–743, 1984.

[24] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. New York, NY, USA: Cambridge Univ. Press, 2000.
[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] 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.*, 2017, pp. 1525–1531.
[27] X. Kong and P. S. Yu, "Semi-supervised feature selection for graph classification," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 793–802.
[28] 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.
[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[30] 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.
[31] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 333–342.

Xiaojun Chen received the PhD degree from the Harbin Institute of Technology, in 2011. He is now an assistant professor with the College of Computer Science and Software, Shenzhen University. His research interests include subspace clustering, topic model, feature selection, and massive data mining. He is a member of the IEEE.

Guowen Yuan is now working toward the master's (MA) degree at the College of Computer Science and Software, Shenzhen University. His research interests include clustering and feature selection.

Feiping Nie received the PhD degree in computer science from Tsinghua University, China, in 2009. His research interests are machine learning and its applications, such as pattern recognition, data mining, computer vision, image processing, and information retrieval. He has published more than 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/IEEE Transactions on Neural Networks*, the *IEEE Transactions on Knowledge and Data Engineering*, the *Transactions on Knowledge Discovery from Data*, *Bioinformatics*, ICML, NIPS, KDD, IJCAI, AAAI, ICCV, CVPR, and ACM MM. His papers have been cited more than 5000 times (Google scholar). He is now serving as an associate editor or PC member for several prestigious journals and conferences in the related fields.

Zhong Ming is a professor with the College of Computer and Software Engineering, Shenzhen University. He is a member of the council and senior member of the Chinese Computer Federation. His major research interests are in software engineering and embedded systems. He led two projects of the National Natural Science Foundation, and two projects of the Natural Science Foundation of Guangdong province, China.