Feature Selection Based on Structured Sparsity: A Comprehensive Study

Jie Gui, *Member, IEEE*, Zhenan Sun, *Member, IEEE*, Shuiwang Ji, *Senior Member, IEEE*, Dacheng Tao, *Fellow, IEEE*, and Tieniu Tan, *Fellow, IEEE*

***Abstract*— Feature selection (FS) is an important component of many pattern recognition tasks. In these tasks, one is often confronted with very high-dimensional data. FS algorithms are designed to identify the relevant feature subset from the original features, which can facilitate subsequent analysis, such as clustering and classification. Structured sparsity-inducing feature selection (SSFS) methods have been widely studied in the last few years, and a number of algorithms have been proposed. However, there is no comprehensive study concerning the connections between different SSFS methods, and how they have evolved. In this paper, we attempt to provide a survey on various SSFS methods, including their motivations and mathematical representations. We then explore the relationship among different formulations and propose a taxonomy to elucidate their evolution. We group the existing SSFS methods into two categories, i.e., vector-based feature selection (feature selection based on lasso) and matrix-based feature selection (feature selection based on $l_{r,p}$-norm). Furthermore, FS has been combined with other machine learning algorithms for specific applications, such as multitask learning, multilabel learning, multiview learning, classification, and clustering. This paper not only compares the differences and commonalities of these methods based on regression and regularization strategies, but also provides useful guidelines to practitioners working in related fields to guide them how to do feature selection.**

***Index Terms*— Dimensionality reduction, feature selection, sparse, structured sparsity.**

Manuscript received August 11, 2015; revised February 1, 2016; accepted April 2, 2016. Date of publication April 22, 2016; date of current version June 15, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61572463, in part by the "Thirteenth Five-Year" National Key Research and Development Program of China under Grant 2016YFD0702002, in part by the grant of the Open Project Program of the National Laboratory of Pattern Recognition (NLPR) under Grant 201700027, in part by the grant of the Open Project Program of the State Key Lab of CAD&CG under Grant A1709, Zhejiang University, in part by the grant of the Shanghai Key Laboratory of Intelligent Information Processing, China under Grant IIPL-2016-003, in part by the Australian Research Council under Project DP-140102164, Project FT-130101457, and Project LE140100061, and in part by the U.S. National Science Foundation under Grant DBI-1147134 and Grant DBI-1350258. *(Corresponding author: Zhenan Sun.)*

J. Gui is with the Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031, China (e-mail: guijie@ustc.edu).

Z. Sun and T. Tan are with the National Laboratory of Pattern Recognition, Center for Research on Intelligent Perception and Computing, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: znsun@cripac.ia.ac.cn; tnt@cripac.ia.ac.cn).

S. Ji is with the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752 USA (e-mail: sji@eecs.wsu.edu).

D. Tao is with the Centre for Quantum Computation & Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: dacheng.tao@uts.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2551724

I. Introduction

DATA are commonly represented by high-dimensional feature vectors in many areas, such as computer vision, pattern recognition, machine learning, and data mining. High dimensionality significantly increases the time and space requirements for processing the data. Moreover, various data mining and machine learning tasks, such as classification, clustering, and regression, that are analytically or computationally manageable in low-dimensional spaces may become difficult in spaces of several hundreds or thousands of dimensions. To solve this issue, feature selection (also known as feature ranking, subset, or variable selection) [1]–[7] techniques are designed to select a subset of features from the high-dimensional feature set for a compact and accurate data representation. Once a small number of relevant features are selected, conventional data analysis techniques can then be applied.

In contrast to other dimensionality reduction techniques such as those based on projection [e.g., principal component analysis (PCA)] [8]–[11] or compression (e.g., using information theory), feature selection techniques do not alter the original representation of the variables, but merely select a subset of them. Thus, they preserve the original semantics of the variables, thereby offering the advantage of interpretability by a domain expert.

As many pattern recognition techniques were originally not designed to cope with high-dimensional data, combining them with FS techniques has become a necessity in many applications. FS has manifold role in improving the performance for data analysis. First, the dimension of a selected feature subset is much lower, thereby reducing the measurement and storage requirements and making the subsequent computation on the input data more efficient. Second, FS reduces training time and provides faster and more cost-effective models. Third, it helps to gain a deeper insight into the underlying processes that generated the data and facilitating data visualization and data understanding. Fourth, it can avoid overfitting and improve model performance, i.e., prediction performance in the case of supervised classification and better cluster detection in the case of clustering. Fifth, the noisy features are eliminated for a better data representation, resulting in a more accurate clustering and classification result.

Feature selection algorithms can be roughly classified into two groups, i.e., filter methods and wrapper methods. The filter methods rely on general characteristics of the data to evaluate and select feature subsets without involving any learning algorithm. Variance and Fisher score [12], [13] might be two

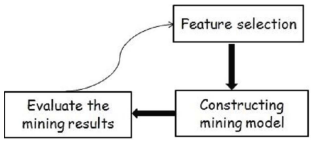


Fig. 1. Differences between filter methods and wrapper methods.

of the simplest filter methods. The wrapper model requires one predetermined learning algorithm and uses its performance as the evaluation criterion. It searches for features better suited to the learning algorithm aiming to improve performance, but it also tends to be more computationally expensive than the filter model. The differences between the filter methods and the wrapper methods are shown in Fig. 1. Wrapper methods have the curve, while filter methods do not have the curve.

Recently, some new methods integrating the theory of sparse representation [14], [15], compressed sensing, and feature selection methods [16]–[21] have been proposed. For example, sparse PCA was used as the feature selection method in [22] and [23]. Sparsity-inducing feature selection algorithms have been successfully applied in face detection [24], face authentication [25], gene expression, and mass spectrometry data classification [26]. These algorithms resort to sparsity-inducing regularization techniques, such as an l_1-norm constraint or penalty term. These methods have been studied in depth by the machine learning community as they raise interesting theoretical issues and carry useful properties. Sparsity-inducing algorithms have the following advantages.

1) They are robust to data noise. The data noise is inevitable, especially for visual data, and the robustness is a desirable property for a satisfying method. Sparsity-inducing algorithms have been shown to be robust to data noise in [27].
2) These approaches are supported by well-grounded theory and appropriate for mathematical analysis.

Analyzing the merits and limitations can be very useful. Hopefully, the gained experience will stimulate further theoretical and numerical progress in the community. Recently, structured sparsity-induced [28]–[30] feature selection algorithms considering the structure of features have shown promising results in many practical applications and have received more and more attention. However, there is still no work that comprehensively studies this exciting field. This paper attempts to make such a timely survey, in which various structured sparsity-inducing feature selection (SSFS) methods are introduced, their relationships are exploited, and open problems and future directions are discussed. We believe that this paper will greatly benefit both beginners and practitioners in this field.

For clarity, the existing SSFS algorithms can be grouped into two categories shown in Fig. 2: vector-based feature selection (feature selection based on lasso [31]) and matrix-based feature selection (feature selection based on $l_{r,p}$-norm). Furthermore, feature selection has been combined with other machine learning algorithms for specific applications, such as multitask learning, multilabel learning, multiview learning, classification, and clustering.

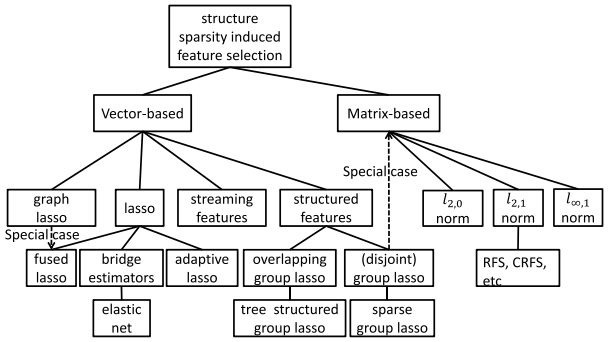


Fig. 2. Taxonomy of structure sparsity-induced feature selection.

A. Difference From Previous Work

The importance and the popularity of sparsity and feature selection methods have led to several previous surveys. Each such survey is discussed in the following to put the current review in context.

1) *Review of Sparsity:* The earliest relevant review was probably due to Wright *et al.* [32], which presented an overview of sparse representation for computer vision and pattern recognition, such as classification, graph construction, and dictionary learning. Cheng *et al.* [14] presented a survey of some recent work on sparse representation, learning, and modeling with an emphasis on visual recognition. Bach *et al.* [33] gave a monograph to present the tools and techniques related to sparsity-inducing penalties from a general optimization perspective. Elad [15] wrote a book on sparse and redundant representations, which covered both theory and applications in signal and image processing. Some other related works can be found in [34]–[36].
2) *Review of Feature Selection [1]–[7]:* Previous reviews mainly focus on either sparsity or feature selection, and SSFS is almost untouched in theses reviews. To the best of our knowledge, this paper is the first one to provide a comprehensive review on SSFS, and the major contributions are summarized in Section I-B.

B. Contributions

The major contributions of this paper are summarized as follows.

1) We provide a survey on recent progress in SSFS, including the motivations and mathematical representations of different algorithms. This part is especially useful for the beginners to be familiar with this area.
2) We exploit the relationships among different kinds of SSFS algorithms, based on which we propose a taxonomy and show how they have evolved. This part provides an in-depth understanding to this research field.
3) We evaluate several representative SSFS methods. Some meaningful findings are obtained, which is beneficial for practical applications.
4) We summarize main challenges and problems of current studies, and point out some future research directions.

C. Notations of This Paper

We summarize the notations and the definitions used in this paper. Let $X = (X_1, \ldots, X_n)$ be n data points in the d-dimensional space. For matrix form $X = (X_{ji})$, its jth ($j = 1, \ldots, d$) row and ith ($i = 1, \ldots, n$) column are denoted by X^j, X_i, respectively. For binary classification, denote label matrix $y = (y_1, \ldots, y_n)^T$, which is the $n \times 1$ vector containing output label $y_i \in \{+1, -1\}$, $i = 1, \ldots, n$. For multiclass classification, denote label matrix Y as an $n \times c$ matrix, where c is the number of classes. Denote α as the regularization parameter regulating the balance between the data misfit and the penalty. $e_n = [1 \cdots 1]^T \in R^n$ is a vector with all its entries equal to 1. Denote W as the projection matrix. For different algorithms, the dimension of the projection matrix may be different, which will be stated in Section III.

The remainder of this paper is organized as follows. Section II describes vector-based feature selection (feature selection based on lasso). Section III groups and compares different matrix-based feature selection algorithms, which are based on $l_{r,p}$-norm. Section IV introduces task-driven feature selection. The experimental results are shown in Section V, and finally, Section VI concludes the survey with discussions on current trends and future directions.

II. Vector-Based Feature Selection

A. l_1-Norm Regularized/Constrained Feature Selection (Feature Selection Based on Lasso)

An interesting way to cope with feature selection in the learning by examples framework is to resort to regularization techniques based on l_1 penalty [37], [38]. In the case of linear problems, a theoretical support for this strategy can be derived from [39], where it is shown that, for most underdetermined linear systems, the minimal l_1 solution equals the sparsest solution. Destrero *et al.* [24], [25], [40] used the Lagrangian formulation of lasso for feature selection in face detection and face authentication. Its objective function is formulated as follows:

$$\min_{w} \ \|y - X^T w\|_2^2 + \gamma \|w\|_1 \tag{1}$$

where $w = (w_1, w_2, \ldots, w_d)$ is the vector of the unknown weights to be estimated.

Lasso produces biased estimates for the large coefficients, and thus, it is suboptimal. To improve the performance of lasso, the adaptive lasso [41] is proposed

$$\text{penalty}(w) = \sum_{i=1}^{d} a_i |w_i| \tag{2}$$

where the only difference between lasso and adaptive lasso is that the latter uses a weight a_i for each coefficient w_i. This paper shows that the adaptive lasso performs as well as if the true model was given in advance. Furthermore, it can be solved by the same efficient algorithm for solving the lasso.

The fused lasso penalty introduced in [42] yields a solution that has sparsity in both the coefficients and their successive differences, i.e., local constancy of the coefficient profile

$$\text{penalty}(w) = \|w\|_1 + \alpha \sum_{i=2}^{d} |w_i - w_{i-1}|. \tag{3}$$

The fused lasso has been successfully used in many practical applications [43], where features can be ordered in some meaningful way. The pairwise fused lasso [44] is a further extension of fused lasso, but it does not assume that the predictors have to be ordered.

The bridge estimator [45] is defined as follows:

$$\text{penalty}(w) = \sum_{i=1}^{d} |w_i|^\gamma. \tag{4}$$

The bridge estimator has two important special cases. When $\gamma = 2$, it is the popular ridge estimator. When $\gamma = 1$, it is the lasso.

It is common that some features are strongly correlated in practical applications. In this case, the lasso tends to select only one of the correlated features. To handle features with strong correlations, elastic net regularization [46] is proposed as

$$\text{penalty}(w) = \alpha \sum_{i=1}^{d} |w_i| + (1 - \alpha) \sum_{i=1}^{d} w_i^2. \tag{5}$$

The elastic net is a mixture of ridge and lasso estimator. Liu *et al.* [47] extended the elastic net as follows:

$$\text{penalty}(w) = \alpha \sum_{i=1}^{d} |w_i|^\gamma + (1 - \alpha) \sum_{i=1}^{d} w_i^2. \tag{6}$$

Based on [24], [25], and [40], Wang *et al.* [48] and Sun *et al.* [49] proposed a robust feature selection (RFS) method via linear programming. There are other variants of lasso, such as trace lasso [50].

B. Structured Features

The penalties introduced in Section II-A assume that the features are independent and ignored the structures of features completely [51]. However, in practical applications, the features have some essential structures, such as disjoint groups [52], [53], overlapping groups [54], graphs [55], and trees [56]. Integrating knowledge about the feature structures may help identify the important features.

1) (Disjoint) Group Lasso: In many practical applications, features form group structures and sparsity alone may not be sufficient to obtain the desired solution. The prior knowledge of the group structure of features can be supplementary to further improve the performance of sparse regularization. For this purpose, group lasso [52], [53] has been proposed. Suppose that features are divided into k disjoint groups and any two groups are nonoverlapping. With the group structure, w is rewritten as k disjoint groups $w = \{w_{G_1}, w_{G_2}, \ldots, w_{G_k}\}$. Group lasso is defined as

$$\text{penalty}(w) = \sum_{i=1}^{k} \beta_i \|w_{G_i}\|_q \tag{7}$$

where $\|\|_q$ is the l_q-norm and β_i is the weight for the ith group. Lasso does not consider group structure and cannot select or not select a group of features simultaneously, while group lasso supports group selection.

A further extension of the group lasso, namely, the sparse group lasso [57], combines both lasso and group lasso, and it produces a solution with simultaneous between- and within-group sparsity. The sparse group lasso regularization is defined as

$$\text{penalty}(w) = (1-\alpha)\|w\|_1 + \alpha \sum_{i=1}^{k} \beta_i \|w_{G_i}\|_q \tag{8}$$

where $\alpha \in [0, 1]$, the first term is for feature selection and the second term is for group selection. When $\alpha = 0$, sparse group lasso reduces to the lasso, and when $\alpha = 1$, it reduces to the group lasso. Similar to the sparse group lasso, exclusive group lasso [58] is proposed to enforce sparsity at intragroup level for feature selection and it is based on $l_{1,2}$-norm.

2) Overlapping Group Lasso: In some applications, the groups overlap and the group lasso cannot handle this case. Thus, overlapping group lasso is proposed in [59]. A general overlapping group lasso regularization is similar to that for group lasso regularization

$$\text{penalty}(w) = \sum_{i=1}^{k} \beta_i \|w_{G_i}\|_q. \tag{9}$$

Note that the groups for overlapping group lasso are overlapping, while the groups in group lasso are disjoint.

In many applications, features can naturally form certain tree structures. For instance, the pixels of the face images can be represented as a tree, where each parent node contains a series of child nodes that enjoy spatial locality. The tree structured group lasso is a special case of the overlapping group lasso with a specific tree structure.

C. Graph Lasso

We often have prior knowledge about pairwise relationships between the features in many practical applications. For instance, many biological studies have demonstrated that genes tend to work in groups based on their biological functions, and there exists some regulatory relationships between genes. In this situation, the prior knowledge can be represented as a graph, where the nodes denote the features, and the edges represent the relationships between features.

Let $G = (V, E)$ be a given graph, where $V = \{1, \ldots, n\}$ is a set of nodes, and E is a set of edges. Node i corresponds to the ith feature and $B \in R^{n\times n}$ is used to denote the adjacency matrix of G. One intuitive way to formulate graph lasso is defined as

$$\text{penalty}(w) = (1-\alpha)\|w\|_1 + \alpha \sum_{ij} B_{ij}(w_i - w_j)^2 \tag{10}$$

where the second term is similar to Laplacian eigenmaps [60], [61]. Thus, this method is named LapLasso.

To perform feature selection with a signed feature graph, GFlasso [62] extends LapLasso as

$$\text{penalty}(w) = (1-\alpha)\|w\|_1 + \alpha \sum_{ij} B_{ij}(w_i - \text{sign}(r_{ij})w_j)^2 \tag{11}$$

where r_{ij} is the correlation between two features. When the ith feature and the jth feature are positively correlated $r_{ij} > 0$, while the ith feature and the jth feature are negatively correlated $r_{ij} < 0$. Some approaches have been proposed to follow and extend GFlasso [63], [64]. Please note that fused lasso can be viewed as a special case of graph lasso. In the fused lasso, only x_i and $x_{i+1} (i = 1, 2, \ldots, n-1)$ have nodes, while more complex structure can exist in graph lasso.

D. Streaming Features

In the standard feature selection problem, all features are given in advance, and another interesting scenario is that candidate features arrive one at a time. In this case, the features are generated dynamically, and the number of features is unknown. This kind of features is named streaming features, and feature selection for streaming features is named streaming feature selection [65]. Streaming feature selection is very useful in many practical applications. For instance, the famous Chinese microblogging Web site weibo[1] produces more than 100 million weibo per day, and many new words (features), such as abbreviations, are generated. When performing feature selection for weibo, it is not practical to wait until all features have arrived. Therefore, streaming feature selection is more preferable in this case. Traditional feature selection methods do not perform well in this case, and Perkins and Theiler [66] proposed a streaming feature selection framework based on a stagewise gradient descent technique and l_p-norm regularizer.

III. Matrix-Based Feature Selection: $l_{r,p}$-Norm Regularized/Constrained Feature Selection

Most of the methods introduced in Section II can only solve binary classification problems and cannot solve multiclass problems. To solve multiclass problems, Nie *et al.* [26] proposed a new feature selection algorithm, which uses $l_{2,1}$-norm instead of l_1-norm as the penalty.

A. $l_{r,p}$-Norm of a Matrix

In this paper, we define the $l_{r,p}$-norm of a matrix $A \in R^{u\times v}$ to be the p-norm of the vector containing of the r-norms of the matrix rows, that is

$$\|A\|_{r,p} = \|(\|A^1\|_r, \ldots, \|A^i\|_r, \ldots, \|A^u\|_r)\|_p \tag{12}$$

where A^i denotes the ith row of A. By comparing the definition of $l_{r,p}$-norm and group lasso, it can be seen that $l_{r,p}$-norm is a special case of group lasso when A^i denotes a group of group lasso and the weights for all groups of group lasso are one. As seen from the above definition of $l_{r,p}$-norm, it is the commonly used Frobenius norm (or l_2-norm) and

[1] http://weibo.com/

l_1-norm (or $l_{1,1}$-norm) when $r = p = 2$ and $r = p = 1$, respectively

$$\|A\|_F = \left(\sum_{i=1}^{u}\sum_{j=1}^{v} A_{ij}^2\right)^{\frac{1}{2}}, \quad \|A\|_{l_1} = \sum_{i=1}^{u}\sum_{j=1}^{v} |A_{ij}|. \quad (13)$$

When $r = 2,\ p = 1$, it is the $l_{2,1}$-norm of a matrix, which was first introduced in [67] as rotational invariant l_1-norm and also used for multitask learning [68], [69] and semisupervised learning [70]. It is defined as

$$\|A\|_{2,1} = \sum_{i=1}^{u} \sqrt{\sum_{j=1}^{v} A_{ij}^2} = \sum_{i=1}^{u} \|A^i\|_2. \quad (14)$$

The $l_{2,1}$-norm encourages row sparsity, i.e., it encourages entire rows of the matrix to have zero elements. More specifically, minimizing the l_1-norm promotes row sparsity, while minimizing the l_2-norm promotes nonsparsity within the rows. A common feature of the previous approaches using the Frobenius norm and l_1-norm is that they treat the two indices i and j in the same way. However, these two indices have different meaning, such as the spatial dimensions and data points, respectively. In the strict matrix format, this subtle distinction is easy to get lost. $l_{2,1}$-norm captures this subtle distinction, which has some structural information.

When $r = 2,\ p = 0$, it is the $l_{2,0}$-norm, which is defined in [71] and has been successfully used in the multitask feature learning (MTFL) problem

$$\|A\|_{2,0} = \sum_{i=1}^{u} \left\| \sum_{j=1}^{v} A_{ij}^2 \right\|_0. \quad (15)$$

The $l_{2,p}$-norm ($0 < p \leq 1$) has been proposed in [72] and has been successfully used in feature selection [73].

When $r = \infty,\ p = 1$, it is the $l_{\infty,1}$-norm, which is defined as follows:

$$\|A\|_{\infty,1} = \sum_{i=1}^{u} \max_j |A_{ij}|. \quad (16)$$

In recent years, the $l_{\infty,1}$-norm has been proposed for joint regularization [74]–[77]. As seen from the definition, the $l_{\infty,1}$-norm is a matrix norm that penalizes the sum of maximum absolute values of each row. Similar to $l_{2,1}$-norm, this regularizer also encourages row sparsity. Thus, the $l_{\infty,1}$-norm can be naturally used in feature selection. Other applications of $l_{\infty,1}$ regularization are simultaneous sparse signal approximation [74], [75], structure learning [77], multiview learning [78], and multitask learning [79].

B. Physical Meaning of $l_{r,p}$-Norm of a Matrix: Positive Correlation Versus Negative Correlation

Suppose that there are v classes and the dimensionality of each example is u for the matrix $A \in R^{u\times v}$. If we require most rows of A to be zero, we have $0 \leq p \leq 1$ in (12) and only $p = 1$ is convex. The choice of r depends on what kind of correlation assumption among classes.

The most usual assumption is that the correlation between different classes is positive, i.e., different classes share as many identical features as possible. This assumption corresponds to the case $1 < r \leq \infty$. Increasing r corresponds to allowing more classes to share the same features.

The other assumption is that the correlation among classes is negative [80], [81], i.e., if a feature is important for one or few classes, it becomes less important for the other classes. This assumption can be implemented by imposing sparsity penalization on the rows of A, and thus, the range of r is $0 \leq r \leq 1$. This assumption will lead to the choice of specific, but maybe not class specific, features. Zhou *et al.* [81] used negative correlation assumption through the $l_{1,2}$-norm. The choice of $0 \leq r, p \leq 1$ would yield very sparse coefficient matrix A, which is not only row sparse but also column sparse. However, recently, there is little empirical or theoretical analysis for this problem.

C. $l_{2,1}$-Norm Regularized/Constrained Feature Selection

1) Efficient and Robust Feature Selection via Joint $l_{2,1}$-Norm Minimization: Nie *et al.* [26] aim to learn a linear function $y = x^T W + b$, such that for n training examples, $Y^i \approx X_i^T W + b$, i.e., $\min_{W,b} \|Y^i - X_i^T W - b\|_2$. For simplicity, the bias b can be absorbed into W when the constant value 1 is added as an additional dimension for each data. Thus, the problem becomes: $\min_W \|Y^i - X_i^T W\|_2$. The objective function of an RFS algorithm [26] is formulated as follows:

$$\min_W \sum_{i=1}^{n} \|Y^i - X_i^T W\|_2 + \alpha\|W\|_{2,1} = \min_W \|Y - X^T W\|_{2,1} + \alpha\|W\|_{2,1} \quad (17)$$

where $\|W\|_{2,1}$ is for feature selection. Another FS algorithm, namely, similarity preserving feature selection (SPFS) [82], has a similar objective formulation to RFS. The differences of SPFS and RFS are detailed in [83]. Note that both SPFS and RFS only aim to preserve the global similarity structure. Neither of them considers the local geometric structure of data. To solve this problem, a global and local structure preservation framework for feature selection [83] is proposed. Zhu *et al.* [84] proposed a method named regularized self-representation (RSR), which regresses to each example itself instead of its label. More specifically, the objective function of RSR is

$$\min_W \|X^T - X^T W\|_{2,1} + \alpha\|W\|_{2,1}. \quad (18)$$

2) Correntropy-Induced Robust Feature Selection: RFS [26] is sensitive to outliers. Since correntropy [85] can enhance robustness, correntropy for RFS (CRFS) [86] is proposed for better robustness, and thus, the objective function is defined as follows:

$$\min_W \sum_{i=1}^{n} \phi((X^T W - Y)^i) + \lambda\|W\|_{2,1} \quad (19)$$

where $\phi()$ is the robust M-estimator and can be defined, such as Cauchy M-estimator and Welsch M-estimator,

and $(X^T W - Y)^i$ is the ith row of $X^T W - Y$. The objective function can be optimized by additive form of half-quadratic or multiplicative form of half-quadratic [87]. According to the additive form of HQ, the objective function (when $\phi()$ is Huber loss function [87]) is equivalent to

$$\min_{W,E} \|Y - X^T W - E\|_F^2 + \alpha \|W\|_{2,1} + \beta \|E\|_1 \tag{20}$$

where $E^i = \delta((X^T W - Y)^i)$ and $\delta()$ is the minimizer function with respect to $\phi()$.

3) Discriminative Least Squares Regression for Feature Selection: Least squares regression (LSR) aims to learn a linear function $y = x^T W + b$, such that for n training examples, $Y^i \approx X_i^T W + b$, i.e., $Y \approx X^T W + e_n b$. To further increase the discriminative ability of LSR, discriminative LSR (DLSR) [88] is proposed. DLSR introduces a technique called ε-dragging, such that $Y \approx X^T W + e_n b - B \odot M$, where $\odot$ is a Hadamard product operator of matrices. The matrix $M \in R^{n \times c}$ records these ε in ε-dragging, and the matrix $B \in R^{n \times c}$ is defined as

$$B_{ij} = \begin{cases} 1, & \text{if } x_i \text{ is in the } j\text{th class} \\ -1, & \text{otherwise.} \end{cases} \tag{21}$$

Thus, the objective of DLSR for feature selection (DLSR-FS) is defined as

$$\min_W \|Y - X^T W - e_n b + B \odot M\|_{2,1} + \alpha \|W\|_{2,1}. \tag{22}$$

4) Feature Selection via Joint Embedding Learning and Sparse Regression: Instead of regressing each example to its label [26], [86], [88], the objective of joint embedding learning and sparse regression (JELSR) [89], [90] is to regress each example X_i to its low-dimensional embedding $Z_i \in R^m$, where m is the dimensionality of embedding. The objective function of JELSR can be formulated as follows:

$$\min_{W, ZZ^T = I_{m \times m}} tr(ZLZ^T) + \beta \|W^T X - Z\|_2^2 + \alpha \|W\|_{r,p}^p \tag{23}$$

where $Z = [Z_1, Z_2, \ldots, Z_n] \in R^{m \times n}$ and L is the graph Laplacian. The first term of (23), $\arg\min_{ZZ^T = I_{m \times m}} tr(ZLZ^T)$, is exactly the same as [91], which is to find the low-dimensional embedding of each example. The second term of (23) is to regress each example to its low-dimensional embedding. The third term is designed for FS, which is the same as [26]. Compared with [26], the improvement of [89] over [26] is to regress each example to its low-dimensional embedding rather than to its label. Two prominent approaches, i.e., multicluster feature selection [92] and minimum redundancy spectral feature selection [93] can be regarded as special cases of JELSR.

5) Feature Selection by Joint Graph Sparse Coding: Zhu *et al.* [94] proposed a feature selection method by joint graph sparse coding (JGSC). JGSC considers both manifold learning and regression simultaneously to perform feature selection. Using the bases to represent the data has been proved to make the learning process easier and leads to better results in practice than the traditional ones, such as raw pixel intensity values. Thus, JGSC first extracts the bases of the data and, then, represents the data sparsely using the extracted bases. The objective function of JGSC is defined as follows:

$$\begin{aligned} &\min_{B,W} tr(WLW^T) + \beta \|X - BW\|_2^2 + \alpha \|W\|_{2,1} \\ &\text{s.t. } \sum_{i=1}^{n} \sum_{j=1}^{d} B_{ij}^2 \le 1 \end{aligned} \tag{24}$$

where X is the data matrix, B is the base (or dictionaries) to be learnt, W is the sparse code to be learnt, L is the Laplacian matrix obtained by the built k-nearest-neighbor graph, and $\sum_{i=1}^{n} \sum_{j=1}^{d} B_{ij}^2 \le 1 (B_{ij}$ is the element in the ith row and the jth column of $B)$ keeps B from having arbitrarily large values, which would lead to very small values of W. The three terms of the objective function are designed to address the manifold learning, regression process, and feature selection, respectively.

6) Unsupervised Feature Selection: Maximum margin criterion (MMC) [95], [96] is a supervised subspace method, a variant of linear discriminant analysis (LDA). The objective function is defined as follows:

$$\min_{W^T W = I} tr(W^T (S_w - S_b) W) \tag{25}$$

where S_w is the within-class scatter matrix and S_b is the between-class scatter matrix [9]. Inspired by MMC, a unsupervised maximum margin feature selection algorithm via sparse constraints is proposed in [97]. A unsupervised discriminative feature selection (UDFS) method is proposed in [98] by combining local discriminative analysis and $l_{2,1}$-norm minimization.

7) Joint Feature Selection and Subspace Learning: The key point of [99] is to add $l_{2,1}$-norm penalty into graph embedding [100]. Its objective function is defined as follows:

$$\begin{aligned} &\min_W \|W\|_{2,1} + \alpha tr(W^T X L X^T W) \\ &\text{s.t. } W^T X D X^T W = I \end{aligned} \tag{26}$$

where the first term is for feature selection and the second term, $\arg\min_{W^T X D X^T W = I} tr(W^T X L X^T W)$, is the objective function of graph embedding.

8) Toward Feature Selection in Networks: Based on Laplacian regularized least squares (LapRLS), Gu and Han [101] presented a supervised feature selection method for networked data. The objective function of LapRLS is formulated as follows:

$$\min_W \|Y - X^T W\|_F^2 + \alpha \|W\|_F^2 + \beta tr(W^T X L X^T W) \tag{27}$$

where L is graph Laplacian [102]. More specifically, the first term in the objective function of LapRLS is the traditional LSR. The second term controls the complexity of the linear classifier. The third term is graph regularization. Based on LapRLS, the objective function of [101] is defined as follows:

$$\min_W \|Y - X^T W\|_F^2 + \alpha \|W\|_F^2 + \beta tr(W^T X L X^T W) + \gamma \|W\|_{2,1} \tag{28}$$

where the last term is for feature selection.

D. $l_{\infty,1}$-Norm Regularized/Constrained Feature Selection

Masaeli *et al.* [103] proposed linear discriminant feature selection, which combines LDA with feature selection as follows:

$$\min_W tr((W^T S_w W)^{-1} W^T S_b W) + \alpha \|W\|_{\infty,1} \tag{29}$$

where S_w is the within-class scatter matrix, S_b is the between-class scatter matrix [9], and the last term is for feature selection. The main problem of this method is that the computational cost of the gradient of $tr((W^T S_w W)^{-1} W^T S_b W)$ is very expensive. Thus, it is only limited to small-scale data.

E. $l_{2,0}$-Norm Regularized/Constrained Feature Selection

Sparse feature selection [71] selects features by solving a smoothed general loss function with a $l_{2,0}$-norm constraint. Cai *et al.* [104] proposed a feature selection approach (FS20), which has one $l_{2,1}$-norm loss function with an explicit $l_{2,0}$-norm equality constraint. The objective function to select k features in the multiclass problems is defined as

$$\begin{aligned} &\min_{W,b} \ \|Y - X^T W - e_n b\|_{2,1} \\ &\text{s.t. } \|W\|_{2,0} = k \end{aligned} \tag{30}$$

where $b = [b_1 \dots b_c]$ is the bias vector. Since the regularization parameter of this method has the explicit meaning, i.e., the number of selected features, it alleviates the problem of tuning the parameter exhaustively, making it a pragmatic feature selection approach.

F. How to do Feature Selection?

As for how to do feature selection, we can rank each feature according to $\|W^i\|_2$ (multiclass classification) or $|w_i|$ (binary classification). The larger $\|W^i\|_2$ or $|w_i|$ is, the more important this feature is. We can either select a fixed number of the most important features or set a threshold and select the feature whose $\|W^i\|_2$ or $|w_i|$ is larger than this value [89], [98]. Please note that $W^T x_i$ (multiclass classification) or $w^T x_i$ (binary classification) is also a new representation of x_i using only a small set of selected features. However, multiplying W^T (or w^T) and x_i belongs to the projection method (the same as PCA) and does not belong to feature selection.

IV. Task-Driven Feature Selection

The focus of this paper is on SSFS. There are some closely related areas for specific tasks with rich literature [105], [106].

A. Multitask Feature Selection

Liu *et al.* [107] proposed a multitask feature selection method called MTFL, which uses $l_{2,1}$-norm instead of l_1-norm as the penalty. To obtain the projection matrix $W \in R^{d \times c}$, the objective function of [107] is defined as

$$\min_W \|Y - X^T W\|_F^2 + \alpha \|W\|_{2,1}. \tag{31}$$

Compared with [24], [25], and [40], the improvement of [107] over the former three is to use the $l_{2,1}$-norm instead of l_1-norm as the penalty. Reference [26] uses the Frobenius norm to regress each example to its label, while [107] uses $l_{2,1}$-norm to regress each example to its label. The commonality of [26], [40], and [107] is that all of their objective functions are to regress each example to its label.

In [108] and [109], a new multitask feature selection method based on MTFL is proposed. Given that some important variables are only correlated with a subset of tasks, the $l_{2,1}$-norm cannot handle them properly. Thus, l_1-norm regularizer is added to impose the sparsity among all elements in W, and the objective function is formulated as

$$\min_W \|Y - X^T W\|_F^2 + \alpha \|W\|_{2,1} + \beta \|W\|_1. \tag{32}$$

Based on [107], Tang and Liu [110] proposed a feature selection algorithm for social media data. The objective function of linear discriminant dimensionality reduction [111] can be relaxed into a form, which is similar to (31). However, the motivations between [111] and [107] are different. MTFL [107] proposed a multitask feature selection method, while LDRR [111] aimed at integrating Fisher score and LDA in a unified framework. Furthermore, their optimization methods are different. Liu *et al.* [107] used the Euclidean projection, while Gu *et al.* [111] used the accelerated proximal gradient descent algorithm.

The well-known multitask feature selection methods in [112] also use the $l_{2,1}$-norm. Liang *et al.* [113] proposed another multitask feature selection method, and the objective function is defined as

$$\min_{W,b} \|Y - X^T W - e_n b\|_F^2 + \alpha \|W\|_{r,p} \tag{33}$$

where $b = [b_1 \ \dots \ b_c]$ is the bias vector, p is set to be 1, and r is set to be 1 and ∞ for the positive and negative correlation assumptions, respectively. The exclusive lasso [81] for multitask feature selection adopts the negative correlation assumption but without row-sparse constraints and corresponds to the case $r = 1$, $p = 2$. A multitask feature selection method via maximum entropy discrimination (MED) was proposed in [114], and the experimental results show that MED multitask learning outperforms single-task learning.

B. Multilabel Feature Selection

The multilabel learning [115] can be viewed as a very special case of multitask learning, where each task is the binary classification problem associated with each class label [116]. Huang *et al.* [117] proposed a feature selection method for multilabel multiclass learning. In particular, feature correlation is added into the objective function, so that feature correlation and feature selection are conducted simultaneously. This method is called FCFS, and the objective function is defined as

$$\min_W \|Y - X^T W\|_F^2 + \alpha \|W\|_{2,1} + \beta tr(W^T C W) \tag{34}$$

where the element in the ith row and the jth column of C, C_{ij}, is the absolute value of the correlation coefficient between the ith feature and the jth feature. FCFS can be seen as a variant of MTFL [107], which adds feature correlation based on MTFL [107].

In [116], a probabilistic multilabel learning model based on novel sparse feature learning is proposed. By employing l_1-norm and $l_{2,1}$-norm, the proposed model has the capacity of capturing both label interdependences and common predictive model structures.

Based on [26] and [118], Ma *et al.* [119] proposed a novel feature selection method named subfeature uncovering with sparsity (SFUS). SFUS can uncover the shared subspace of original features, which is beneficial for multilabel learning.

C. Multiview Feature Selection

To better understand, classify, and search video and image information, many features have been proposed, such as color, texture, or shape. How to integrate these multiview features [120] and identify the important ones from them for specific tasks has become an increasingly important problem. To solve this problem, Xiao *et al.* [121], [122] proposed two-view feature selection methods.

In [123]–[125], multiview feature selection can be performed beyond the limit of two views. To better demonstrate the technical details of these methods, important notations used in this section are presented. Given a multiview feature data set $X \in R^{d\times n}$ with n examples and k views (each view has a particular meaning and statistical property, e.g., texture, color, and shape), the dimension for the ith view representation is d_i, i.e., $d = \sum_{i=1}^{k} d_i$.

In multiview features fusion, the features of a specific view can be more or less discriminative for specific classes. For example, the color features substantially increase the detection of stop signs, while they are almost irrelevant for finding planes in images. Thus, a new l_1-norm group (G_1-norm) is proposed [126], which is defined as $\|W\|_{G_1} = \sum_{i=1}^{c}\sum_{j=1}^{k}\|W_i^j\|_2$ and $W_i^j \in R^{d_j}$.

The objective function of [125] is defined as

$$\min_W \|Y - X^T W\|_F^2 + \alpha\|W\|_{2,1} + \beta\|W\|_{G_1}. \tag{35}$$

In [123], the hinge loss is used, because the hinge loss-based support vector machine (SVM) has shown the state-of-the-art performance in classifications. The only difference between the objective function of [123] and [125] is that, the former uses the hinge loss, while the latter uses the LSR. Wang *et al.* [126] proposed a joint classification and regression learning model based on G_1-norm and $l_{2,1}$-norm.

Gui *et al.* [127] proposed a framework of joint feature extraction and feature selection for multiview learning. The contributions of this paper are twofold. First, Gui *et al.* [127] consider not only the independent information of each view and the complementary properties of different views, but also view consistency in linear multiview feature extraction. In particular, view consistency models the correlations between all possible combinations of any two kinds of view. Second, Gui *et al.* [127] simultaneously perform feature extraction and feature selection for multiview learning based on the $l_{2,1}$-norm of the projection matrix.

D. Simultaneous Feature Selection and Classification

SVM is a widely used classification technique [128]. However, a major limitation is that the SVM cannot perform automatic feature selection [129], [130]. A sparse representation of SVM with respect to input features is desirable for many applications. Wang *et al.* [131] proposed the hybrid huberized SVM with a combination of l_1-norm and l_2-norm. In [132], by introducing a 0–1 control variable to each input feature, l_0-norm sparse SVM (SSVM) is converted into a mixed integer programming (MIP) problem. Rather than directly solving the MIP, an efficient cutting plane algorithm combining with multiple kernel learning is proposed to solve its convex relaxation. Fung and Mangasarian [133] proposed an algorithm, which not only performed feature selection but also conducted data selection for SVM. Sun *et al.* [134] combined feature selection and l_1-l_2 support vector regression for cancer prognosis. However, most feature selection algorithms [131]–[134] can only be used for the continuous data type. In [135], a novel method for SSVMs with $l_p(p < 1)$ regularization is proposed, which can achieve simultaneous classification and feature selection with both continuous and discrete data.

Most of the above algorithms with sparse representation [131]–[135] mainly focus on binary classification and cannot solve multiclass classification directly. Furthermore, complex optimization procedures are often required. Cawley *et al.* [136] developed sparse multinomial logistic regression via Bayesian l_1 regularization. This algorithm can deal with multiclass feature selection. Extensive experimental results demonstrated that it is a powerful feature selection algorithm [136]. A new feature selection algorithm based on the Gauss–Seidel method was proposed for the sparse logistic regression (SLogReg) problem [137]. The proposed method was simple and extremely easy to implement. The performance of this method is determined by the value of a regularization parameter, which must be carefully tuned in order to optimize performance. Generally speaking, the best regularization parameter is difficult to obtain in practice and choosing a suitable regularization parameter via cross validation is time-consuming. In [138], it is demonstrated that a simple Bayesian approach can be taken to eliminate the regularization parameter. The improved algorithm is typically two or three orders of magnitude faster than the original algorithm, since there is no longer a model selection step. In [42], a novel method for SLogReg with $l_p(p < 1)$ regularization is proposed. A linear SLogReg model [139] is proposed to combine feature selection and classification into a regularized optimization problem with the constraint of group lasso. Similar to [135], this method can also conduct classification and feature selection simultaneously. Cai *et al.* [140] proposed a new $l_{2,1}$-norm SVM, that is, multiclass hinge loss with $l_{2,1}$-norm regularization term to naturally select the features for multiclass without bothering further heuristic strategy. Different from [129] and [131]–[135], which focus on binary classification, this method [140] do multiclass FS directly.

E. Simultaneous Feature Selection and Clustering

Li *et al.* [141] proposed a feature selection method by combining feature selection with clustering for single view data, namely, nonnegative discriminative feature selection (NDFS). The objective function of NDFS is

defined as

$$\min_{W,F} \|F - X^T W\|_F^2 + \beta tr(F^T L F) + \alpha \|W\|_{2,1}$$
$$s.t.\ F^T F = I_c, \quad F \geq 0 \tag{36}$$

where $F \in R^{n \times c}$ is the scaled cluster indicator matrix and L is the normalized graph Laplacian matrix. The same as [142], the matrix F satisfies $F^T F = I_{c \times c}$, where I_c is the $c \times c$ identity matrix. The first term of (36) is for regression. The second term of (36) characterizes the local geometrical structure [143], [144]. The third term of (36) is for feature selection.

Wang *et al.* [124] proposed a feature selection method by combining feature selection with clustering for multiview data. The objective function in [124] is defined as

$$\min_{W,F} \|F - X^T W - e_n b\|_F^2 + \beta \|W\|_{G_1} + \alpha \|W\|_{2,1}$$
$$\text{s.t. } F^T F = I_c \tag{37}$$

where $F \in R^{n \times c}$ is the scaled cluster indicator matrix, $b = [b_1 \ldots b_c]$ is the bias vector, and group l_1-norm (G_1-norm) is defined in [123] and [125].

Witten and Tibshirani [145] proposed a framework for feature selection in sparse clustering. They applied l_1-norm as feature selection method embedded in clustering. This framework can be applied to any similarity-based clustering algorithms, such as k-means clustering and hierarchical clustering.

Zeng and Cheung [146] aim to obtain an appropriate representation of the data through FS or kernel learning for local learning-based clustering (LLC). More specifically, a weight is associated with each feature or kernel and incorporated into the regularization of LLC to consider the relevance of each feature or kernel for clustering. Accordingly, the weights are computed iteratively in the whole process. The resulting weighted regularization with an additional constraint on the weights is equivalent to the sparse penalty. Extensive experimental results show that the proposed method is both effective and efficient.

F. Some Other Methods

Instead of considering k-nearest neighbor or ε-ball-based methods as in typical graph construction [147], Xu *et al.* [148] proposed the l_1 graph based on sparse representation for feature selection. That is to say, the only difference between [147] and [148] is that, the former uses k-nearest neighbor to construct the graph, while the latter uses the l_1 graph based on sparse representation.

Considering supervised learning (focusing on logistic regression) in the presence of very many irrelevant features, Ng [149] shows that using the l_1-norm regularization of the parameters, the sample complexity (i.e., the number of training examples required to learn well) grows logarithmically in the number of irrelevant features. This logarithmic rate matches the best known bounds for feature selection. Ng [149] also shows that the logistic regression with l_2-norm regularization has a worst case sample complexity that grows at least linearly in the number of irrelevant features. Sun *et al.* [150] also consider feature selection for classification with a large number of irrelevant features. The key idea is to decompose an arbitrarily complex nonlinear problem into a set of locally linear ones through local learning and then learn feature weights globally with l_1 penalty and the large margin framework. More specifically, one of the most popular margin formulations, logistic regression, is used.

A definition of relevancy based on the spectral properties of the Laplacian of the features measurement matrix is presented in [151]. Then, feature selection is based on a continuous ranking of the features defined by a least-squares problem. An interesting property of the feature relevance function is that the sparse solutions for the ranking values naturally occur as a result of a biased nonnegativity of a key matrix in the whole process. The experimental results show that this method typically achieves high accuracy even when only a small fraction of the features are relevant.

Feature selection with specific multivariate performance measures is critical to the success of many practical applications, such as text classification and image retrieval. Most feature selection methods are usually designed for minimizing classification or clustering error. Mao and Tsang [152] proposed a generalized sparse regularizer. Based on this regularizer, a unified feature selection framework for general loss functions was presented. More specifically, the novel feature selection algorithm by optimizing multivariate performance measures was studied.

TABLE I
BRIEF DESCRIPTION OF ALL DATA SETS

Data set	Number of examples	Dimensionality	Classes
WebKB-WC	1210	4189	7
coil20	1440	256	20
AR	840	768	120
UMIST	575	2576	20
protein	116	20	6
vehicle	846	18	4
CAR	174	9182	11
LUNG	203	3312	5
MLL	72	5848	3
GLA	180	49151	4
TOX	171	5748	4

V. EXPERIMENTS

A. Data Sets

We did our experiments on 11 data sets: one WebKB data set WebKB-WC,[2] one image classification data set coil20 [153], two face data sets including AR and Umist,[3] two UCI data sets[4] including protein and vehicle, and five microarray data sets including CAR [154], LUNG [155], MLL [156], GLA [157], and TOX.[5] For AR, the face images of 120 persons are used to construct a data set [158], and seven images of each person are randomly selected. The brief description of all data sets is shown in Table I.

[2] http://www.cs.cmu.edu/~webKB/
[3] http://images.ee.umist.ac.uk/danny/database.html
[4] http://archive.ics.uci.edu/ml/
[5] http://featureselection.asu.edu/datasets.php

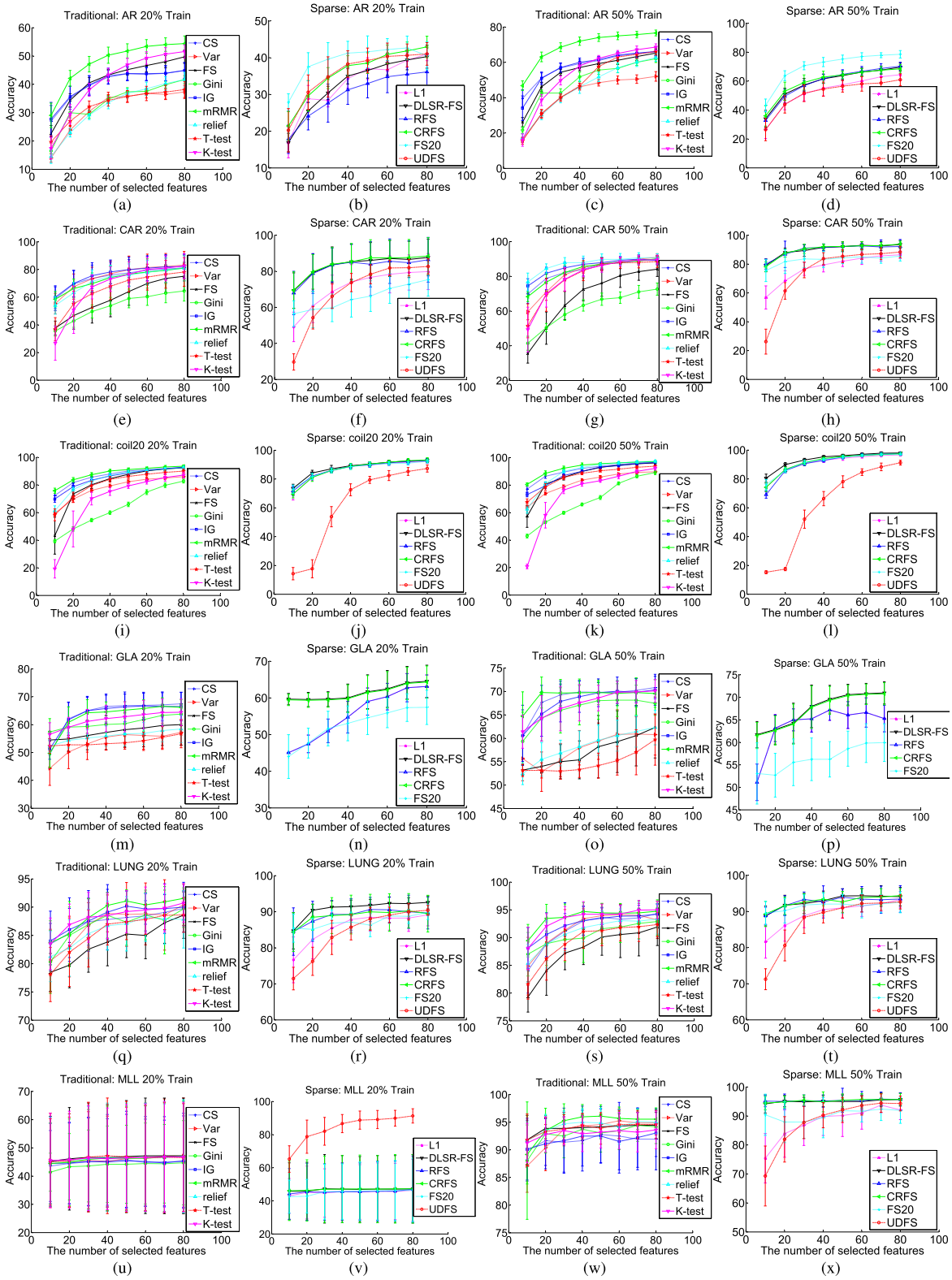


Fig. 3. Group I: classification accuracy versus the number of selected features. Eight different numbers of selected features are evaluated, as indicated by the horizontal axis. Four figures are used to illustrate the accuracy and the standard deviation for each data set. The first two figures show the results obtained by traditional and sparse methods, respectively, when 20% are for training. The last two figures show the results when 50% are for training.

B. Algorithms and Parameter Settings

We have done two groups of experiments:

1) Nine traditional feature selection methods, including chi-square (CS), variance, Fisher score (FS) [12], [13], Gini coefficient (Gini) [159], information gain (IG) [160], minimum redundancy maximum relevance (mRMR) [161], relief [162], students T-test (T-test) [163], Kruskal–Wallis test (K-test) [164].[6]

[6]The source codes of CS, FS, and Gini are available at http://featureselection.asu.edu/.

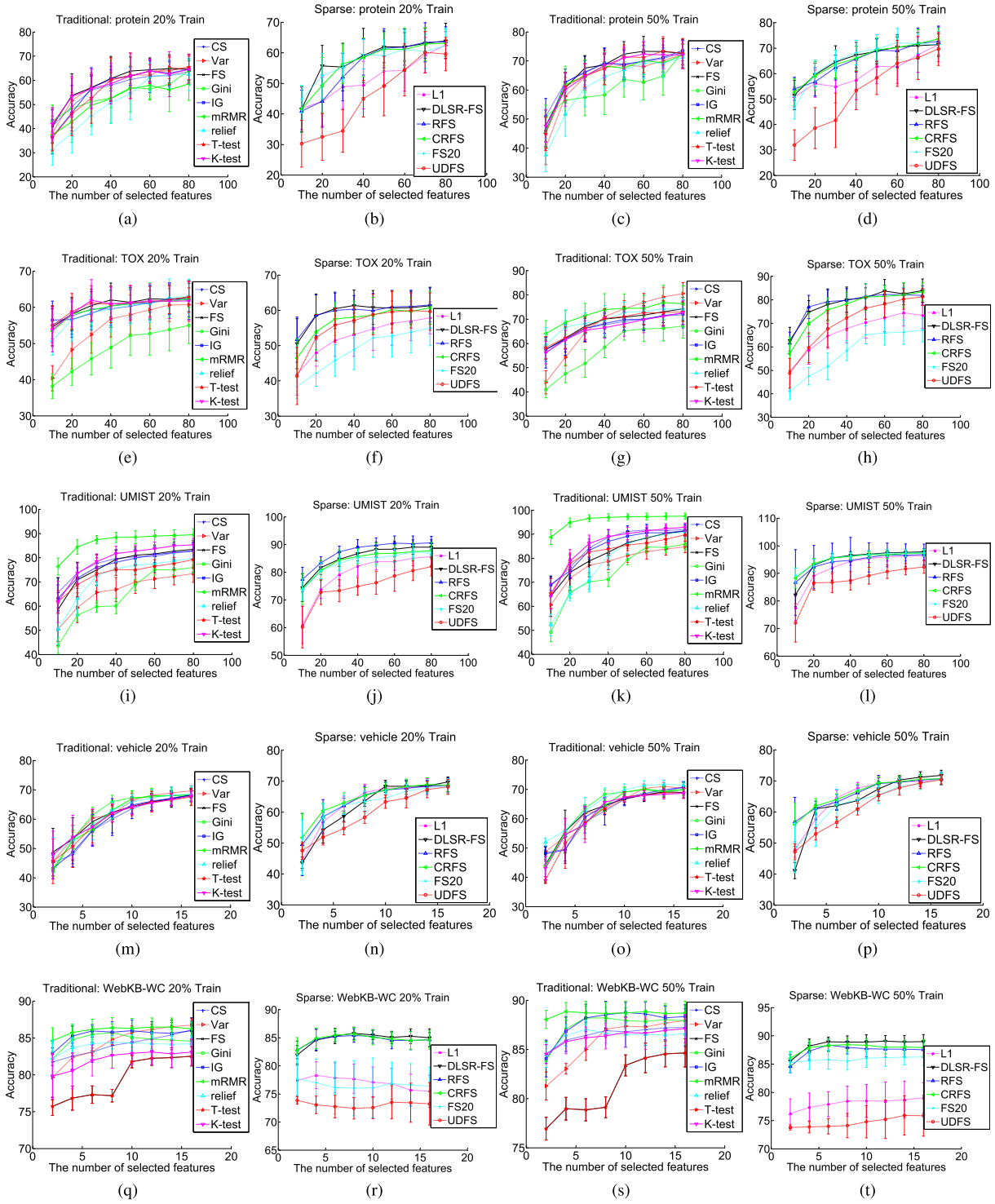


Fig. 4. Group II: classification accuracy versus the number of selected features. Eight different numbers of selected features are evaluated, as indicated by the horizontal axis. Four figures are used to illustrate the accuracy and the standard deviation for each data set. The first two figures show the results obtained by traditional and sparse methods, respectively, when 20% are for training. The last two figures show the results when 50% are for training.

2) Six SSFS methods, including the L1 [24], [25], [40], DLSR-FS [88], RFS [26], CRFS [86], FS20 [104], and UDFS [98].

The feature selection performance is evaluated by average classification accuracy using LibSVM. The free parameters for the tested methods were determined in the following ways.

1) The regularization parameter λ of all SSFS methods is set by tenfold cross validation from {0.01, 0.1, 1, 10, 100, 1000, 10 000, 100 000}.

TABLE II

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 20% EXAMPLES ARE USED FOR TRAINING ON THE DATASET AR, CAR, COIL20 AND GLA

Method\Data set	AR	CAR	Coil20	GLA
CS	44.9250±2.7716	81.2406±9.4074	92.7802±1.1266	**67.5000±4.0837**
Variance	37.3333±2.3202	83.2707±9.8145	86.0259±1.0743	57.2535±4.6824
FS	49.7333±2.3419	75.1504±10.0973	92.4310±1.1963	60.0352±7.3384
Gini	41.7000±1.3960	64.5113±7.2727	82.7802±1.3576	63.8028±3.5450
IG	44.9250±2.7716	82.2932±8.9327	92.7931±1.2150	66.5141±5.1533
mRMR	**54.3917±2.6974**	81.0902±9.0077	**93.5474±0.9342**	66.1972±2.7456
relief	41.6667±2.6103	80.9023±9.0613	92.4698±1.1024	58.4507±4.9044
T-test	38.1833±2.8460	78.2331±10.3188	90.0172±1.6058	56.7254±5.0068
K-test	51.5833±2.3644	82.6692±8.9769	87.1379±2.1427	64.4718±4.7423
L1	40.9750±1.7898	79.8496±10.0105	92.2284±1.2122	63.1338±3.1358
DLSR-FS	40.3000±3.1502	87.6692±10.1862	93.0647±1.2524	64.3310±4.3938
RFS	36.1417±4.9864	86.3910±12.4183	93.0129±1.2791	63.1338±3.1358
CRFS	42.9583±2.9025	**88.4962±10.1465**	93.1552±1.6949	64.3310±4.3938
FS20	40.5500±3.8258	75.3759±9.1497	92.3621±1.2061	57.4296±4.6685
UDFS	41.0333±3.0730	82.7068±4.2400	87.3578±2.4767	± -

TABLE III

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 50% EXAMPLES ARE USED FOR TRAINING ON THE DATASET AR, CAR, COIL20 AND GLA

Method\Data set	AR	CAR	Coil20	GLA
CS	66.0000±2.2896	89.3529±2.2159	96.8819±0.6682	70.8427±2.3961
Variance	52.0556±2.7080	91.0000±2.0470	89.7014±0.8950	60.7865±4.3791
FS	65.2639±2.2004	84.1176±4.1843	96.1736±0.8877	62.3034±6.5079
Gini	62.0556±2.1416	72.6471±3.5073	89.1389±1.4859	67.4719±2.5739
IG	66.1528±2.4236	89.2941±2.6542	96.4444±0.8068	70.3371±3.3217
mRMR	76.6389±1.6525	90.1765±1.8704	97.1667±0.5683	69.5506±2.9493
relief	62.7500±2.1280	90.8235±2.1565	97.4236±0.6668	62.3034±3.7006
T-test	66.1806±1.9874	88.5294±3.4274	93.7778±0.9146	59.7191±5.4616
K-test	68.6528±1.8221	89.6471±2.6201	92.1667±1.2920	70.1124±2.4462
L1	64.4861±3.9109	86.4706±2.5369	96.7083±0.8641	65.2809±2.9493
DLSR-FS	69.6667±2.8007	93.5882±2.7459	**97.9931±0.6521**	**70.9865±2.4617**
RFS	70.3194±2.7698	92.2353±4.8421	97.6736±0.5501	65.2809±2.9493
CRFS	68.5000±2.9455	**93.8824±2.0913**	97.4583±0.8040	70.7865±2.4617
FS20	**78.5694±2.5913**	85.5882±3.4072	97.1458±0.8044	59.9438±4.1781
UDFS	61±4.0516	88.4118±3.9421	91.2778±1.6751	-

2) The regularization parameter C of LibSVM is set by tenfold cross validation[7] from {0.0001, 0.001, 0.01, 0.1, 1, 10, 100}.

Since the two data sets (GLA and WebKB-WC) are large, choosing the suitable parameters via cross validation is very time-consuming. We select a proper C, which has the best performance, from the candidate set {0.0001, 0.001, 0.01, 0.1, 1, 10, 100} using variance. For other algorithms, we use the same C as variance. Then, we select a proper λ, which has the best performance from the candidate set {0.01, 0.1, 1, 10, 100, 1000, 10 000, 100 000} using DLSR-FS, and use the same λ for the other SSFS methods except FS20. For FS20, default parameters are used for these two data sets (GLA and WebKB-WC). We still cannot get the results of UDFS for the data set GLA even though this code has run two months on a server with CPU 2.80-GHz and 128-GB memory.

[7]The process of tenfold cross validation is described here. We first split the whole data set into a training set and a testing set, and then, we take the training set and split it into tenfolds. During the cross validation, we take ninefolds for training and the left fold for testing, and repeat the process ten times and choose the parameter settings $\{\lambda, C\}$ with the highest average accuracy. Then, the parameters will be used for the whole training set and classify the testing set. For the traditional feature selection methods, only parameter C in LibSVM needs to be set by cross validation. In the whole process, the number of selected features d is fixed.

C. Experimental Results and Discussion

The accuracy curves of all feature selection methods on the 11 data sets are shown in Figs. 3 and 4. The final accuracy is calculated as the mean of the 20 random splits. In total, eight different numbers of selected features are illustrated. For the two low-dimensional data sets, protein and vehicle, the number of features is set to be $[2, 4, 6, \ldots, 16]$, respectively. For the remaining nine data sets, the number of features to be selected is $[10, 20, 30, \ldots, 80]$, respectively. For more clarity, four figures are used to illustrate the accuracy and standard deviation for each data set. The first two figures show the results obtained by traditional and sparse methods, respectively, when 20% are for training. The last two figures show the results when 50% are for training. Furthermore, the mean accuracy and the standard deviation of 20 trials are shown in Tables II–VII. These values are obtained with $d = 16$ for the data sets, protein and vehicle, and $d = 80$ for the remaining nine data sets.

From the experimental results, we can safely draw the following three conclusions.

TABLE IV

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 20% EXAMPLES ARE USED FOR TRAINING ON THE DATASET LUNG, MLL, PROTEIN AND TOX

Method\Data set	LUNG	MLL	protein	TOX
CS	89.7813±3.0827	45.1754±16.1639	63.6111±4.4842	62.3704±4.9336
Variance	90.4375±3.6502	47.1930±20.4662	64.7778±5.2246	60.7407±4.6378
FS	88.5938±4.1026	47.1930±20.4662	64.5556±6.2252	62.7407±3.9063
Gini	89.7188±3.2922	44.7368±14.9329	64.0556±4.2057	55.0000±5.0069
IG	90.0000±4.4061	45.4386±16.8174	64.5556±4.4152	62.0000±5.3881
mRMR	91.5625±1.8488	46.8421±19.6937	58.5556±6.8412	62.2593±5.1024
relief	87.8438±5.0678	47.0175±20.0538	62.3889±6.6134	**63.0741±4.7331**
T-test	88.6250±3.0657	46.5789±19.1178	**65.5556±5.3171**	62.9259±4.3222
K-test	90.7500±3.4210	46.7544±19.4513	64.5556±5.8151	61.7778±5.2556
L1	89.1250±3.0388	46.4035±18.6285	62.5556±4.3759	57.8519±4.5234
DLSR-FS	**92.5938±1.7927**	47.2807±20.6726	63.8889±5.8108	61.2963±5.2633
RFS	89.6563±4.4303	47.0175±20.0691	63.6667±4.8699	61.5185±4.8317
CRFS	89.4688±5.0863	47.2807±20.6726	63.1667±4.4621	60.7407±4.6673
FS20	89.8750±2.0310	46.5789±19.1017	62.3333±5.7079	55.0000±5.0069
UDFS	90.4063±1.3589	**91.3158±4.3143**	59.6111±5.4690	59.6667±5.3183

TABLE V

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 50% EXAMPLES ARE USED FOR TRAINING ON THE DATASET LUNG, MLL, PROTEIN AND TOX

Method\Data set	LUNG	MLL	protein	TOX
CS	94.3500±1.7685	91.9444±5.5486	72.9825±4.1293	71.7816±5.4215
Variance	93.2000±3.3705	94.8611±2.2004	72.2807±3.6633	80.5747±4.4798
FS	91.9000±3.3302	94.4444±2.3241	72.6316±4.8492	74.2529±4.7308
Gini	92.9000±2.5080	94.1667±3.0302	73.0702±4.0112	67.1839±5.0296
IG	94.2000±1.8055	93.0556±4.2583	**73.5965±3.5729**	72.4138±4.6120
mRMR	94.7500±2.0463	95.5556±2.0412	72.0175±3.6998	76.3793±7.3462
relief	93.3500±2.7069	95.0000±2.0787	71.1404±3.2574	74.0805±3.9228
T-test	92.3500±2.1972	94.5833±2.5572	72.6316±5.0968	73.1034±4.0703
K-test	**95.0000±1.5492**	93.6111±3.4134	72.1053±4.8651	72.7011±4.9019
L1	93.3000±2.2825	92.0833±3.3188	71.4912±6.7919	73.3908±5.4676
DLSR-FS	94.1000±2.5865	95.6944±2.2352	71.4035±4.2288	**83.9655±4.9502**
RFS	93.4500±3.6807	95.6944±2.2352	72.6316±6.0367	81.8966±4.1423
CRFS	94.4000±1.9079	**95.6944±2.0554**	73.5965±4.3851	83.4483±3.8536
FS20	92.5500±2.9065	91.9444±4.4704	71.3158±5.4435	67.1839±5.0296
UDFS	92.6500±1.9564	94.3056±2.5572	69.7368±6.5379	81.3218±3.6507

TABLE VI

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 20% EXAMPLES ARE USED FOR TRAINING ON THE DATASET UMIST, VEHICLE, AND WEBKB_WC

Method\Data set	UMIST	vehicle	WebKB_WC
CS	82.7473±3.6562	68.1711±2.4381	86.0766±1.6655
Variance	73.4615±3.6246	69.6534±1.7288	**86.7547±0.8726**
FS	83.4945±3.6769	68.1932±2.0608	82.5207±1.3149
Gini	75.9890±2.2797	67.6106±2.8460	84.5652±1.6566
IG	82.8352±3.5533	68.2301±2.0798	85.9990±1.2486
mRMR	89.5055±2.5106	68.3481±2.0850	86.2474±0.9381
relief	79.6923±2.6258	68.5546±2.7513	84.2029±1.0745
T-test	79.1868±3.6520	67.5664±2.9649	82.5207±1.3149
K-test	85.3187±2.6625	67.8466±1.8662	83.0849±1.8039
L1	85.6703±3.1846	68.6283±2.0347	75.4969±2.8797
DLSR-FS	89.0659±2.6718	**69.7788±1.5795**	85.0052±1.5907
RFS	**90.6703±2.3326**	68.7758±2.2735	84.6480±1.2988
CRFS	87.8022±3.0553	68.7168±2.1561	84.6377±1.6532
FS20	86.9451±1.9203	68.3628±2.4945	76.3820±3.4142
UDFS	82.0549±3.3600	68.2080±2.4332	73.2557±3.7782

TABLE VII

AVERAGE ACCURACY (%) AND THE STANDARD DEVIATION OF THE SELECTED TOP 80 FEATURES WHEN ABOUT 50% EXAMPLES ARE USED FOR TRAINING ON THE DATASET UMIST, VEHICLE, AND WEBKB_WC

Method\Data set	UMIST	vehicle	WebKB_WC
CS	92.0000±2.3883	70.7820±1.7399	87.2434±1.9261
Variance	84.7627±2.4432	70.3910±2.0760	87.9470±0.8387
FS	91.3559±1.8239	68.9810±1.8642	84.6523±1.4377
Gini	85.8983±2.3150	70.0592±2.1928	87.9139±1.5219
IG	91.4237±2.3972	70.7109±1.9604	88.3526±1.2022
mRMR	97.5763±1.2375	69.8697±1.5741	88.7086±1.1946
relief	90.9661±2.0738	71.4810±1.3284	86.6142±1.1474
T-test	89.7627±2.7195	68.8270±1.7534	84.6523±1.4377
K-test	92.9153±1.7870	68.6848±1.7864	87.1275±1.4318
L1	96.3898±1.5507	70.6635±1.6411	79.0066±2.7061
DLSR-FS	**97.8136±1.1165**	**71.7891±1.6306**	**88.9921±1.2402**
RFS	96.8305±3.8055	70.7227±2.0322	87.5497±1.3303
CRFS	97.3729±1.0925	70.6872±1.7607	87.9305±1.4280
FS20	97.5085±1.3053	70.7227±2.0364	86.2666±1.5024
UDFS	92.3220±2.2772	70.3791±1.5861	75.8775±3.6114

First, mRMR usually performs better than the other traditional feature selection methods. From Tables II–VII, we can see that mRMR performs better than the other traditional methods nine times, while the second best method, variance performs better than the other traditional methods six times in all 22 experiments. From Figs. 3 and 4, we can see the same trend.

Second, no single method can always beat the other methods. Comparing all the feature selection methods, including the traditional and SSFS methods in Tables II–VII, DLSR, CRFS, and mRMR perform the best 8, 3 times, and twice, respectively, while CS, relief, IG, T-test,

K-test, variance, RFS, UDFS, and FS20 perform the best only once.

Third, from Tables II–VII, we can see that the SSFS methods perform the best 15 times in all 22 experiments. However, the improvement of the SSFS methods over the traditional methods is marginal. Maybe, combining SSFS methods with deep learning [165] can bring some improvements.

VI. Conclusion

This paper presents a comprehensive survey of various aspects of SSFS. We elaborate on two architectures, i.e., vector-based and matrix-based ones. We believe this survey will help readers to gain a thorough understanding of the SSFS research landscape. As machine learning and data mining develops and expands to new application areas, SSFS also faces new challenges. We represent here some challenges in research and development of SSFS.

The approaches to sparsity learning are very diverse and motivated by various theoretical results, but a unifying theoretical framework is lacking. It further addresses a problem stemming from the very core of the success of this field—a dilemma faced by most machine learning and data mining practitioners; namely, the more the algorithms become available, the more difficult it is to select a suitable one for a data mining task. For regression, we can regress each example to its label or its low-dimensional representation. For penalty, we can use l_1-norm, $l_{2,1}$-norm, or $l_{2,0}$-norm. How to choose an appropriate SSFS method for a specific application can be considered as one of the most promising future lines of work for the SSFS community.

A second line of future research is SSFS for big data with high dimensionality and a large number of examples. High dimensionality causes two major problems for SSFS. One is the so-called curse of dimensionality. As most existing SSFS algorithms have higher time complexity about d, it is very time-consuming and difficult to scale up with high dimensionality. The other difficulty faced by SSFS with data of high dimensionality is the relative shortage of examples. That is, the dimensionality d can sometimes greatly exceed the number of examples n. In such cases, we can consider algorithms, such as those discussed in [166].

Other interesting opportunities for future research will be SSFS with example selection. In the age of big data, the number of examples is extremely large. SSFS can also be extended to example selection, which is a sister issue of scaling up algorithms. Traditional feature selection algorithms perform dimensionality reduction using all training data. When the number of training data set is very large, random sampling [167], [168] is commonly used to sample a subset of examples. However, random sampling does not exploit any data characteristic. The concept of active feature selection is first introduced in [169]. Active feature selection actively selects the examples for feature selection. It avoids pure random sampling and is realized by selective sampling [169] that takes the full use of data characteristics when selecting examples. The key idea of selective sampling is to select only those examples with high probabilities to be informative in determining feature relevance. Selective sampling aims to achieve better results with a significantly smaller number of examples than that of random sampling. Although some selective sampling methods based on class information or data variance have proved effective on representative algorithms [169], more research efforts are needed to investigate the effectiveness of selective sampling over the large volume of SSFS algorithms.

To conclude, we would like to note that, in order to maintain an appropriate size of the article, we had to limit the number of referenced studies. We therefore apologize to the authors of papers that were not cited.

References

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

[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," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[4] S. Alelyani, J. Tang, and H. Liu, "Feature selection for clustering: A review," in *Data Clustering: Algorithms and Applications*, C. C. Aggarwal and C. K. Reddy, Eds. Boca Raton, FL, USA: CRC Press, 2013.

[5] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," in *Data Classification: Algorithms and Applications*, C. C. Aggarwal, Ed. Boca Raton, FL, USA: CRC Press, 2013.

[6] C. Freeman, D. Kulić, and O. Basir, "An evaluation of classifier-specific filter measure performance for feature selection," *Pattern Recognit.*, vol. 48, no. 5, pp. 1812–1826, 2015.

[7] A.-C. Haury, P. Gestraud, and J.-P. Vert, "The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures," *PLoS ONE*, vol. 6, no. 12, p. e28210, 2011.

[8] D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1088–1099, Jul. 2006.

[9] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[10] D. Tao, X. Li, X. Wu, and S. J. Maybank, "General tensor discriminant analysis and Gabor features for gait recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1700–1715, Oct. 2007.

[11] D. Tao, J. Cheng, M. Song, and X. Lin, "Manifold ranking-based matrix factorization for saliency detection," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published.

[12] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Hoboken, NJ, USA: Wiley, 2001.

[13] Q. Gu, Z. Li, and J. Han, "Generalized Fisher score for feature selection," in *Proc. 27th Conf. Uncertainty Artif. Intell.*, 2011, pp. 266–273.

[14] H. Cheng, Z. Liu, L. Yang, and X. Chen, "Sparse representation and learning in visual recognition: Theory and applications," *Signal Process.*, vol. 93, no. 6, pp. 1408–1425, 2013.

[15] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer, 2010.

[16] Y. Kim and J. Kim, "Gradient Lasso for feature selection," in *Proc. 21st Int. Conf. Mach. Learn.*, 2013, pp. 60–67.

[17] Y. Cong, S. Wang, J. Liu, J. Cao, Y. Yang, and J. Luo, "Deep sparse feature selection for computer aided endoscopy diagnosis," *Pattern Recognit.*, vol. 48, no. 3, pp. 907–917, 2015.

[18] H. Yan and J. Yang, "Sparse discriminative feature selection," *Pattern Recognit.*, vol. 48, no. 5, pp. 1827–1835, 2015.

[19] X. Zhu, H.-I. Suk, and D. Shen, "Matrix-similarity based loss function and feature selection for Alzheimer's disease diagnosis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3089–3096.

[20] S. Xiang, X. Shen, and J. Ye, "Efficient nonconvex sparse group feature selection via continuous and discrete optimization," *Artif. Intell.*, vol. 224, pp. 28–50, Jul. 2015.

[21] J. Liu, J. Ye, and R. Fujimaki, "Forward-backward greedy algorithms for general convex smooth functions over a cardinality constraint," in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. 503–511.

[22] N. Naikal, A. Y. Yang, and S. S. Sastry, "Informative feature selection for object recognition via sparse PCA," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 818–825.

[23] R. Luss and A. d'Aspremont, "Clustering and feature selection using sparse principal component analysis," *Optim. Eng.*, vol. 11, no. 1, pp. 145–157, 2010.

[24] A. Destrero, C. De Mol, F. Odone, and A. Verri, "A regularized framework for feature selection in face detection and authentication," *Int. J. Comput. Vis.*, vol. 83, no. 2, pp. 164–177, 2009.

[25] A. Destrero, C. De Mol, F. Odone, and A. Verri, "A sparsity-enforcing method for learning face features," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 188–201, Jan. 2009.

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

[27] J. Gui, Z. Sun, W. Jia, R. Hu, Y. Lei, and S. Ji, "Discriminant sparse neighborhood preserving embedding for face recognition," *Pattern Recognit.*, vol. 45, no. 8, pp. 2884–2893, 2012.

[28] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Structured sparsity through convex optimization," *Statist. Sci.*, vol. 27, no. 4, pp. 450–468, 2012.

[29] C. Hegde, P. Indyk, and L. Schmidt, "A nearly-linear time framework for graph-structured sparsity," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 928–937.

[30] W. Y. Wang, K. Mazaitis, and W. W. Cohen, "A soft version of predicate invention based on structured sparsity," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3918–3924.

[31] J. Liu, P. Wonka, and J. Ye, "A multi-stage framework for Dantzig selector and Lasso," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 1189–1219, 2012.

[32] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[33] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, Jan. 2012.

[34] I. Rish and G. Grabarnik, *Sparse Modeling: Theory, Algorithms, and Applications*. Boca Raton, FL, USA: CRC Press, 2014.

[35] I. Rish, G. A. Cecchi, A. Lozano, and A. Niculescu-Mizil, *Practical Applications of Sparse Modeling*. Cambridge, MA, USA: MIT Press, 2014.

[36] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: Algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, May 2015.

[37] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. Roy. Statist. Soc. B (Methodol.)*, vol. 58, no. 1, pp. 267–288, 1996.

[38] Y. Li, J. Si, G. Zhou, S. Huang, and S. Chen, "FREL: A stable feature selection algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1388–1402, Jul. 2015.

[39] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ^1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2006.

[40] A. Destrero, C. De Mol, F. Odone, and A. Verri, "A regularized approach to feature selection for face detection," in *Proc. 8th Asian Conf. Comput. Vis.*, 2007, pp. 881–890.

[41] H. Zou, "The adaptive Lasso and its oracle properties," *J. Amer. Statist. Assoc.*, vol. 101, no. 476, pp. 1418–1429, Dec. 2006.

[42] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and smoothness via the fused Lasso," *J. Roy. Statist. Soc. B (Statist. Methodol.)*, vol. 67, no. 1, pp. 91–108, 2005.

[43] Z. Sun, F. Wang, and J. Hu, "LINKAGE: An approach for comprehensive risk prediction for care management," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1145–1154.

[44] S. Petry, C. Flexeder, and G. Tutz, "Pairwise fused Lasso," Dept. Statist., Ludwig Maximilian Univ. Munich, Munich, Germany, Tech. Rep. 102, 2011.

[45] J. Huang, J. L. Horowitz, and S. Ma, "Asymptotic properties of bridge estimators in sparse high-dimensional regression models," *Ann. Statist.*, vol. 36, no. 2, pp. 587–613, 2008.

[46] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Roy. Statist. Soc. B (Statist. Methodol.)*, vol. 67, no. 2, pp. 301–320, 2005.

[47] Z. Liu *et al.*, "Sparse logistic regression with Lp penalty for biomarker identification," *Statist. Appl. Genet. Molecular Biol.*, vol. 6, no. 1, pp. 1–20, 2007.

[48] L. Wang, Z. Sun, and T. Tan, "Robust regularized feature selection for iris recognition via linear programming," in *Proc. 21st Int. Conf. Pattern Recognit.*, Nov. 2012, pp. 3358–3361.

[49] Z. Sun, L. Wang, and T. Tan, "Ordinal feature selection for iris and palmprint recognition," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3922–3934, Sep. 2014.

[50] E. Grave, G. Obozinski, and F. Bach, "Trace Lasso: A trace norm regularization for correlated designs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2187–2195.

[51] J. Ye and J. Liu, "Sparse methods for biomedical data," *ACM SIGKDD Explorations Newslett.*, vol. 14, no. 1, pp. 4–15, 2012.

[52] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Statist. Soc. B (Statist. Methodol.)*, vol. 68, no. 1, pp. 49–67, 2006.

[53] S. Zhang, J. Huang, H. Li, and D. N. Metaxas, "Automatic image annotation and retrieval using group sparsity," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 838–849, Jun. 2012.

[54] R. Jenatton, J.-Y. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," *J. Mach. Learn. Res.*, vol. 12, pp. 2777–2824, Feb. 2011.

[55] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," *J. Mach. Learn. Res.*, vol. 12, pp. 3371–3412, Jan. 2011.

[56] S. Kim and E. P. Xing, "Tree-guided group Lasso for multi-task regression with structured sparsity," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 543–550.

[57] J. Zhou, J. Liu, V. A. Narayan, and J. Ye, "Modeling disease progression via fused sparse group Lasso," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 1095–1103.

[58] D. Kong, R. Fujimaki, J. Liu, F. Nie, and C. Ding, "Exclusive feature learning on arbitrary structures via $\ell_{1,2}$-norm," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1655–1663.

[59] J. Liu and J. Ye, "Moreau–Yosida regularization for grouped tree structure learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1459–1467.

[60] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[61] M. Slawski, W. Zu Castell, and G. Tutz, "Feature selection guided by structural information," *Ann. Appl. Statist.*, vol. 4, no. 2, pp. 1056–1080, 2010.

[62] S. Kim and E. P. Xing, "Statistical estimation of correlated genome associations to a quantitative trait network," *PLoS Genet.*, vol. 5, no. 8, p. e1000587, 2009.

[63] Y. Zhu, X. Shen, and W. Pan, "Simultaneous grouping pursuit and feature selection over an undirected graph," *J. Amer. Statist. Assoc.*, vol. 108, no. 502, pp. 713–725, 2013.

[64] S. Yang, L. Yuan, Y.-C. Lai, X. Shen, P. Wonka, and J. Ye, "Feature grouping and selection over an undirected graph," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 922–930.

[65] X. Wu, K. Yu, W. Ding, H. Wang, and X. Zhu, "Online feature selection with streaming features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 5, pp. 1178–1192, May 2013.

[66] S. Perkins and J. Theiler, "Online feature selection using grafting," in *Proc. 20th Int. Conf. Mach. Learn.*, 2003, pp. 592–599.

[67] C. Ding, D. Zhou, X. He, and H. Zha, "R_1-PCA: Rotational invariant L_1-norm principal component analysis for robust subspace factorization," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 281–288.

[68] A. Argyriou, A. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 41–48.

[69] G. Obozinski, B. Taskar, and M. Jordan, "Multi-task feature selection," Dept. Statist, Univ. California, Berkeley, Berkeley, CA, USA, Tech. Rep. 743, 2006.

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

[71] D. Luo, C. Ding, and H. Huang, "Towards structural sparsity: An explicit ℓ_2/ℓ_0 approach," in *Proc. IEEE 10th Int. Conf. Data Mining*, Dec. 2010, pp. 344–353.

[72] L. Wang, S. Chen, and Y. Wang, "A unified algorithm for mixed $l_{2,p}$-minimizations and its application in feature selection," *Comput. Optim. Appl.*, vol. 58, no. 2, pp. 409–421, 2014.

[73] M. Zhang, C. Ding, Y. Zhang, and F. Nie, "Feature selection at the discrete limit," in *Proc. 38th AAAI Conf. Artif. Intell.*, 2014, pp. 1355–1361.

[74] B. A. Turlach, W. N. Venables, and S. J. Wright, "Simultaneous variable selection," *Technometrics*, vol. 47, no. 3, pp. 349–363, 2005.

[75] J. A. Tropp, "Algorithms for simultaneous sparse approximation. Part II: Convex relaxation," *Signal Process.*, vol. 86, no. 3, pp. 589–602, Mar. 2006.

[76] A. Quattoni, M. Collins, and T. Darrell, "Transfer learning for image classification with sparse prototype representations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[77] M. Schmidt, K. Murphy, G. Fung, and R. Rosales, "Structure learning in random fields for heart motion abnormality detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[78] Y. Jia, M. Salzmann, and T. Darrell, "Factorized latent spaces with structured sparsity," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 982–990.

[79] A. Quattoni, X. Carreras, M. Collins, and T. Darrell, "An efficient projection for $l_{1,\infty}$ regularization," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 857–864.

[80] Y. Zhang and D.-Y. Yeung, "A convex formulation for learning task relationships in multi-task learning," in *Proc. Conf. Uncertainty Artif. Intell.*, 2010, pp. 733–742.

[81] Y. Zhou, R. Jin, and S. C. H. Hoi, "Exclusive Lasso for multi-task feature selection," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 988–995.

[82] Z. Zhao, L. Wang, H. Liu, and J. Ye, "On similarity preserving feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 619–632, Mar. 2013.

[83] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1083–1095, Jun. 2013.

[84] P. Zhu, W. Zuo, L. Zhang, Q. Hu, and S. C. K. Shiu, "Unsupervised feature selection by regularized self-representation," *Pattern Recognit.*, vol. 48, no. 2, pp. 438–446, 2015.

[85] W. Liu, P. P. Pokharel, and J. C. Príncipe, "Correntropy: Properties and applications in non-Gaussian signal processing," *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5286–5298, Nov. 2007.

[86] R. He, T. Tan, L. Wang, and W.-S. Zheng, "$l_{2,1}$ regularized correntropy for robust feature selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2504–2511.

[87] M. Nikolova and M. K. Ng, "Analysis of half-quadratic minimization methods for signal and image recovery," *SIAM J. Sci. Comput.*, vol. 27, no. 3, pp. 937–966, 2005.

[88] 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.

[89] C. Hou, F. Nie, D. Yi, and Y. Wu, "Feature selection via joint embedding learning and sparse regression," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1324–1329.

[90] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, "Joint embedding learning and sparse regression: A framework for unsupervised feature selection," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 793–804, Jun. 2014.

[91] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[92] 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.

[93] Z. Zhao, L. Wang, and H. Liu, "Efficient spectral feature selection with minimum redundancy," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, pp. 673–678.

[94] X. Zhu, X. Wu, W. Ding, and S. Zhang, "Feature selection by joint graph sparse coding," in *Proc. SIAM Int. Conf. Data Mining*, 2013, pp. 803–811.

[95] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 157–165, Feb. 2006.

[96] J. Liu, S. Chen, X. Tan, and D. Zhang, "Comments on 'efficient and robust feature extraction by maximum margin criterion,"' *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1862–1864, Nov. 2007.

[97] S. Yang, C. Hou, F. Nie, and Y. Wu, "Unsupervised maximum margin feature selection via $L_{2,1}$-norm minimization," *Neural Comput. Appl.*, vol. 21, no. 7, pp. 1791–1799, 2012.

[98] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "$l_{2,1}$-norm regularized discriminative feature selection for unsupervised learning," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 1589–1594.

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

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

[101] Q. Gu and J. Han, "Towards feature selection in network," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, 2011, pp. 1175–1184.

[102] F. Chung, *Spectral Graph Theory*, vol. 92. Providence, RI, USA: AMS, 1997.

[103] M. Masaeli, G. Fung, and J. G. Dy, "From transformation-based dimensionality reduction to feature selection," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 751–758.

[104] X. Cai, F. Nie, and H. Huang, "Exact top-k feature selection via $\ell_{2,0}$-norm constraint," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 1240–1246.

[105] D. Hernández-Lobato, J. M. Hernández-Lobato, and Z. Ghahramani, "A probabilistic model for dirty multi-task feature selection," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 1073–1082.

[106] D. Han and J. Kim, "Unsupervised simultaneous orthogonal basis clustering feature selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5016–5023.

[107] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient $l_{2,1}$-norm minimization," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 339–348.

[108] H. Wang *et al.*, "Sparse multi-task regression and feature selection to identify brain imaging predictors for memory performance," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 557–562.

[109] S. Lee, J. Zhu, and E. P. Xing, "Adaptive multi-task Lasso: With application to eQTL detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1306–1314.

[110] J. Tang and H. Liu, "Feature selection with linked data in social media," in *Proc. SIAM Int. Conf. Data Mining*, 2012, pp. 118–128.

[111] Q. Gu, Z. Li, and J. Han, "Linear discriminant dimensionality reduction," in *Proc. Eur. Conf. Mach. Learn.*, 2011, pp. 549–564.

[112] G. Obozinski, B. Taskar, and M. I. Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems," *Statist. Comput.*, vol. 20, no. 2, pp. 231–252, Apr. 2010.

[113] Y. Liang, S. Liao, L. Wang, and B. Zou, "Exploring regularized feature selection for person specific face verification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1676–1683.

[114] T. Jebara, "Multitask sparsity via maximum entropy discrimination," *J. Mach. Learn. Res.*, vol. 12, no. 1, pp. 75–110, 2011.

[115] X. Chang, F. Nie, Y. Yang, and H. Huang, "A convex formulation for semi-supervised multi-label feature selection," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1171–1177.

[116] Y. Guo and W. Xue, "Probabilistic multi-label classification with sparse feature learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 1373–1379.

[117] L.-L. Huang, J. Tang, S.-B. Chen, C. Ding, and B. Luo, "An efficient algorithm for feature selection with feature correlation," in *Proc. Intell. Sci. Intell. Data Eng.*, 2013, pp. 639–646.

[118] S. Ji, L. Tang, S. Yu, and J. Ye, "A shared-subspace learning framework for multi-label classification," *ACM Trans. Knowl. Discovery Data*, vol. 4, no. 2, pp. 1–29, 2010.

[119] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, and N. Sebe, "Web image annotation via subspace-sparsity collaborated feature selection," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1021–1030, Aug. 2012.

[120] C. Xu, D. Tao, and C. Xu, "Multi-view intact space learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 12, pp. 2531–2544, Dec. 2015.

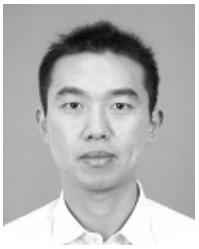
[121] L. Xiao, Z. Sun, R. He, and T. Tan, "Coupled feature selection for cross-sensor iris recognition," in *Proc. IEEE 6th Int. Conf. Biometrics, Theory, Appl. Syst.*, Sep./Oct. 2013, pp. 1–6.

[122] L. Xiao, Z. Sun, R. He, and T. Tan, "Margin based feature selection for cross-sensor iris recognition via linear programming," in *Proc. 2nd IAPR Asia Conf. Pattern Recognit.*, 2013, pp. 246–250.

[123] H. Wang, F. Nie, H. Huang, and C. Ding, "Heterogeneous visual features fusion via sparse multimodal machine," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3097–3102.

[124] H. Wang, F. Nie, and H. Huang, "Multi-view clustering and feature learning via structured sparsity," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 352–360.

[125] H. Wang *et al.*, "Identifying quantitative trait loci via group-sparse multitask regression and feature selection: An imaging genetics study of the ADNI cohort," *Bioinformatics*, vol. 28, no. 2, pp. 229–237, 2012.
[126] H. Wang *et al.*, "Identifying disease sensitive and quantitative trait-relevant biomarkers from multidimensional heterogeneous imaging genetics data via sparse multimodal multitask learning," *Bioinformatics*, vol. 28, no. 12, pp. i127–i136, 2012.
[127] J. Gui, D. Tao, Z. Sun, Y. Luo, X. You, and Y. Y. Tang, "Group sparse multiview patch alignment framework with view consistency for image classification," *IEEE Trans. Image Process.*, vol. 23, no. 7, pp. 3126–3137, Jul. 2014.
[128] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 447–461, Mar. 2016.
[129] P. S. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 82–90.
[130] L. Laporte, R. Flamary, S. Canu, S. Déjean, and J. Mothe, "Nonconvex regularizations for feature selection in ranking with sparse SVM," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1118–1130, Jun. 2014.
[131] L. Wang, J. Zhu, and H. Zou, "Hybrid huberized support vector machines for microarray classification and gene selection," *Bioinformatics*, vol. 24, no. 3, pp. 412–419, 2008.
[132] M. Tan, L. Wang, and I. W. Tsang, "Learning sparse SVM for feature selection on very high dimensional datasets," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 1047–1054.
[133] G. Fung and O. L. Mangasarian, "Data selection for support vector machine classifiers," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2000, pp. 64–70.
[134] B.-Y. Sun, Z.-H. Zhu, J. Li, and B. Linghu, "Combined feature selection and cancer prognosis using support vector machine regression," *IEEE-ACM Trans. Comput. Biol. Bioinformatics*, vol. 8, no. 6, pp. 1671–1677, Nov./Dec. 2011.
[135] Z. Liu, S. Lin, and M. T. Tan, "Sparse support vector machines with L_p penalty for biomarker identification," *IEEE-ACM Trans. Comput. Biol. Bioinformatics*, vol. 7, no. 1, pp. 100–107, Jan./Mar. 2010.
[136] G. C. Cawley, N. L. C. Talbot, and M. Girolami, "Sparse multinomial logistic regression via Bayesian L1 regularisation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 209–216.
[137] S. K. Shevade and S. S. Keerthi, "A simple and efficient algorithm for gene selection using sparse logistic regression," *Bioinformatics*, vol. 19, no. 17, pp. 2246–2253, 2003.
[138] G. C. Cawley and N. L. C. Talbot, "Gene selection in cancer classification using sparse logistic regression with Bayesian regularization," *Bioinformatics*, vol. 22, no. 19, pp. 2348–2355, 2006.
[139] Y. Qian, J. Zhou, M. Ye, and Q. Wang, "Structured sparse model based feature selection and classification for hyperspectral imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2011, pp. 1771–1774.
[140] X. Cai, F. Nie, H. Huang, and C. Ding, "Multi-class $\ell_{2,1}$-norm support vector machine," in *Proc. IEEE 11th Int. Conf. Data Mining*, Dec. 2011, pp. 91–100.
[141] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Proc. AAAI Conf. Artif. Intell.*, 2012, pp. 1026–1032.
[142] Y. Yang, H. T. Shen, F. Nie, R. Ji, and X. Zhou, "Nonnegative spectral clustering with discriminative regularization," in *Proc. AAAI Conf. Artif. Intell.*, 2011, pp. 555–560.
[143] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[144] S. X. Yu and J. Shi, "Multiclass spectral clustering," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 313–319.
[145] D. M. Witten and R. Tibshirani, "A framework for feature selection in clustering," *J. Amer. Statistical Assoc.*, vol. 105, no. 490, pp. 713–726, 2010.
[146] H. Zeng and Y.-M. Cheung, "Feature selection and kernel learning for local learning-based clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1532–1547, Aug. 2011.
[147] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 507–514.
[148] J. Xu, G. Yang, H. Man, and H. He, "L_1 graph based on sparse coding for feature selection," in *Proc. 10th Int. Symp. Neural Netw.*, 2013, pp. 594–601.
[149] A. Y. Ng, "Feature selection, L_1 vs. L_2 regularization, and rotational invariance," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, pp. 78–85.
[150] Y. Sun, S. Todorovic, and S. Goodison, "Local-learning-based feature selection for high-dimensional data analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1610–1626, Sep. 2010.
[151] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *J. Mach. Learn. Res.*, vol. 6, pp. 1855–1887, Jan. 2005.
[152] Q. Mao and I. W.-H. Tsang, "A feature selection method for multivariate performance measures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2051–2063, Sep. 2013.
[153] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Dept. Comput. Sci., Columbia Univ., New York, NY, USA, Tech. Rep. CUCS-005-96, 1996.
[154] C. L. Nutt *et al.*, "Gene expression-based classification of malignant gliomas correlates better with survival than histological classification," *Cancer Res.*, vol. 63, no. 7, pp. 1602–1607, 2003.
[155] A. Bhattacharjee *et al.*, "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses," *Proc. Nat. Acad. Sci. USA*, vol. 98, no. 24, pp. 13790–13795, 2001.
[156] S. A. Armstrong *et al.*, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genet.*, vol. 30, no. 1, pp. 41–47, 2002.
[157] L. Sun *et al.*, "Neuronal and glioma-derived stem cell factor induces angiogenesis within the brain," *Cancer Cell*, vol. 9, no. 4, pp. 287–300, 2006.
[158] A. Martínez and R. Benavente, "The AR face database," Dept. Elect. Comput. Eng., Ohio State Univ., Columbus, OH USA, Tech. Rep. 24, Jun. 1998.
[159] L. Čehovin and Z. Bosnić, "Empirical evaluation of feature selection methods in classification," *Intell. Data Anal.*, vol. 14, no. 3, pp. 265–281, 2010.
[160] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York, NY, USA: Wiley, 2006.
[161] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
[162] I. Kononenko, "Estimating attributes: Analysis and extensions of RELIEF," in *Proc. Eur. Conf. Mach. Learn.*, 1994, pp. 171–182.
[163] D. C. Montgomery, G. C. Runger, and N. F. Hubele, *Engineering Statistics*. New York, NY, USA: Wiley, 2009.
[164] L. Deng, J. Pei, J. Ma, and D. L. Lee, "A rank sum test method for informative gene discovery," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 410–419.
[165] D. Tao, X. Lin, L. Jin, and X. Li, "Principal component 2-D long short-term memory for font recognition on single Chinese characters," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 756–765, Mar. 2016.
[166] E. P. Xing, M. I. Jordan, and R. M. Karp, "Feature selection for high-dimensional genomic microarray data," in *Proc. 18th Int. Conf. Mach. Learn.*, 2001, pp. 601–608.
[167] W. Cochran, *Sampling Techniques*. Hoboken, NJ, USA: Wiley, 2007.
[168] B. Gu, F. Hu, and H. Liu, "Sampling: Knowing whole from its part," in *Instance Selection and Construction for Data Mining*. Boston, MA, USA: Kluwer Academic Publishers, 2001, pp. 21–38.
[169] H. Liu, H. Motoda, and L. Yu, "Feature selection with selective sampling," in *Proc. Int. Conf. Mach. Learn.*, 2002, pp. 395–402.



Jie Gui (M'12) received the B.S. degree in computer science from Hohai University, Nanjing, China, in 2004, the M.S. degree in computer applied technology from the Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, China, in 2007, and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, in 2010.

He is currently an Associate Professor with the Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei. His current research interests include machine learning, pattern recognition, data mining, and image processing.

Zhenan Sun (M'07) received the B.S. degree in industrial automation from the Dalian University of Technology, Dalian, China, in 1999, the M.S. degree in system engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2002, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, in 2006.

He has been a Faculty Member with the National Laboratory of Pattern Recognition (NLPR), CASIA, since 2006, where he is currently a Professor with NLPR. He has authored or co-authored over 100 technical papers. His current research interests include biometrics, pattern recognition, and computer vision.

Dr. Sun is a member of the IEEE Computer Society and the IEEE Signal Processing Society. He is also an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY and the IEEE BIOMETRICS COMPENDIUM.

Shuiwang Ji (SM'15) received the Ph.D. degree in computer science from Arizona State University, Tempe, AZ, USA, in 2010.

He is currently an Associate Professor with the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA, USA. His current research interests include machine learning, data mining, computational neuroscience, and bioinformatics.

Dr. Ji is an Editorial Board Member of *Data Mining and Knowledge Discovery*. He received the National Science Foundation CAREER Award in 2014. He is also an Associate Editor of *BMC Bioinformatics* and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Dacheng Tao (F'15) is currently a Professor of Computer Science with the Centre for Quantum Computation and Intelligent Systems, and the Faculty of Engineering and Information Technology, University of Technology Sydney (UTS), Sydney, NSW, Australia. He mainly applies statistics and mathematics to data analytics problems. His current research interests include computer vision, data science, image processing, machine learning, and video surveillance. His research results have expounded in one monograph and over 200 publications at prestigious journals and prominent conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the *Journal of Machine Learning Research*, the *International Journal of Computer Vision*, the Conference on Neural Information Processing Systems, the International Conference on Machine Learning, Computer Vision and Pattern Recognition, the International Conference on Computer Vision, the European Conference on Computer Vision, the International Conference on Artificial Intelligence and Statistics, the International Conference on Data Mining series, and Association for Computing Machinery's Special Interest Group on Knowledge Discovery and Data Mining.

Mr. Tao is a fellow of Optical Society of America, International Association for Pattern Recognition, and SPIE. He received several best paper awards, such as the Best Theory/Algorithm Paper Runner Up Award in the IEEE International Conference on Data Mining (ICDM)'07, the Best Student Paper Award in the IEEE ICDM'13, and the 2014 ICDM 10-Year Highest-Impact Paper Award. He also received the 2015 Australian Scopus-Eureka Prize, the 2015 ACS Gold Disruptor Award, and the 2015 UTS Vice-Chancellors Medal for Exceptional Research.

Tieniu Tan (F'03) received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, in 1984, and the M.S. and Ph.D. degrees from Imperial College London, London, U.K., in 1986 and 1989, respectively, all in electronics engineering.

He joined the Department of Computer Science, University of Reading, Reading, U.K., in 1989, where he was a Research Fellow, Senior Research Fellow, and Lecturer. In 1998, he returned to China to join the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CAS), Beijing, China, as a Full Professor, where he was the Director General of the Institute of Automation from 2000 to 2007, and the Director of NLPR from 1998 to 2013. He is currently the Director of the Center for Research on Intelligent Perception and Computing with the Institute of Automation and also serves as the Deputy President of CAS. He has authored over 500 research papers in refereed international journals and conferences in the areas of image processing, computer vision, and pattern recognition, and has authored and edited 11 books. He holds more than 70 patents. His current research interests include biometrics, image and video understanding, and information forensics and security.

Dr. Tan is a member (Academician) of CAS, a fellow of The World Academy of Sciences for the advancement of sciences in developing countries and the International Association of Pattern Recognition, and an International Fellow of the U.K. Royal Academy of Engineering. He has given invited talks and keynotes at many universities and international conferences, and received numerous national and international awards and recognitions. He is the Editor-in-Chief of the *International Journal of Automation and Computing*.