Semi-supervised Feature Selection via Insensitive Sparse Regression with Application to Video Semantic Recognition

Tingjin Luo, Chenping Hou, *Member, IEEE,* Feiping Nie, Hong Tao, and Dongyun Yi

Abstract—Feature selection plays a significant role in dealing with high-dimensional data to avoid the curse of dimensionality. In many real applications, like video semantic recognition, handling few labeled and large unlabeled data samples from the same population is a recently addressed challenge in feature selection. To solve this problem, we propose a novel semi-supervised feature selection method via insensitive sparse regression (ISR). Specifically, we compute the soft label matrix by the special label propagation, which can predict the labels of the unlabeled data. To guarantee the robustness of ISR to the false labeled instances or outliers, we propose Insensitive Regression Model (IRM) by capped l_2-l_p-norm loss. The soft label is imposed as the weights of IRM to fully utilize the label information. Meanwhile, to perform feature selection, we incorporate $l_{2,q}$-norm regularizer with IRM as the structural sparsity constraint when $0 < q \leq 1$. Moreover, we put forward an effective approach for solving the formulated non-convex optimization problem. We analyze the performance of convergence rigorously and discuss the parameter determination problem. Extensive experimental results on several public data sets verify the effectiveness of our proposed algorithm in comparison with the state-of-art feature selection methods. Finally, we apply our method to video semantic recognition successfully.

Index Terms—Dimensionality reduction, Semi-supervised feature selection, Video semantic recognition, Insensitive sparse regression, Capped l_2-l_p-norm loss.

1 Introduction

In recent years, the dimensionality of data becomes larger and larger—even more than millions of features with the innovations of information techniques, while many of these features may be irrelevant in many real applications, e.g., face recognition [1], [2], [3], [4],object recognition [5], [6], [7], video semantic recognition [8], [9], [10] and bioinformatics [11], [12], [13], [14]. The availability of massive and high-dimensional data along with new scientific tasks have reshaped statistical theory and data analysis. Directly processing such data not only degrade its performance but also is time-consuming. To solve this problem, many feature selection and feature extraction techniques have been introduced for dimensionality reduction. Feature extraction usually maps the original features into a lower dimensional space [2], [4], [6], [7], [15], [16], [17], [18]. Compared with feature extraction, feature selection identifies the optimal subset of the original features. By maintaining the original features, feature selection improves the interpretability of the data, which is preferred in many real applications, such as genetic analysis [12], [13], [19] and video semantic recognition [8], [20]. In addition, feature selection allows us to just focus on the feature subset that we are concerned about, but not the whole set. We focus on feature selection in this paper for its superiority over feature extraction.

According to the availability of labels, feature selection algorithms can be roughly classified into three groups: unsupervised [3], [21], [22], [23], semi-supervised [19], [24], [25], [26] and supervised algorithms [12], [13], [27], [28]. Without labels, unsupervised feature selection evaluates feature relevance by exploiting data variance and separability, i.e. Laplacian Score (LapScor) [29], Multi-Cluster Feature Selection (MCFS) [30] and Infinite Feature Selection(InfFS) [22]. Because discriminative information is enclosed in the class labels, supervised feature selection often selects discriminative features by evaluating features correlation, i.e. Fisher Score [31], Robust Feature Selection (RFS) [13], Spectral feature selection (SPEC) [32] and Feature Selection via Eigenvector Centrality(ECFS) [28], [33]. For the unsupervised methods, it is very difficult to select the discriminative features without available labels. For supervised methods, sufficient labeled training data are required to guarantee high accuracy and reliable performance. Nevertheless, it is very expensive and time-consuming to label training data in real-word applications, especially for video semantic recognition. Supervised algorithms could fail to identify the discriminative features in these cases. This motivates many researchers to develop semi-supervised methods to select the most discriminative features.

Semi-supervised feature selection belongs to the area of Semi-Supervised Learning (SSL) [34], which is an effective way of processing both labeled and unlabeled data. Inspired by SSL, semi-supervised feature selection algorithms use

- *Tingjin Luo, Chenping Hou, Hong Tao and Dongyun Yi are with College of Science, National University of Defense Technology, Changsha, Hunan, 410073, China. E-mail: luotingjin03@nudt.edu.cn, hcpnudt@hotmail.com, taohong.nudt@gmail.com, dongyun.yi@gmail.com.*
- *Feiping Nie is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xian 710072, Shaanxi, China. (E-mail: feipingnie@gmail.com).*

This work was supported by the National Science Foundation of China (61473302 and 61503396). The first two authors contribute equally and Chenping Hou is the corresponding author.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2018.2810286, IEEE Transactions on Knowledge and Data Engineering

the local structure of both labeled and unlabeled data to select the discriminative features. For example, He et al. [35] proposed Locality Sensitive Discriminant Feature by constructing two graphs, i.e. within-class graph and between-class graph, to discover both geometrical and discriminant structures. Similarly, Xu et al. [25] adopted manifold regularization to consider the geometry of data distribution, and then proposed a new Discriminative Semi-Supervised feature selection via Manifold Regularization [25] which selects features through maximizing the classification margin between different classes. After analyzing the popular trace ratio criterion in dimensionality reduction, Nie et al. [19] proposed an efficient semi-supervised Feature Selection algorithm with Trace Ratio Criterion (TRCFS).

However, in practice, false labeled samples often exist in the real datasets. For example, in image processing, many datasets are obtained from Internet and annotated by non-professional technician, such as website image annotation, which will generate the inaccurate label information. In data mining, many spam detection datasets are collected from different sources, such as Email server, Website and social networks etc. Different persons annotate the data with different background knowledge and personality, which will lead to mismatching of labels. When the false labeled instances or outliers exist in the training data, traditional semi-supervised feature selection methods robustness performance can be improved.

Besides, semi-supervised feature selection has been widely investigated due to its importance in real applications. In this paper, we focus on video semantic recognition, which plays a crucial role in multimedia analysis. Obviously, video is a type of high-dimensional data. In video semantic recognition, the number of labeled videos is always very small while a large amount of video clips are unlabeled. In this case, Ma et al. [20] proposed a Structural Feature Selection with Sparsity (SFSS), which can jointly select the most relevant features by manifold learning and $l_{2,1}$-norm regularization. To use the multi-label information of videos, Ma et al. [36] proposed subspace feature uncovering with sparsity (SUFS), which uncovers a feature subspace shared among multiple different classes. By integrating the polynomials and Greens functions into the spline basis functions, Han et al. [8] proposed Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R). Although these methods are successfully applied to multimedia data understanding, such as image annotation and video concept detection, their performance can still be improved in real applications.

To select a subset of more discriminative features from few labeled and large unlabeled high-dimensional data, we propose a novel semi-supervised feature selection algorithm via Insensitive Sparse Regression (ISR). To enlarge label information of training data, label propagation is adapted to calculate the soft labels of both labeled and unlabeled data by using the local structural information of high dimensional data. To improve the robustness to the false labeled data or outliers, we propose Insensitive Regression Model (IRM) by adding the soft label information into capped l_2-l_p-norm loss function. And then we incorporate IRM and $l_{2,q}$-norm regularizer into the formulation of ISR to perform feature selection task. Although this formulation is a non-convex problem with respect to two groups of parameters, we propose an efficient method to solve it. We analyze the performance of ISR in aspects of the convergence behavior and parameter determination. Extensive experimental results on different kinds of data sets verify the performance of ISR simultaneously. Finally, we provide the main procedure of ISR for multimedia analysis and apply it to video semantic recognition. The main contributions of this paper are summarized as:

(1) We present Insensitive Regression Model (IRM) to maximize the use of the enlarged label information obtained by label propagation.

(2) We propose a novel semi-supervised feature selection approach based on Insensitive Sparse Regression (ISR). It incorporates capped l_2-l_p-norm loss function with $l_{2,p}$-norm regularization, which can improve the robustness performance to the noise and false labels.

(3) We present an efficient method to solve our proposed non-convex formulation and analyze the performance of ISR. We also verify the effectiveness of our method according to extensive experimental results on several data sets.

(4) We apply ISR into video semantic recognition applications and demonstrates its promising performance.

The rest of this paper is organized as follows. Section II provides notations and related work. We formulate the proposed ISR algorithm and provide an effective solution to this problem in Section III. In Section IV, We analyze the convergence behavior, parameter determination, and sensitivity to false labels of ISR exhaustively. Section V provides promising comparison results on various kinds of data sets. In Section VI, we illustrate the main process of ISR for multimedia and apply ISR into video semantic recognition. Finally, Section VII presents the conclusion.

2 Related Work

2.1 Notations and Definitions

In this paper, matrices and vectors are written as boldface uppercase letters and boldface lowercase letters, respectively. For instance, a matrix $\mathbf{W} \in \mathbb{R}^{s\times t}$, its i-th row, j-th column are denoted by $\mathbf{w}^i$, $\mathbf{w}_j$ respectively. For semi-supervised feature selection, the training set $\mathbf{X} \in \mathbb{R}^{d\times n}$ often consists of two parts: the labeled data $\mathbf{X}_L$ and the unlabeled data $\mathbf{X}_U$. Without loss of generality, let the first $n_l (n_l \leq n)$ samples in $\mathbf{X}$ is the labeled data $\mathbf{X}_L$ and its corresponding labels $\mathbf{Y}_L = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{n_l}] \in \{0,1\}^{C\times n_l}$ are provided for the C semantic categories. Then we have $\mathbf{X} = [\mathbf{X}_L, \mathbf{X}_U]$ and $n = n_l + n_u$. Define the initial label matrix $\mathbf{Y} = [\mathbf{Y}_1, \dots, \mathbf{Y}_n]^T \in \mathbb{R}^{n\times(C+1)}$, where $\mathbf{Y}_i \in \mathbb{R}^{C+1} (i = 1 \dots n)$ is the initial label of the i-th data. Let $\mathbf{F}_i \in \mathbb{R}^{C+1} (i = 1 \dots n)$ is the learned soft label of the i-th data point where $0 \leq F_{ij} \leq 1, j = 1 \dots C+1$. For the labeled data, if $\mathbf{x}_i$ is labeled the j-th class then $Y_{ij} = 1$ and 0 otherwise. For the unlabeled data point $\mathbf{x}_i$, $Y_{ij} = 1$ when $j = C+1$ and 0 otherwise. Additionally, we define $\mathbf{F} = [\mathbf{F}_1, \dots, \mathbf{F}_n]^T \in \mathbb{R}^{n\times(C+1)}$ is the predicted soft label matrix, $\mathbf{I}$ as an identity matrix, $\mathbf{1}_n = [1, \dots, 1]^T \in \mathbb{R}^{n\times 1}$ and $\mathbf{0}_n = [0, \dots, 0]^T \in \mathbb{R}^{n\times 1}$.

2.2 Label Propagation

Label propagation is one of the most important methods in Semi-Supervised Learning (SSL) [34]. The basic idea is to propagate label information from labeled data to unlabeled data according to the training data distribution [37]. In the literature, many traditional label propagation methods, such as Gaussian Field and Harmonic Function (GFHF) [38] and Local and Global Consistency (LGC) [39], are proposed to solve this problem. Based on the spectral graph theory, LGC and GFHF are unified into the following framework:

$$\mathbf{F}^* = \arg\min \|\mathbf{F}\|_{\mathcal{G}}^2 + \mu\|\mathbf{F}-\mathbf{Y}\|_F^2, \tag{1}$$

where $\|\mathbf{F}\|_{\mathcal{G}}^2$ represents global smoothness over graph $\mathcal{G}$ and $\|\mathbf{F}-\mathbf{Y}\|_F^2$ measures the empirical loss for labeled samples. μ balances global smoothness and empirical loss terms. Specifically, in LGC, the global smoothness function is defined by the semi-inner product $\|\mathbf{F}\|_{\mathcal{G}}^2 = \langle\mathbf{F},\mathbf{\Delta}\mathbf{F}\rangle = Tr(\mathbf{F}^T\mathbf{\Delta}\mathbf{F})$, where $\mathbf{\Delta}$ is the normalized graph Laplacian. If we set $\mu = \infty$ and use standard graph Laplacian quantity $\mathbf{L}$ for the smoothness term, i.e. $\|\mathbf{F}\|_{\mathcal{G}}^2 = Tr(\mathbf{F}^T\mathbf{L}\mathbf{F})$, the above framework reduces to the formulation of GFHF.

To improve the robustness to outliers, Nie et al. [37] proposed GGSSL based on random walks. In each random walks step, the label information of each data point includes two parts of information from its neighbors and its initial label. Thus the label information of data at $t+1$ iteration is propagated as

$$\mathbf{F}(t+1) = \mathbf{I}_\alpha\mathbf{P}\mathbf{F}(t) + \mathbf{I}_\beta\mathbf{Y}, \tag{2}$$

where, $\mathbf{I}_\alpha$ is a diagonal matrix with the i-th entry $\mathbf{I}_{\alpha ii} = \alpha$, $\mathbf{I}_\beta = \mathbf{I}-\mathbf{I}_\alpha$. α balances the information of $\mathbf{x}_i$ from the initial label and its neighbors. $\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$ is a stochastic matrix, where $\mathbf{A}$ is the weight matrix of $\mathbf{X}$ and $\mathbf{D}$ is a diagonal matrix with the i-th element $D_{ii} = \sum_j A_{ij}$. Finally, Eq. (2) will converge to the following equation.

$$\mathbf{F} = \lim_{t\to\infty}\mathbf{F}(t) = (\mathbf{I}-\mathbf{I}_\alpha\mathbf{P})^{-1}\mathbf{I}_\beta\mathbf{Y}. \tag{3}$$

LGC and GFHF can also be interpreted as random walks, but the transition probability matrix and the stop condition are different. In LGC and GFHF, the walks can only stop at the labeled data points, while in GGSSL, the random walks can stop at the labeled or unlabeled data points. To some extent, LGC and GFHF are two special cases of GGSSL.

2.3 Representative Feature Selection Methods

2.3.1 Unsupervised Feature Selection Methods

(1) LapScor is one of the famous unsupervised feature selection approaches. The basic idea of LapScor is to evaluate the features according to their local preserving power. First, we construct a kNN graph $\mathcal{G}$ with n nodes and compute a weight matrix $\mathbf{A}$ on $\mathcal{G}$. Then, the Laplacian score for the i-th feature is computed as

$$a_i = \hat{\mathbf{f}}_i^T\mathbf{L}\hat{\mathbf{f}}_i/\hat{\mathbf{f}}_i^T\mathbf{D}\hat{\mathbf{f}}_i, \tag{4}$$

where $\mathbf{D}$ is a diagonal matrix whose element $D_{ii} = \sum_{j=1}^n A_{ij}$, $\mathbf{L} = \mathbf{D}-\mathbf{A}$ and $\hat{\mathbf{f}}_i = \mathbf{f}_i - \mathbf{f}_i^T\mathbf{D}\mathbf{1}\mathbf{1}/\mathbf{1}^T\mathbf{D}\mathbf{1}$.Finally, we rank the features based on $\{a_i\}_{i=1}^d$ and choose the features corresponding to the s largest value a_i.

(2) MCFS computes the embedding by Laplacian Eigenmap(LE) [16] at first and then use regression coefficient to rank each feature. The formulation of MCFS is

$$\min_{\mathbf{w}_i\in\mathbb{R}^d}\left\|\hat{\mathbf{y}}_i - \mathbf{X}^T\mathbf{w}_i\right\|^2 + \beta\|\mathbf{w}_i\|_1, \tag{5}$$

where $\hat{\mathbf{y}}_i$ is the i-th feature of new low-dimensional representation. Clearly, the problem (5) is known as Lasso, which has been widely investigated in literature. After computing the regression coefficient, they define the MCFS score for the j-th feature as $v_j = \max_i|\mathbf{w}_{i,j}|$. Similar with LapScor, we rank and select the features based on the values $\{v_j\}_{j=1}^d$.

(3) InfFS [22] is the recent unsupervised graph-based filter method. InfFS builds a graph whose node is each feature. In the graph, a path is a selection of features. Thus, it assigns a score of "importance" to each feature by taking into account all the possible feature subsets as paths on a graph, that is

$$A_{ij} = \alpha\sigma_{ij} + (1-\alpha)c_{ij}, \tag{6}$$

where $\alpha\in[0,1]$, $\sigma_{ij} = \max(\sigma_i,\sigma_j)$ with σ_i being the standard deviation of i-th feature f_i and $c_{ij} = 1 - |spearman(f_i,f_j)|$ with $spearman(\cdot)$ indicating Spearman's rank correlation coefficient. The final energy scores for each feature are obtained by

$$\mathbf{s} = \mathbf{S}\mathbf{1}, \tag{7}$$

where $\mathbf{S} = (I-r\mathbf{A})^{-1} - I$ encodes all the energy information of features. By ranking the energy scores, we can obtain a rank for the feature to be selected.

2.3.2 Supervised Feature Selection Methods

(1) Fisher score(FisherScor) [31] is a supervised feature selection method. It selects the features such that the feature values of samples within the same class are small while the feature values of samples from different classes are large. Fisher score of each feature f_i is evaluated as:

$$fishers(f_i) = \sum_{j=1}^c n_j(\mu_{ij}-\mu_i)^2/\sum_{j=1}^c n_j\sigma_{ij}^2, \tag{8}$$

where n_j, μ_i, μ_{ij} and σ_{ij}^2 are the number of instances in j-th class, mean value of f_i, mean value of f_i for samples in j-th class and variance of f_i for samples in j-th class, respectively. Similar to LapScor, the top k features is greedily selected with the largest Fisher scores.

(2) RFS is another supervised feature selection. In practice, the noisy instances or outliers are common in real world applications. To solve this problem, Nie et al. [13] proposed a robust feature selection (RFS) methods to employ joint $\ell_{2,1}$-norm minimization on both loss and regularization. The objective function of RFS is

$$\min_{\mathbf{W}}\left\|\mathbf{W}^T\mathbf{X}-\mathbf{Y}\right\|_{2,1} + \lambda\|\mathbf{W}\|_{2,1}. \tag{9}$$

(3) ECFS [28], [33] is one of graph-based supervised methods and obtains the important nodes through the Eigenvector Centrality(EC). Similar with InfFS, ECFS ranks features by identifying the most important nodes on an affinity graph where features are the nodes. Meanwhile, the adjacency matrix $\mathbf{A}$ of the graph is given by

$$\mathbf{A} = \alpha\mathbf{K} + (1-\alpha)\Sigma, \tag{10}$$

where $\Sigma_{ij} = \max(\sigma_i,\sigma_j)$ and $\mathbf{K}$ is a kernel matrix. $\mathbf{K} = \mathbf{k}\cdot\mathbf{m}^T$ with k_i is Fisher information and m_i is the mutual information between the ranked features and the features truely related to their classes. By the graph theory, the feature scores are obtained by computing the principle eigenvector of $\mathbf{A}$.

2.3.3 *Semi-supervised Feature Selection Methods*

(1) TRCFS [19] is an efficient semi-supervised feature selection algorithm to select relevant features. To solve the problem that trace ratio criterion in dimensionality reduction tends to select features with very small variance, TRCFS adopts the trace ratio criterion for feature selection with a rescale preprocessing. The objective function of TRCFS can be formulated as

$$\arg\max_{\mathbf{W}} tr(\mathbf{W}^T\widetilde{\mathbf{S}}_b\mathbf{W})/tr(\mathbf{W}^T\widetilde{\mathbf{S}}_w\mathbf{W}), \tag{11}$$

where two scatter matrices $\tilde{\mathbf{S}}_w$ and $\tilde{\mathbf{S}}_b$ based on soft labels matrix $\mathbf{F}$ are computed by

$$\tilde{\mathbf{S}}_w = \frac{1}{n}\mathbf{X}(\mathbf{B} - \mathbf{F}_c\mathbf{D}\mathbf{F})\mathbf{X}^T \tag{12}$$

$$\tilde{\mathbf{S}}_b = \frac{1}{n}\mathbf{X}(\mathbf{F}_c\mathbf{D}\mathbf{F}_c^T - \frac{1}{n}\mathbf{B}\mathbf{1}\mathbf{1}^T\mathbf{B}^T)\mathbf{X}^T, \tag{13}$$

where $\mathbf{F}_c$ is formed by the first C columns of $\mathbf{F}$, $\mathbf{B}$ and $\mathbf{D}$ are diagonal matrix, the i-th diagonal elements are $B_{ii} = \sum_j F_{ij}$ and $D_{ii} = 1/\sum_j F_{ji}$, respectively.

(2) S^{2}FS^{2}R [8] is another famous semi-supervised feature selection approach. It exploits the local geometry underlying the huge amount of unlabeled data by splines developed in Sobolev space. Integrating the polynomials and Greens functions into the local spline, it preserves data distribution and label information by combining within-class and spline scatters matrix. Based on the idea of graph embedding, S^{2}FS^{2}R computes the optimal $\mathbf{W}$ by adding $l_{2,1}$-norm regularization with the orthogonal constraint:

$$arg \min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} tr(\mathbf{W}^T\mathbf{M}\mathbf{W}) + \lambda\|\mathbf{W}\|_{2,1}, \tag{14}$$

where $\mathbf{M} = \mathbf{S}_W + \mu\boldsymbol{\Xi}$, $\mathbf{S}_W$ is the within-class scatter matrix and $\boldsymbol{\Xi}$ represents the spline scatter matrix. The within-class scatter matrix $\mathbf{S}_W$ is estimated by Eq.(15).

$$\mathbf{S}_W = \sum_{j=1}^{C}\frac{1}{N_j}\sum_{\mathbf{x}\in\varpi_j}(\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T, \tag{15}$$

where $\mathbf{m}_j$ is the average vector of the j-th class. However, when false labeled instances or outliers exist in the training data, the performance of S^{2}FS^{2}R will be degraded.

3 Semi-supervised Feature Selection via Insensitive Sparse Regression

In this section, we first introduce the formulation of our method formally. Then we propose a two-step strategy to solve the proposed problem. For convenience, we refer our proposed formulation as semi-supervised feature selection via Insensitive Sparse Regression (ISR).

3.1 Soft Labels Computation via Label Propagation

Inspired by Semi-Supervised Learning (SSL) [19], we propose to use a special label propagation method via linear regression based on manifold regularization, which can detects outliers and false labeled samples in data effectively. A crucial component of a graph based SSL method is the estimation of a weighted graph from the training data $\mathbf{X}$. There are many methods to construct the weight matrix $\mathbf{A}$ of a graph in the literature, such as binary weighting, Gaussian kernel and LLE weighting. In this paper, we build the weight matrix $\mathbf{A}$ of the graph by Gaussian kernel in Eq. (16), where the hyper parameter σ in the Gaussian function is automatically determined by $\mathbf{X}$ [40],

$$A_{ij} = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma^2}, \tag{16}$$

where $\sigma = \tau\sqrt{\bar{d}/\ln(n)}$, $\tau = 0.3$, $\bar{d}$ is the average of squared Euclidean distances for all pairs of data samples and n is the number of training data.

Similar to GFHF [34] and LGC [39], the soft labels for the unlabeled data can be computed by imposing label fitness and manifold smoothness constraints. Therefore, the objective function of our label propagation can be unified as the following weighted linear regression model:

$$\mathbf{F} = \arg\min_{\mathbf{F}} tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + tr[(\mathbf{F} - \mathbf{Y})^T\mathbf{U}(\mathbf{F} - \mathbf{Y})], \tag{17}$$

where $\mathbf{L} = \mathbf{D} - \mathbf{A}$ is the Laplacian matrix, $\mathbf{U}$ is a diagonal matrix with the i-th entry $U_{ii} = \eta_l$ if $i = 1, \ldots, l$ and $U_{ii} = \eta_u$ if $i = l+1, \ldots, l+u$. $\eta_l = \eta_\infty$ (η_∞ is a very large number) and $\eta_u = 0$ are the weighting parameters for the labeled and unlabeled data points respectively. When $\mathbf{U} = \mathbf{I}$, the problem of (17) reduces to the formulation of Eq. (1). $\mathbf{Y}$ is initialized as described in Subsection 2.1.

Since Eq. (17) is an unconstrained quadratic programming problem with respect to $\mathbf{F}$, we can obtain its closed form solution. Taking the derivative of Eq. (17) with respect to $\mathbf{F}$ and setting it to zero, we have

$$\mathbf{L}\mathbf{F} + \mathbf{U}(\mathbf{F} - \mathbf{Y}) = \mathbf{0}_n \Rightarrow \mathbf{F} = (\mathbf{L} + \mathbf{U})^{-1}\mathbf{U}\mathbf{Y}. \tag{18}$$

Note that the obtained label for each data is the probabilistic value. It can be easily verified that the sum of each row of F is equal to 1. We have

$$\begin{aligned} &\left\{\begin{array}{l} \mathbf{L}\mathbf{1}_n = \mathbf{0}_n \\ \mathbf{Y}\mathbf{1}_{C+1} = \mathbf{1}_n \end{array}\right\} \Rightarrow \mathbf{U}\mathbf{Y}\mathbf{1}_{C+1} = (\mathbf{L} + \mathbf{U})\,\mathbf{1}_n \\ &\Rightarrow (\mathbf{L} + \mathbf{U})^{-1}\mathbf{U}\mathbf{Y}\mathbf{1}_{C+1} = \mathbf{1}_n \Leftrightarrow \mathbf{F}\mathbf{1}_{C+1} = \mathbf{1}_n. \end{aligned} \tag{19}$$

Therefore, F_{ij} can be seen as an estimation of the posterior probability of $\mathbf{x}_i$ belonging to the j-th class. When $j = C + 1$, F_{ij} is the probability of i-th point which stops at one of the unlabeled data point after random walks. In other words, it denotes the probability of $\mathbf{x}_i$ to be the outliers. In the next subsection, it is convenient to select discriminative features via label regression with soft labels.

3.2 Formulation of ISR

To obtain the optimal $\mathbf{W}$, the regression model has been widely applied into many applications for its efficiency and simplicity. The difference of various regression models lies in the selection of various loss functions and regularizer. In literature, there are many ways to define loss functions, like hinge loss, l_2-norm loss [37], and $l_{2,1}$-norm loss [13]. However, l_2-norm loss and hinge loss are highly susceptible to the false labeled samples or outliers in training data [13], [23]. For semi-supervised learning, $l_{2,1}$-norm loss only uses labeled data and ignores many unlabeled data. Meanwhile, in real-world applications, the false labeled samples often appear in the data sets and the number of labeled data is few while large amount of unlabeled samples exist. Thus, traditional loss functions may not get the optimal performance in this case.

In general, we adopt p power of l_2-norm to measure the loss of instances, that is $\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p$ and $0 \le p \le 2$. The soft label matrix $\mathbf{F}$ includes more sufficient and richer semantic information than only labels of labeled data after label propagation. It can enlarge the discriminative and robust performance when it is used in our regression model. Therefore, $\mathbf{F}$ is introduced into our formulation as the weight of its corresponding loss of each data and each semantic class, that is $F_{ij}\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon)$. By analysis of label prorogation, F_{ij} can be seen as an estimation of the posterior probability of $\mathbf{x}_i$ belonging to the j-th class. In other words, if $\mathbf{x}_i$ does not belong to the j-th class, the value of F_{ij} will be very close to zero, even for the false labeled data or outliers. And then the value of $F_{ij}\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon)$ will also become very close to zero. The current loss has little effect on the optimization of the optimal $\mathbf{W}$. Thus the influence of the false labeled instances or outliers in the training data is further removed by the soft label matrix $\mathbf{F}$. To eliminate the impact of false labeled samples or outliers and measure the loss of the current data point $\mathbf{x}_i$ with the j-th class simultaneously, we propose to use capped l_2-l_p-norm loss function

$$\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon), \tag{20}$$

under the constraints $\varepsilon > 0$ and $0 < p \le 2$, where $\mathbf{t}_j = [\underbrace{0,\ldots,0}_{j-1}, 1, \underbrace{0,\ldots,0}_{c-j}]^T$ is the class indicator vector for the j-th class, $\mathbf{b} \in \mathbb{R}^{c\times 1}$ is the bias term, $\varepsilon > 0$ is a threshold parameter. If $\mathbf{x}_i$ is a false labeled point or an outlier point in $\mathbf{X}$, the values of $\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p$ for $j = 1,\ldots,C$, are all very large. The hard-thresholding operator simply sets $\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon)$ to ε if the value of $\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p$ is larger than ε. Therefore, the impact of false labeled data or outlier will be eliminated.

In a word, the formulation of our proposed regression model named Insensitive Regression Model (IRM) is summarized as

$$\sum_{i=1}^{n}\sum_{j=1}^{C} F_{ij}\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon). \tag{21}$$

Note that $l_{2,1}$-norm and l_2-norm loss are two special cases of capped l_2-l_p-norm loss and they are $p = 1, \varepsilon \to \infty$ and $p = 2, \varepsilon \to \infty$, respectively. The threshold ε and $\mathbf{F}$ are able to jointly accelerate the convergence of objective function.

The second term of ISR is the regularizer which is designed for feature selection. Evoked by the basic idea of sparse regression [13], we also want to add sparse constraints on transformation matrix to measure its sparsity in regression. The magnitude of each row of $\mathbf{W}$ represents the weight of each feature. To perform feature selection, we need to push some rows of $\mathbf{W}$ shrink to zero, and then remove the corresponding features. Consequently, the corresponding features can be neglected since these features are redundant for regression.

From the sparsity perspective, the ideal regularization term will be $l_{2,0}$-norm regularizer [12]. However $l_{2,0}$-norm minimization problem is NP-hard and be very difficult to solve directly, due to its non-convex and non-smooth properties. Lots of researchers prefer to choose the convex $l_{2,1}$-norm as the regularizer of regression model [13], [27]. Because $l_{2,1}$-norm minimization has rotational invariant property and can be solved by an iterative algorithm [13] directly. However, we approximate the $l_{2,0}$-norm by $l_{2,q}$-norm with better sparsity when $0 < q \le 1$ as in [41], [42], [43], [44]. The $l_{2,q}$-norm of $\mathbf{W}$ is the q power of l_2-norm of $\mathbf{w}^i$, i.e. the i-th row of $\mathbf{W}$ to measure its contribution in regression,

$$\sum_{i=1}^{d}\left(\left\|\mathbf{w}^i\right\|_2\right)^q = \sum_{i=1}^{d}\left(\sum_{j=1}^{C}|W_{ij}|^2\right)^{q/2} = \|\mathbf{W}\|_{2,q}^q \tag{22}$$

where q keeps the balance between the sparsity and convexity of the regularizer and $0 < q \le 1$. Obviously, when $q = 1$, the above formulation will reduce to $l_{2,1}$-norm regularizer. The closer the value of q is to zero, the better approximation the objective function is to the original feature selection problem. When $q = 0$, the regularizer is non-convex. When $q = 1$, it is the closest convex approximation of $l_{2,0}$-norm.

Finally, by combining Eq. (21) and Eq. (22), the formulation of ISR can be summarized as follows.

$$\min_{\mathbf{W},\mathbf{b}} \sum_{i=1}^{n}\sum_{j=1}^{C} F_{ij}\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon) + \gamma\|\mathbf{W}\|_{2,q}^q, \tag{23}$$

where γ controls the regularization effect.

Similar with [3], [13], we adopt the l_2-norm of $\mathbf{w}^i$ to evaluate the importance of each feature. The larger this value is, the more important this element is. Once we obtain the optimal transformation matrix $\mathbf{W}^*$, we can rank each feature $\mathbf{f}_i$ according to $\left\|\mathbf{w}^{*i}\right\|_2$ in descending order and then select the top ranked features. In the following, we select a fixed number, i.e., s, features for evaluation.

Since IRM in Eq. (21) is non-convex, the problem (23) is also non-convex. It is difficult to get its closed solution directly. In this paper, we propose an efficient iterative algorithm to solve our proposed ISR for $0 < q \le 1$. The details of optimization will be introduced in next subsection.

3.3 Optimization and Solution

Since both of terms are non-convex, ISR cannot be solved directly. Based on the concept of function, the functions $f(\mathbf{W},\mathbf{b}) = \left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p$ and $g(\mathbf{W},\mathbf{b}) = \varepsilon$ are continuous with respect to $\mathbf{W}$ and $\mathbf{b}$. Thus the term $\min(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p, \varepsilon)$ is continuous. By Proposition 3.2.3 and Proposition 3.2.8 of [45], we will know that the conditions of problem (23) hold. Therefore, we can use Lagrangian multiplier method to solve it. In [13], Nie et al. adopted an iterative reweighed algorithm to solve the $\ell_{2,1}$-norm minimization problems. In [43], Gong et al. proposed a common procedure for solving the l_2-l_1-norm optimization problem. Evoked by these works, we replace the original non-convex formulation as a convex problem with the proved convergence behavior.

Denote $\tilde{F}_{ij} =$

$$\frac{p}{2}F_{ij}\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^{p-2} Ind(\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p \le \varepsilon) \tag{24}$$

where $Ind(\cdot)$ is an indicative function, which is equal to 1 if $\left\|\mathbf{W}^T\mathbf{x}_i + \mathbf{b} - \mathbf{t}_j\right\|_2^p \le \varepsilon$ and 0 otherwise. Then the formulation in Eq. (23) will be converted into the following reweighted least square regression problem with $l_{2,q}$-norm regularization:

$$\min_{\mathbf{W},\mathbf{b}} \sum_{i=1}^{n} \sum_{j=1}^{C} \tilde{F}_{ij} \left\| \mathbf{W}^T \mathbf{x}_i + \mathbf{b} - \mathbf{t}_j \right\|_2^2 + \gamma \left\| \mathbf{W} \right\|_{2,q}^q . \tag{25}$$

According to the matrix calculus theory, Eq. (25) will be expanded to the following formulation.

$$\begin{aligned} \min_{\mathbf{W},\mathbf{b}} \sum_{i=1}^{n} \sum_{j=1}^{C} \tilde{F}_{ij} \left(\mathbf{W}^T \mathbf{x}_i + \mathbf{b} \right)^T \left(\mathbf{W}^T \mathbf{x}_i + \mathbf{b} \right) + \sum_{i,j} \tilde{F}_{ij} \mathbf{t}_j^T \mathbf{t}_j \\ -2 \sum_{i=1}^{n} \sum_{j=1}^{C} \tilde{F}_{ij} \left(\mathbf{W}^T \mathbf{x}_i + \mathbf{b} \right)^T \mathbf{t}_j + \gamma \left\| \mathbf{W} \right\|_{2,q}^q . \end{aligned} \tag{26}$$

Because $\mathbf{t}_j^T \mathbf{t}_j = 1$ is a constant with respect to $\mathbf{W}$ and $\mathbf{b}$, we can omit the second term. Thus, the above formulation is equal to

$$\begin{aligned} J(\mathbf{W},\mathbf{b}) = Tr \left[\left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right) \mathbf{S} \left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right)^T \right] \\ - 2Tr \left[\tilde{\mathbf{F}} \left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right) \right] + \gamma \left\| \mathbf{W} \right\|_{2,q}^q . \end{aligned} \tag{27}$$

Where $\mathbf{S}$ is a n-dim diagonal matrix with the i-th element being

$$S_{ii} = \sum\nolimits_{j=1}^{C} \tilde{F}_{ij} . \tag{28}$$

Recall the basic idea in solving the sparse regularization problem as in [13], we take the derivative of $\|\mathbf{W}\|_{2,q}^q$ with respect to $\mathbf{W}$. For convenience, we denote $\ell(\mathbf{W}) = \|\mathbf{W}\|_{2,q}^q$. When $\mathbf{w}^i \neq 0$ for $i = 1, \ldots, d$, the derivative of $\ell(\mathbf{W})$ w.r.t $\mathbf{W}$ is

$$\partial \ell(\mathbf{W}) / \partial \mathbf{W} = \mathbf{D} \mathbf{W}, \tag{29}$$

where $\mathbf{D}$ is a diagonal matrix with the i-th element as:

$$D_{ii} = \frac{q}{2} \left\| \mathbf{w}^i \right\|_2^{q-2} . \tag{30}$$

Recalling the definition of D_{ii}, if $\mathbf{w}^i \neq 0$, we can get $Tr(\mathbf{W}^T \mathbf{D} \mathbf{W}) = \|\mathbf{W}\|_{2,q}^q / 2$. Thus, the minimization of $Tr(\mathbf{W}^T \mathbf{D} \mathbf{W})$ will add the row sparsity on $\mathbf{W}$ when $0 < q \leq 1$. When $\mathbf{D}$ and $\tilde{\mathbf{F}}$ are fixed, the derivative of J in Eq. (27) can be regarded as the derivative of the following objective function.

$$\begin{aligned} J(\mathbf{W},\mathbf{b},\tilde{\mathbf{F}},\mathbf{D}) = Tr \left[\left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right) \mathbf{S} \left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right)^T \right] \\ - 2Tr \left[\tilde{\mathbf{F}} \left(\mathbf{W}^T \mathbf{X} + \mathbf{b} \mathbf{1}^T \right) \right] + \gamma Tr(\mathbf{W}^T \mathbf{D} \mathbf{W}). \end{aligned} \tag{31}$$

With this approximation, ISR can be effectively solved with proved convergence. Although the problem in Eq. (31) is much easier than that in Eq. (23), it is also difficult to get its solution directly, since $\tilde{\mathbf{F}}$ is also dependent on $\mathbf{W}$ and $\mathbf{b}$. Recalling the definition of $\mathbf{S}$ in Eq. (28), we know that when $\tilde{\mathbf{F}}$ and $\mathbf{D}$ is fixed, Eq. (31) becomes a regularized least square problem. Take derivative of $J(\mathbf{W},\mathbf{b},\tilde{\mathbf{F}},\mathbf{D})$ with respect to $\mathbf{W}$ and $\mathbf{b}$ and set them to zeros, the optimal solutions to the problem in Eq. (31) are

$$\frac{\partial J(\mathbf{W},\mathbf{b},\tilde{\mathbf{F}},\mathbf{D})}{\partial \mathbf{W}} = \gamma \mathbf{D} \mathbf{W} + \mathbf{X} \mathbf{S} \mathbf{X}^T \mathbf{W} + \mathbf{X}(\mathbf{S} \mathbf{1} \mathbf{b}^T - \tilde{\mathbf{F}}) \tag{32}$$

TABLE 1
The Main Procedure of Proposed ISR.

Input: Data set $\{\mathbf{x}_i | i = 1, 2, \cdots, n\}$ with $\mathbf{Y}_l$, balance parameter γ, ϵ, neighborhood size k, and selected feature number s.
Output: Selected feature index set $\{r_1, r_2, \cdots, r_s\}$.

```
Stage one: Computing Soft Labels via Label Propagation
1. Construct the nearest neighborhood graph $\mathcal{G}$;
2. Compute the similarity matrix $\mathbf{A}$ and soft label $\mathbf{F}$ by Eq. (17);
Stage two: Alternative optimization
1. Initialize $\mathbf{D} = \mathbf{I}_{d \times d}$ and $\tilde{F}_{ij} = F_{ij}$;
2. Alternatively update $\tilde{\mathbf{F}}$, $\mathbf{D}$ and $\mathbf{W}$ until convergence.
  a. Fix $\mathbf{D}$ and $\tilde{\mathbf{F}}$, update $\mathbf{W}$ and $\mathbf{b}$ by solving the problem Eq. (34);
  b. Fix $\mathbf{W}$ and $\mathbf{b}$, update $\tilde{\mathbf{F}}$ and $\mathbf{D}$ by Eq. (24) and Eq. (30);
Stage three: Feature selection
1. Compute the scores for all features $\{\|\hat{\mathbf{w}}_i\|_2\}_{i=1}^d$;
2. Sort these scores and select the largest $s$ values. Their
   corresponding indexes of the selected features $\{r_1, r_2, \cdots, r_s\}$.
```

$$\frac{\partial J(\mathbf{W},\mathbf{b},\tilde{\mathbf{F}},\mathbf{D})}{\partial \mathbf{b}} = \mathbf{W}^T \mathbf{X} \mathbf{S} \mathbf{1} + \mathbf{1}^T \mathbf{S} \mathbf{1} \mathbf{b} - \tilde{\mathbf{F}}^T \mathbf{1} \tag{33}$$

$$\mathbf{b} = \frac{1}{\mathbf{1}^T \mathbf{S} \mathbf{1}} \left(\tilde{\mathbf{F}}^T - \mathbf{W}^T \mathbf{X} \mathbf{S} \right) \mathbf{1}, \mathbf{W} = \left[\mathbf{X} \mathbf{L}_s \mathbf{X}^T + \gamma \mathbf{D} \right]^{-1} \mathbf{X} \mathbf{C}_s \tilde{\mathbf{F}}, \tag{34}$$

where $\mathbf{C}_s = \mathbf{I} - \frac{1}{\mathbf{1}^T \mathbf{S} \mathbf{1}} \mathbf{S} \mathbf{1} \mathbf{1}^T, \mathbf{L}_s = \mathbf{C}_s \mathbf{S}$.

When $\mathbf{W}$ and $\mathbf{b}$ are computed, we can update $\tilde{\mathbf{F}}$, $\mathbf{S}$ and $\mathbf{D}$ by employing the formulations in Eq. (24), Eq. (28) and Eq. (30) directly.

In summary, we solve the optimization problem in Eq. (27) in an alternative way. More concretely, we initialize $\mathbf{D} = \mathbf{I}_{d \times d}$ and $\tilde{F}_{ij} = F_{ij}$ and update $\mathbf{W}$ and $\mathbf{b}$ by Eq. (34). After that, we fix $\mathbf{W}$ and $\mathbf{b}$, and then compute $\tilde{\mathbf{F}}$, $\mathbf{D}$ and $\mathbf{S}$. Additionally, the experimental results show that our proposed algorithm converges fast. The number of iterations is often less than 15. The main procedure of ISR is listed in Table 1.

Remark 1. *When computing $\mathbf{D}$, its diagonal element $D_{ii} = \frac{q}{2} \left\| \mathbf{w}^i \right\|_2^{q-2}$. In practice, $\left\| \mathbf{w}^i \right\|_2$ could be very close to zero but not zero. However, $\left\| \mathbf{w}^i \right\|_2$ can be zero theoretically. In this case, we regularize D_{ii} as $D_{ii} = q \Big/ (2 \left\| \mathbf{w}^i \right\|_2^{2-q} + \tau)$, where τ is a very small constant. When $\tau \to 0$, it is easy to see that $q \Big/ (2 \left\| \mathbf{w}^i \right\|_2^{2-q} + \tau)$ approximates $\frac{q}{2} \left\| \mathbf{w}^i \right\|_2^{q-2}$.*

4 DISCUSSIONS

4.1 Convergence Analysis

As mentioned above, the formulation of ISR is solved in an alternative way; namely, we fix one group of variables and optimize the other. In this subsection, we will give a theoretical analysis about the convergence of our method in Eq.(23) by the above iteration process. The following two propositions show that we can obtain the optimal solution in each step.

Proposition 1. *The procedures of ISR shown in Table 1 will monotonically decrease the objective function of (31) in each step.*

Due to the limitation of space we would like to get rid of the detailed proof of proposition 1 in the main parts of this paper. The detailed proof of proposition 1 is presented on supplementary material. The main idea of the proof is listed as follows.

According to Lemma 1 (listed in the supplementary material), we have

$$\left\|\mathbf{w}^i_{(k+1)}\right\|_2^q \Big/ \left\|\mathbf{w}^i_{(k)}\right\|_2^q - \frac{q}{2}\left\|\mathbf{w}^i_{(k+1)}\right\|_2^2 \Big/ \left\|\mathbf{w}^i_{(k)}\right\|_2^2 \le 1-\frac{q}{2} \quad (35)$$

Assume that we have derived D and $\tilde{\mathbf{F}}$ as $\mathbf{D}_{(k)}$ and $\tilde{\mathbf{F}}_{(k)}$ in the k-th step. In the $(k+1)$-th iteration, we fix $\mathbf{D}$ and $\tilde{\mathbf{F}}$ as $\mathbf{D}_{(k)}$ and $\tilde{\mathbf{F}}_{(k)}$ and then optimize $\mathbf{W}$ and $\mathbf{b}$. When $\tilde{\mathbf{F}}$ and $\mathbf{D}$ are fixed in Eq. (31), the formulation in Eq. (31) is the regularized least square regression model. In other word, there is a closed solution for Eq. (31) in this case. So we have the following inequality:

$$J(\mathbf{W}_{(k+1)},\mathbf{b}_{(k+1)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k)}) \le J(\mathbf{W}_{(k)},\mathbf{b}_{(k)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k)}). \quad (36)$$

Combining Eq. (35) and Eq. (36), we can get

$$\begin{aligned}
&Tr\left[\left(\mathbf{W}^T_{(k+1)}\mathbf{X}+\mathbf{b}_{(k+1)}\mathbf{1}^T\right)\mathbf{S}\left(\mathbf{W}^T_{(k+1)}\mathbf{X}+\mathbf{b}_{(k+1)}\mathbf{1}^T\right)^T\right]\\
&-2Tr\left[\tilde{\mathbf{F}}_{(k)}\left(\mathbf{W}^T_{(k+1)}\mathbf{X}+\mathbf{b}_{(k+1)}\mathbf{1}^T\right)\right]+\gamma\sum_{i=1}^{d}\left\|\mathbf{w}^i_{(k+1)}\right\|_2^q\\
&\le Tr\left[\left(\mathbf{W}^T_{(k)}\mathbf{X}+\mathbf{b}_{(k)}\mathbf{1}^T\right)\mathbf{S}\left(\mathbf{W}^T_{(k)}\mathbf{X}+\mathbf{b}_{(k)}\mathbf{1}^T\right)^T\right]\\
&-2Tr\left[\tilde{\mathbf{F}}_{(k)}\left(\mathbf{W}^T_{(k)}\mathbf{X}+\mathbf{b}_{(k)}\mathbf{1}^T\right)\right]+\gamma\sum_{i=1}^{d}\left\|\mathbf{w}^i_{(k)}\right\|_2^q,
\end{aligned} \quad (37)$$

where $\mathbf{w}^i_{(k+1)}$ and $\mathbf{w}^i_{(k)}$ are the i-th row of the matrix $\mathbf{W}_{(k+1)}$ and $\mathbf{W}_{(k)}$ respectively. That is to say, the equation is equivalent to the following inequality

$$J(\mathbf{W}_{(k+1)},\mathbf{b}_{(k+1)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k+1)}) \le J(\mathbf{W}_{(k)},\mathbf{b}_{(k)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k)}). \quad (38)$$

Similar to [43], we prove that the procedure of updating $\tilde{\mathbf{F}}_{(k)}$ also decreases the objective function value. In other words, we have the following inequality holds:

$$\begin{aligned}
&J(\mathbf{W}_{(k+1)},\mathbf{b}_{(k+1)},\tilde{\mathbf{F}}_{(k+1)},\mathbf{D}_{(k+1)}) \le\\
&J(\mathbf{W}_{(k+1)},\mathbf{b}_{(k+1)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k+1)}).
\end{aligned} \quad (39)$$

Finally, combining Eq. (38) and Eq. (39), we can arrive at

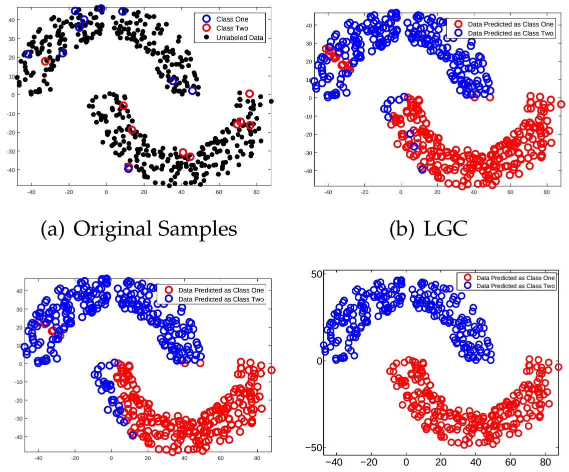
$$J(\mathbf{W}_{(k+1)},\mathbf{b}_{(k+1)},\tilde{\mathbf{F}}_{(k+1)},\mathbf{D}_{(k+1)}) \le J(\mathbf{W}_{(k)},\mathbf{b}_{(k)},\tilde{\mathbf{F}}_{(k)},\mathbf{D}_{(k)}). \quad (40)$$

This inequality indicates that the objective function in Eq. (31) will monotonically decrease in each iteration until the algorithm converges.

Proposition 2. *The procedures of ISR shown in Table 1 will monotonically decrease the objective function in Eq. (23) in each step and then converge to the local optimum of the problem.*

Based on the conclusion of proposition 1, we adopt the similar strategy in [46] to prove it.

On the one side, in each iteration of ISR in Table 1, when one group of variables is fixed, we can find the optimal solution to the problem in (31). Thus the derived solution of ISR satisfies the KKT condition of problem (31). Taking the derivative w.r.t. $\mathbf{W}$ and $\mathbf{b}$ respectively and setting them to zero, we get the KKT conditions Eq. (32) and Eq. (33) of the problem (31).



(a) Original Samples (b) LGC

(c) GFHF (d) IRM

Fig. 1. Examples illustrating the sensitivity of IRM to the false labeled instances on Two Moon data set. (a) shows the original data samples and the selected labeled samples. (b)-(d) show the prediction results by LGC, GFHF and IRM, respectively.

On the other side, according to optimization theory, the Lagrangian function of Eq. (23) is

$$J\left(\mathbf{W},\mathbf{b}\right)=\sum_{i=1}^{n}\sum_{j=1}^{C}F_{ij}\min(\left\|\mathbf{W}^T\mathbf{x}_i+\mathbf{b}-\mathbf{t}_j\right\|_2^p,\varepsilon)+\gamma\left\|\mathbf{W}\right\|_{2,q}^q. \quad (41)$$

Taking the derivative w.r.t. $\mathbf{W}$ and $\mathbf{b}$ respectively and setting them to zero, we get the KKT conditions of the problem in Eq. (41). Using the matrix calculus, we can write the KKT conditions of the problem in Eq. (41) as follows

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial\mathbf{W}}=\gamma\mathbf{D}\mathbf{W}+\mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W}+\mathbf{X}\mathbf{S}\mathbf{1}\mathbf{b}^T-\mathbf{X}\tilde{\mathbf{F}}, \quad (42)$$

$$\frac{\partial J\left(\mathbf{W},\mathbf{b}\right)}{\partial\mathbf{b}}=\mathbf{W}^T\mathbf{X}\mathbf{S}\mathbf{1}+\mathbf{1}^T\mathbf{S}\mathbf{1}\mathbf{b}-\tilde{\mathbf{F}}^T\mathbf{1}. \quad (43)$$

According to the definition of D_{ii} and $\tilde{\mathbf{F}}$ in the main procedures of ISR, we can see that Eq. (42) and Eq. (43) are the same as Eq. (32) and Eq. (33). Therefore, the converged solution of Algorithm in Table 1 satisfies Eq. (42) and Eq. (43), the KKT conditions of problem in Eq. (23). Thus the solution of algorithm in Table 1 is a local minimal solution to the problem in Eq. (23). The detailed proof of proposition 2 is also shown in supplementary material.

4.2 Computational Complexity Analysis

In this subsection, we analyze the computational complexity of ISR. The computational cost of constructing the graph Laplacian is $O(n^2d)$, where n and d are the number data and features, respectively. Following the idea in [37], the label propagation can be viewed as a process of random walks. Given the weight matrix A, the computation complexity of calculating the soft label matrix F with (11) is $O(knc)$, where k and c are the number of neighbors and classes. The major computational cost of our ISR lies in the updating of $\mathbf{W}=\mathbf{P}_s^{-1}\mathbf{Q}_s$, where $\mathbf{Q}_s=\mathbf{X}\big(\mathbf{I}-\frac{1}{\mathbf{1}^T\mathbf{S}\mathbf{1}}\mathbf{S}\mathbf{1}\mathbf{1}^T\big)\tilde{\mathbf{F}},\mathbf{P}_s=\mathbf{X}\big(\mathbf{I}-\frac{1}{\mathbf{1}^T\mathbf{S}\mathbf{1}}\mathbf{S}\mathbf{1}\mathbf{1}^T\big)\mathbf{S}\mathbf{X}^T+\gamma\mathbf{D}$ are a $d\times c$ matrix and a $d\times d$ matrix, respectively. The computational cost of matrix inverting is $O(d^3)$. In fact, the computation of matrix inverse can be avoided. Note that we are aiming to compute $\mathbf{P}_s^{-1}\mathbf{Q}_s$, which is the solution of the following problem:

$$\min_{\mathbf{W}} \mathbf{W}^T\mathbf{P}_s\mathbf{W} - 2\mathbf{W}^T\mathbf{Q}_s. \tag{44}$$

The solution of this minimization problem can be get iteratively using gradient descent method using the updating formula: $\mathbf{W}_{t+1} = \mathbf{W}_t - \alpha(\mathbf{P}_s\mathbf{W}_t - \mathbf{Q}_s)$, with a computational cost of $O(Td^2c)$, where T is the number of iterations and α is a step size of gradient. Therefore, the total computational cost of our ISR is upper bounded by $O(Td^2c) + O(n^2d) + O(knc)$. Moreover, the number of iterations T and the number of classes c are often much less than the dimension of features d for high dimensional data. Thus, our algorithm can obtain the optimal $\mathbf{W}$ efficiently.

4.3 Sensitivity to False Labels

As adding capped l_2-l_p-norm loss into IRM, ISR is insensitive to the false labels or outliers. In this section, to discuss the sensitivity of false labeled samples, we present the robustness performance of IRM on a synthetic data.

These experiments include two parts: one is a simple example showing the robustness performance of IRM (shown in Fig. 1), another is the comparison of the average error rates (listed in Table 2). We generate 250 samples per class of two-moon artificial data randomly. After label propagation, we only use IRM to learn the projection matrix $\mathbf{W}$ and $\mathbf{b}$ without regularization term, that is $\gamma = 0$ in ISR. And then the labels of the training samples are predicted by $\min\limits_{j} \{a_j \,|a_j = \left\|\mathbf{W}^T\mathbf{x} + \mathbf{b} - \mathbf{t}_j\right\|, j = 1 \ldots C\}$.

In the first experiment, we choose 10 labeled samples which includes one sample with false label and the remaining 240 unlabeled instances as shown in Fig.1(a). The results of IRM and label propagation methods: LGC and GFHF depict in Fig. 1. From the results in Fig. 1, we can see that ISR can eliminate the effect of false labeled instances (as Fig. 1(d)) by IRM. For LGC and GFHF, as shown Fig. 1(b), even in the case of only one false labeled sample for each class, the instances around the false labeled points is still badly affected after label propagation method.

To further analyze the performance of IRM, we compute the average error rates under different ratios of false labels in different number of labeled points. In this experiment, we randomly select 10, 20 and 50 labeled samples with $\{10\%, 20\%, 30\%, 40\%\}$ false labeled instances from original 250 samples. Then we repeat the tests 50 times and calculate the average error rates. The results are listed in Table 2. Obviously, the error rates of IRM is much less than the results of LGC and GFHF, and original ratios. For example, when the number of labeled samples is 50 and the ratio of false labeled samples is up to 40%, the error rate of IRM is 29.05% less than label propagation and 28.06% less than original ratio. Consequently, IRM can eliminate the effect of the propagated label mistakes of false labeled samples and is much more robust than label propagations, when its ratio is less than fifty percent of labeled instances.

5 Experiments

5.1 Data Description

TABLE 2
The error rates of label propagation methods LGC, GFHF and IRM on 2D toy data with different number of labeled samples $n_l = \{10, 20, 50\}$ and different ratios of noisy instances $\{10\%, 20\%, 30\%, 40\%\}$.

n_l	Ratio	LGC	GFHF	IRM
10	10%	12.48±0.66	12.27±0.70	1.31±0.25
	20%	24.87±0.90	24.11±0.86	9.60±0.68
	30%	32.80±0.90	31.87±0.91	14.11±0.96
	40%	40.53±0.83	39.74±0.78	25.74±1.24
20	10%	12.36±0.42	11.00±0.45	4.31±0.25
	20%	22.11±0.55	20.70±0.57	5.46±0.30
	30%	30.94±0.64	28.97±0.74	7.91±0.63
	40%	40.55±0.89	39.09±0.93	19.82±0.91
50	10%	7.90±0.24	6.74±0.25	2.34±0.18
	20%	20.74±0.32	14.93±0.29	3.55±0.22
	30%	31.33±0.35	26.76±0.37	5.96±0.53
	40%	40.99±0.57	39.07±0.46	11.94±0.75

TABLE 3
Data sets Descriptions

Dataset	Size	Dim	#Class	Type
Umist	575	644	20	Image, Face
Coil20	1440	2048	20	Image, Object
USPS	9298	256	10	Image, Handwritten
PIE	11554	1024	68	Image, Face
KSA	20000	1590	10	Image, Action
MNIST	70000	784	10	Image, Handwritten
Epsilon	400000	2000	2	Variables, Vision
Covtype	581012	54	2	Variables, Forest

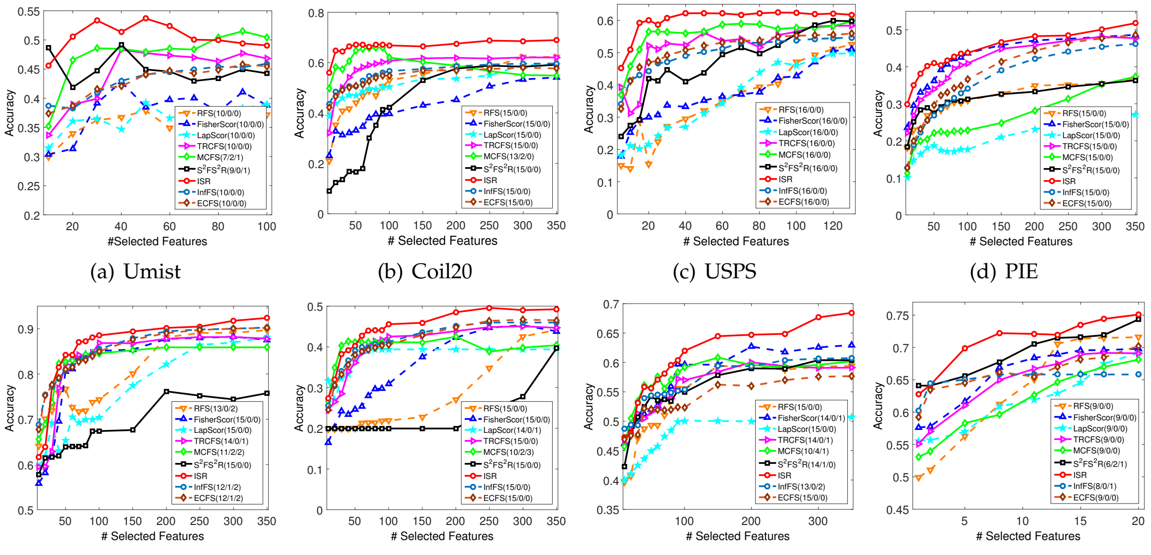
In our experiments, eight public data sets with various statistical characters are collected to present the performance of different feature selection methods. These data sets include five image data sets including Umist[1], Coil20[2], MNIST[3], USPS[4], PIE[5], one action recognition data set Kinect Skeleton Action (KSA)[6] and two large scale learning data sets, Covertype[7] and Epsilon[8]. All data sets are normalized with zero-man and unit length. We summarize the character of them in Table 3.

5.2 Evaluation Metric

To test the quality of selected features, we employ two different kinds of evaluation metrics for the classification task, i.e., the classification accuracy achieved by classifier using the selected features; REDundancy rate (RED) [3] contained in the selected features. The redundancy is a popular evaluation metric for feature selection. It measures the quality of selected features directly, without employing the subsequent tasks. This measurement assesses the averaged correlation among all feature pairs. A large value indicates that many selected features are correlated. Thus redundancy is expected to exist in the set of selected features $\mathcal{F}$.

To compute the classification accuracy, we use the linear Support Vector Machine classifier (SVM) [47] to perform classification on the data with selected features. We randomly select a fixed number of labeled and unlabeled examples from each category as training data and the rest

[1] http://images.ee.umist.ac.uk/danny/database.html
[2] http://www.cs.columbia.edu/CAVE/research/coil-20.html
[3] http://yann.lecun.com/exdb/mnist/
[4] http://www.cad.zju.edu.cn/home/dengcai/Data/USPS
[5] http://www.uk.research.att.com/facedatabase.html
[6] http://www.cs.cmu.edu/ kevinma/data/KSA.mat
[7] http://archive.ics.uci.edu/ml/datasets/Covertype
[8] http://largescale.ml.tu-berlin.de/largescale/epsilon/



(a) Umist (b) Coil20 (c) USPS (d) PIE

(e) KSA (f) MNIST (g) Epsilon (h) CovType

Fig. 2. Classification accuracy of SVM with different number of selected features. The three numbers in brackets of legends represent the times of *win*, *balance* and *fail* of ISR for t-test with a threshold of 0.05 statistical significance under different numbers of selected features.

are assigned as testing data. Meanwhile, we take the results of all features as baseline and compare ISR with six state-of-the-art feature selection methods: Laplacian score (LapScor) [29], MCFS [30], Fisher score (FisherScor) [31], robust feature selection via joint $l_{2,1}$-norms minimization (RFS) [13], TRCFS [19], S^2FS^2R [8], Infinite Feature Selection(InfFS) [22] and Feature Selection via Eigenvector Centrality(ECFS) [28], [33] on above eight public data sets, where the parameters ϵ and γ are selected by grid search in a heuristic way. Other parameters are empirically determined as the traditional learning approaches [48], [49].

5.3 Comparison between ISR and Other Feature Selection Algorithms

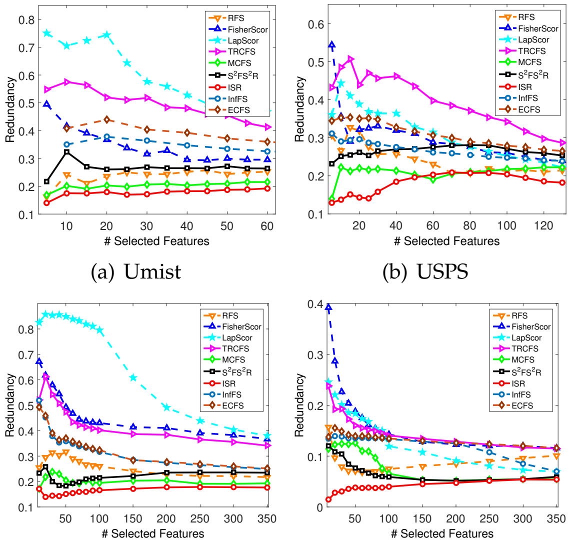
We compare the classification accuracy and redundancy rate (RED) of different methods on these data sets. There are different number of sample sizes for different data sets. Thus we randomly select 3, 3, 10, 10, 15, 100, 100 and 100 instances with label information per class from Umist, Coil20, USPS, PIE, KSA, MNIST, Epsilon and CovType and 40% unlabeled samples as the training data, and the remained samples are used for testing. All the tests were repeated 50 times, and we then calculate the average classification accuracy and redundancy rate (RED). Moreover, to analyze the performance of ISR, we compare our method with other approaches by Students t-test in the experiments. Since different data sets have different dimensions of features, we select various number of features according to the ranked feature indexes $\{r_1, r_2, \cdots, r_s\}$. Similar to [3], other parameters are determined by cross validation if necessary.

For classification task, each feature selection algorithm is first performed on training data to select optimal features. Then we use SVM to classify the testing samples that represented by the selected features. The mean classification accuracy results and the results of t-test are shown in Fig. 2. From another point view, we adopt the redundancy rate (RED) to comprehensively evaluate the performance of feature selection methods. Fig. 3 represents the corresponding average redundancy rate (RED) when different numbers of features are selected by different feature selection algorithms. Due to the limitation of space, we only report the results on four representative data sets.

As seen from the results in Fig. 2, the classification accuracies of different methods vary with the increase of the number of selected features. For all data sets except Umist and Coil20, all feature selection approaches achieve higher classification accuracy with more selected features. A similar tendency can be found on Coil20, with only ISR's performance fluctuating. For Umist data set, the accuracy achieved by each method fluctuates within a certain range. With more features, the data can be characterized better and gradually close to the accuracy of baseline method. Meanwhile, the experimental results of t-test are listed in Fig. 2 with a threshold of 0.05 statistical significance. From the statistical view, all of these results indicated that ISR achieves significantly better results comparing to the other algorithms in most cases.

Generally, in most of cases, ISR outperforms all the other feature selection methods on all data sets for classification accuracy. Especially, on the Epsilon data set, ISR achieves 8.07% to 10.25% improvement compared to the best result of all the other methods. Fig. 3 presents that the feature subsets selected by ISR on all four data sets consistently have lower redundancy rate than other methods. Besides, in most cases, redundancy rate of selected feature subset decreases as the number of selected features increases. In terms of the RED results, our method consistently performs better than all the other approaches and baseline. The results also show that our ISR outperforms other methods in most cases, even when we take all the features as the input.

In summary, our method can enhance feature selection performance for classification via insensitive sparse regression model. There are two main reasons for this. First, ISR adopts label propagation to enlarge label information of the training data. Second, we use the learning mechanism by adding sparse constraints and capped l_2-l_p-norm loss for



(a) Umist (b) USPS

(c) KSA (d) MNIST

Fig. 3. The redundancy rate of the sets of features of different size selected by different feature selection methods.

feature selection. Thus ISR tends to be more robust to false labeled instances and select lower redundancy rate features.

5.4 Computational Time Comparison

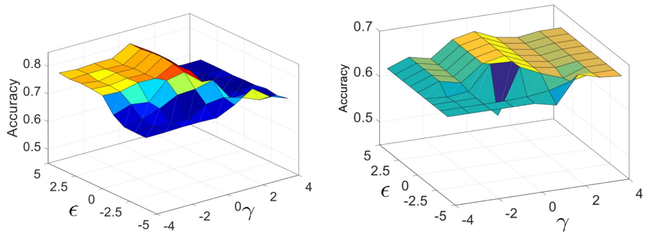
In this section, we provide some experimental results to compare the computational time. To show the influence of data size n and dimensionality d, we select two data sets, i.e., Coil20 and CovType, since they have the largest n and d among eight data sets. With a naive MATLAB implementation, we report the computational time of ISR and other feature selection methods. We have tested all the algorithms on a laptop with 8 processors (2.27 GHz for each) and 32 GB RAM memory. The results are shown in Table 4.

The experimental results of Table 4 indicate that LapScor costs the least time in all the cases, InfFS and S^{2}FS^{2}R costs the most time in most of cases. The computational costs of our ISR change with different number of instances and different dimension of features.

5.5 Parameter Determination

We provide some results of ISR with different parameters in this section. There are two groups of parameters be determined in our method. One group is p and q, which control the model complexity of ISR. Similar with the settings of [43], [49], we choose $p = q = 1$ in our following experiments. Besides, the experimental results, e.g. Fig. 2, Fig. 3, Fig. 6–Fig. 11, also indicate that ISR under this setting obtains relatively good performance.

Another group is α, ε and γ. α is a step size of gradient, which is computed by the traditional line search method [45] automatically. ε is the threshold parameter and γ controls the trade-off between the sparse regression and sparsity. Since parameter determination is still an open problem, we determine the parameters, i.e., ε and γ, in a heuristic way. More concretely, we determine two parameters by grid search, i.e. ϵ varies from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$ and γ varies from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}\}$. The ACC results of SVM with different ε and γ on COIL20 and USPS data sets are shown in Fig. 4.



(a) Coil20 (b) USPS

Fig. 4. ACC of SVM on COIL20 and USPS with different ϵ and γ. ϵ varies from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$ and γ varies from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}\}$. The x-axis and y-axis represent ϵ and γ in $\log$ scale, respectively. The z-axis is the accuracy results.

As seen from Fig. 4, parameter determination takes influence on the performance of ISR. Different combinations of parameters may result in different selection of features. Then, the ACC results of SVM change.

6 Application to Video Semantic Recognition

Video semantic recognition (VSR) is a fundamental task and still a challenging problem of machine learning and pattern recognition. The difficulties of VSR lie in high complexity of visual patterns and high dimensionality of visual features. In this section, we apply our ISR into sports and consumer action recognition on four benchmark video data sets.

6.1 Video Data Sets

There are totally four public video data sets, Columbia Consumer Video (CCV) [9], Kodak consumer video (Kodak) [10], Olympic sports action videos (Olympic)[11] and UCF sports action videos (UCF)[12]. In the following, we will briefly introduce the detailed description of four data sets.

(1) CCV: CCV contains 9317 web videos over 20 semantic categories, including events like baseball and parade and scenes like beach. It was collected with extra care to ensure relevance to consumer interest and originality of video content without post-editing. Consumer videos contain very diverse content and have much fewer textual tags and descriptions, which motivates the content analysis based on both acoustic and visual features. Similar with [9], we extract three types of features for each video by BoWs representation: (1) 5000-D space-time interest points (STIP) feature [50], (2) 5000-D scale-invariant feature transform (SIFT) feature and (3) 4000-D Mel-frequency cepstral coefficients (MFCC) feature.

(2) Kodak: Kodak extracts 5166 key frames from 1358 consumer video clips as its data sets. Among these key frames, 3590 key frames belonging to 22 concepts are annotated by students from Columbia University. We use all annotated key frames in our experiments for video concept detection. Each key frame is represented by 73-D edged

[9] http://www-nlpir.nist.gov/projects/tv2005/
[10] http://www.ee.columbia.edu/ln/dvmm/consumervideo/
[11] http://vision.standford.edu/Datasets/OlympicSports/
[12] http://crcv.ucf.edu/data/UCF_Sports_Action.php

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2018.2810286, IEEE Transactions on Knowledge and Data Engineering

TABLE 4
Computational time (CPU time in second) of different methods on COIL20 and CovType data sets.

Data	FisherScor	RFS	MCFS	TRCFS	S^2FS^2R	InfFS	ECFS	ISR
COIL20	0.03	0.03	1.48	0.35	6.75	40.67	4.93	3.03
CovType	6203	11258	15724	87507	175731	212369	65420	54085

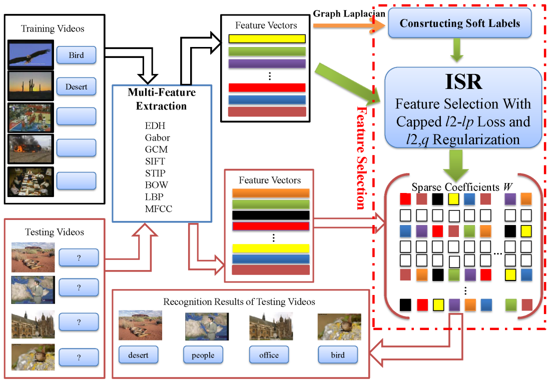


Fig. 5. The illustration of the general process for video semantic recognition by ISR. The red frame presents the core part of our algorithm which analyzes the feature space for practical applications.

direction histogram, 48-D Gabor and 225-D grid color moment, which are combined to be a 346-D vector of global features to represent each key frame.

(3) Olympic: Currently, Olympic contains 783 video sequences of athletes practicing different sports at a resolution of 640x480. All video sequences are obtained from YouTube and annotated by Amazon Mechanical Turk. It contains 16 types of sports actions, e.g. high jump, long jump, triple jump, bowling, tennis serve, platform diving, discus throw, and gymnastic vault. We extract 7830 key frames from all video sequences. Similar with [9], each key frame is represented by a 8500-D spatial pyramid BoWs feature.

(4) UCF: It consists of about 200 video sequences at a resolution of 720x480 collected from various sports which are typically featured on broadcast television channels such as the BBC and ESPN. The collection represents a natural pool of actions featured in a wide range of scenes and viewpoints. UCF includes 12 actions categories, like diving side, golf swinging back, kicking, lifting, running, skating, swinging bench and walking. It is very challenging for recognizing realistic actions from videos, due to large variations in camera motion, cluttered background and illumination conditions. We extract 7675 key frames from 200 video clips as UCF data sets. Similar with [9], each key frame is represented by a 10880-D spatial pyramid BoWs feature.

6.2 Framework and Evaluation Metric

To apply our method into video semantic recognition, we first give the main procedure in processing videos. As shown in Fig. 5, the main steps of ISR for VSR are (1) to represent the videos by different types of visual features, like SIFT and Bag of Words (BoW); (2) to enhance the label information of the training videos by manifold fitting on the data structure; (3) to learn the sparse coefficients $\mathbf{W}$ by ISR via sparse robust regression model and (4) to select more discriminative feature subset to recognize the semantic class of each testing video by obtained optimal feature subset.

Similar to [10], [20], [51], we adopt two metrics: Mean Average Precision (MAP) [10], [20] and Area Under Curve (AUC) [51] to evaluate the performance of ISR in video semantic recognition. Simultaneously, we also use microaveraging F_1-Score ($MicroF_1$) [8] to evaluate the stability and discriminating capability of ISR. Different with the traditional classification problem, one video may have multiple concepts or semantic categories in video semantic recognition. Therefore, we use $MicroF_1$ as the evaluation metric.

In our experiments, we randomly sample 60% unlabeled data points and different ratio of labeled samples as the training sets according to the size of data sets. The remaining samples are used as the corresponding testing sets. Following [10], the linear SVM is a better classifier for human action recognition, especially for the BoW histogram representations. Therefore, we utilize linear SVM as the classifier. The sampling processes were repeated fifty times to generate 50 random training/testing partitions, and then the average performance of fifty-round repetitions is reported.

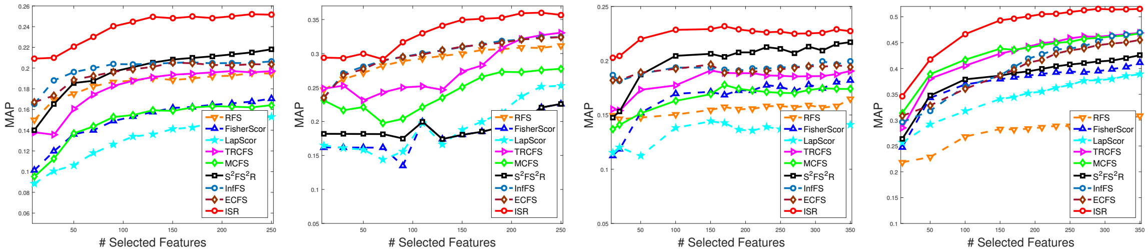
6.3 Video Semantic Recognition Results

6.3.1 Performance of Semi-supervised Feature Selection

To investigate the performance of semi-supervised feature selection deeply, we set the ratio of labeled training videos in the sampled training videos to 1% on CCV, Kodak, Olympic and UCF. And we set the number of selected features to different values of $\{10, 30, 50, 70, 90, 110, 130, 150, 170, 190, 210, 230, 250\}$ on CCV and Kodak, $\{10, 50, 100, 150, 170, 190, 210, 230, 250, 270, 290, 310, 330, 350\}$ on Olympic and UCF. Once the index $\{r_1, r_2, \cdots, r_s\}$ of the selected features is obtained, we train a classifier based on the selected features of the training videos.

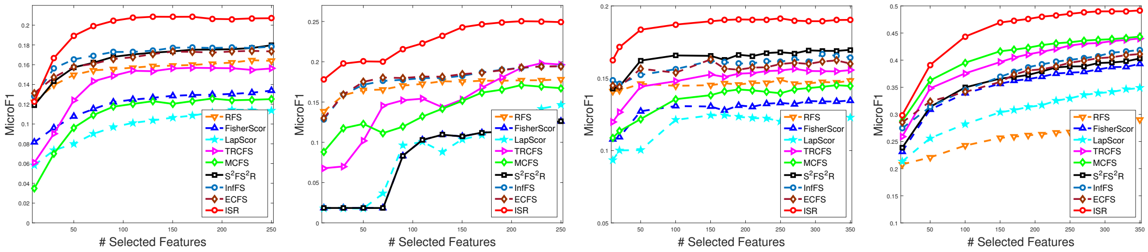
Fig. 6, Fig. 7 and Fig. 8 present the performance (MAP, $MicroF_1$ and AUC) of video semantic recognition of different feature selection methods. We observe that 1) When the number of selected features increases, ISR has a better performance in most cases; 2) Compared with the unsupervised and supervised feature selection methods, ISR has competitive or better performance than that of Laplacian score, Fisher score, RFS, MCFS, InfFS and ECFS, by the preservation of local geometry structure of unlabeled videos via graph Laplacian and enhancing the label information via label propagation; 3) Compared with the semi-supervised methods, TRCFS and S^2FS^2R, ISR outperforms other algorithms. Especially when the number of selected features is very small, ISR outperforms all the compared methods.

Interestingly, although MAP, $MicroF_1$ and AUC show the same trends in most cases, there maybe some slight differences between them. For example, on Olympic data with s = 50, S^2FS^2R has the larger MAP and $MicroF_1$ than RFS while RFS achieves the larger AUC value than S^2FS^2R. This may be caused by the fact that they evaluate the performances of action recognition in different aspects. Comprehensively considering the results of MAP, $MicroF_1$ and AUC on four video data sets, the performance of ISR is the best in most of cases.



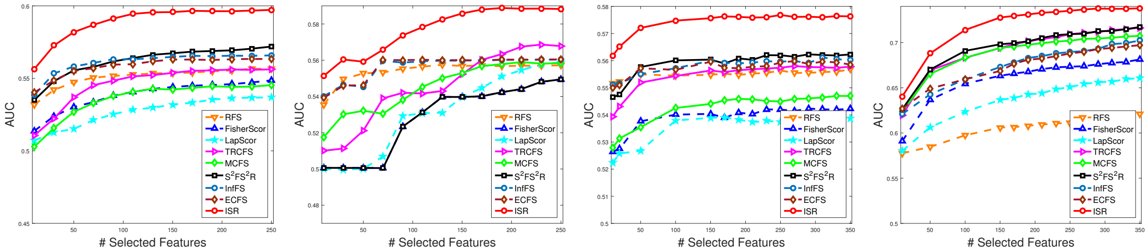
(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 6. The MAP scores of ISR and the state-of-the-art methods with different numbers of selected features on four video data sets.



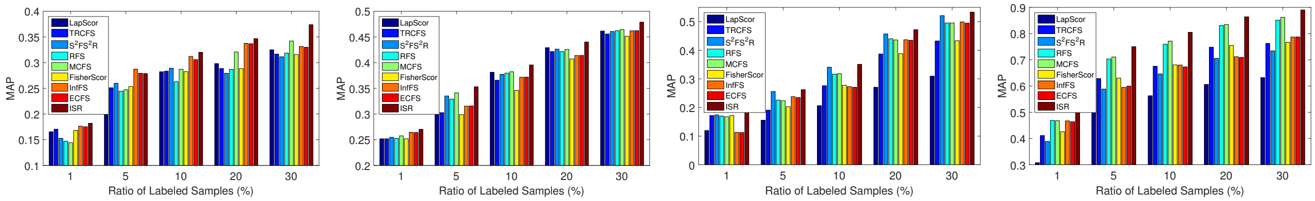
(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 7. The $MicroF_1$ scores of ISR and the state-of-the-art methods with different numbers of selected features on four video data sets.



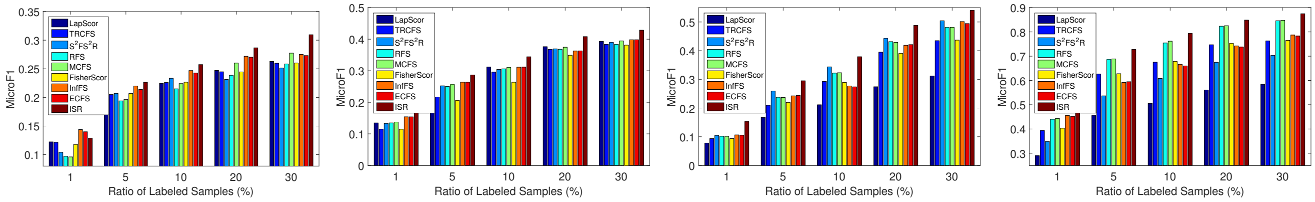
(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 8. The AUC scores of ISR and the state-of-the-art methods with different numbers of selected features on four video data sets.



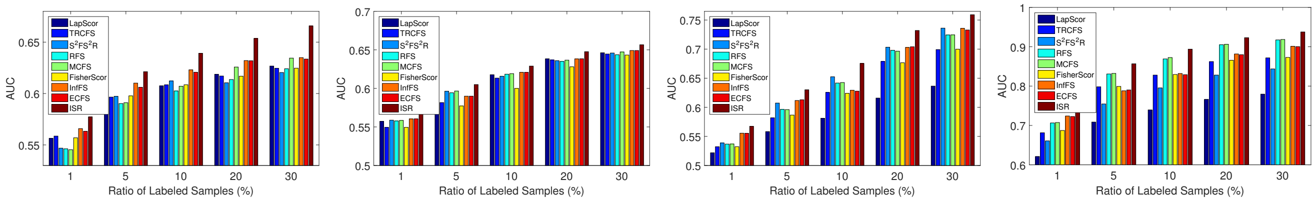
(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 9. The MAP scores of ISR and the state-of-the-art methods with different ratios of labeled samples on four video data sets.



(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 10. The $MicroF_1$ scores of ISR and the state-of-the-art methods with different ratios of labeled samples on four video data sets.



(a) CCV (b) Kodak (c) Olympic (d) UCF

Fig. 11. The AUC scores of ISR and the state-of-the-art methods with different ratios of labeled samples on four video data sets.

6.3.2 Performance of Different Ratios of Labeled Videos

In this section, we investigate the performance of ISR compared with the state-of-the-art methods for video semantic recognition. To show the impacts of different ratios of labeled training videos for semi-supervised methods, we report results when the ratios of labeled training videos are set to 1% , 5%, 10%, 20% and 30%. And we fixed the number of selected features to {10, 50, 100, 150, 170, 190, 210, 230, 250, 270, 290, 310, 330, 350}. We evaluate the performance of feature selection algorithms in video semantic recognition applications by computing the average MAP, $MicroF_1$ and AUC values of video semantic recognition when the range of the number of selected features changes between 10 and 350. Fig. 9, Fig. 10 and Fig. 11 present the results of MAP, $MicroF_1$ and AUC values with different ratios of labeled videos in training set using linear SVM classifier.

From the results of Fig. 9, Fig. 10 and Fig. 11, we observe that: 1) the proposed framework of ISR outperforms the state-of-the-art methods for different settings of the ratio of labeled training videos; 2) As the number of labeled training samples increases, the performance of feature selection increases. When the number of labeled videos is very few, for example, 1% on CCV and UCF, ISR has better performance than other compared methods. On CCV, MAP, $MicroF_1$ and AUC of ISR are at least 4.2%, 3.7% and 1.9% higher than that of others. 3) Similar to the results in Section 6.3.1, the evaluation criterions, MAP, $MicroF_1$ and AUC, show the same trend. 4) In general, the performance of semi-supervised feature selection methods are better than that of supervised and unsupervised algorithms.

7 Conclusion

In this paper, to select the most discriminative features, we propose a new semi-supervised feature selection method named ISR, by using a few labeled instances. After enlarging the label information of training data, we design capped l_2-l_p-norm loss to make ISR robust to the false labeled samples or outliers. To select the most important features by ranking scores, $l_{2,q}$-norm regularization is imposed into ISR to preserve the structural sparsity. Various experiments verify the effectiveness of our ISR. Finally, we have also applied ISR to video semantic recognition.

One of our future work is to find a more effective way to solve our non-convex formulation. There are many methods to tackle the non-convex optimization problem, such as linear local approximation and block gradient decent. Moreover, the converged speed of different methods is various. As videos include various features, another future work is to further investigate the heterogeneity among these features. We also want to apply ISR to other video applications, e.g. action recognition and video retrieval.

References

[1] B. Moore, “Principal component analysis in linear systems: Controllability, observability, and model reduction,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 17–32, 1981.

[2] X. He and P. Niyogi, “Locality preserving projections,” in *Advances in neural information processing systems*, vol. 16, 2004, pp. 153–160.

[3] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, “Joint embedding learning and sparse regression: A framework for unsupervised feature selection,” *IEEE Transactions on Cybernetics*, vol. 44, no. 6, pp. 793–804, 2014.

[4] X. Zhu, Z. Huang, Y. Yang, H. T. Shen, C. Xu, and J. Luo, “Self-taught dimensionality reduction on the high-dimensional small-sized data,” *Pattern Recognition*, vol. 46, no. 1, pp. 215–229, 2013.

[5] S. Bucak, R. Jin, and A. Jain, “Multiple kernel learning for visual object recognition: A review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1354–1369, July 2014.

[6] X. Li, D. Hu, and F. Nie, “Large graph hashing with spectral rotation.” in *AAAI*, 2017, pp. 2203–2209.

[7] X. Zhu, X. Li, and S. Zhang, “Block-row sparse multiview multilabel learning for image classification,” *IEEE transactions on cybernetics*, vol. 46, no. 2, pp. 450–461, 2016.

[8] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe, and X. Zhou, “Semisupervised feature selection via spline regression for video semantic recognition,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–13, 2014.

[9] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui, “Consumer video understanding: A benchmark database and an evaluation of human and machine performance,” in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*. ACM, 2011, p. 29.

[10] S. Wang, Y. Yang, Z. Ma, X. Li, C. Pang, and A. G. Hauptmann, “Action recognition by exploring data distribution and feature correlation,” in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1370–1377.

[11] J. G. Dy, C. E. Brodley, A. Kak, L. S. Broderick, and A. M. Aisen, “Unsupervised feature selection applied to content-based retrieval of lung images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 3, pp. 373–378, 2003.

[12] X. Cai, F. Nie, and H. Huang, “Exact top-k feature selection via l2,0-norm constraint,” in *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*. AAAI Press, 2013, Conference Proceedings, pp. 1240–1246.

[13] F. Nie, H. Huang, X. Cai, and C. H. Ding, “Efficient and robust feature selection via joint l2, 1-norms minimization,” in *Advances in Neural Information Processing Systems*, 2010, pp. 1813–1821.

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

[15] J. Chen and Y. Liu, “Locally linear embedding: a survey,” *Artificial Intelligence Review*, vol. 36, no. 1, pp. 29–48, 2011.

[16] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” *Advances Feature subset selection and ranking for data dimensionality reduction in neural information processing systems*, vol. 14, pp. 585–591, 2001.

[17] X. He, D. Cai, S. Yan, and H.-J. Zhang, “Neighborhood preserving embedding,” in *Tenth IEEE International Conference on Computer Vision*, vol. 2, 2005, Conference Proceedings, pp. 1208 – 1213.

[18] F. Nie, G. Cai, and X. Li, “Multi-view clustering and semi-supervised classification with adaptive neighbours.” in *AAAI*, 2017, pp. 2408–2414.

[19] Y. Liu, F. Nie, J. Wu, and L. Chen, “Efficient semi-supervised feature selection with noise insensitive trace ratio criterion,” *Neurocomputing*, vol. 105, pp. 12–18, 2013.

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

[21] H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[22] G. Roffo, S. Melzi, and M. Cristani, “Infinite feature selection,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 4202–4210.

[23] X. Zhu, X. Li, S. Zhang, C. Ju, and X. Wu, “Robust joint graph sparse coding for unsupervised spectral feature selection,” *IEEE transactions on neural networks and learning systems*, vol. 28, no. 6, pp. 1263–1275, 2017.

[24] Z. Zhao and H. Liu, “Semi-supervised feature selection via spectral analysis.” in *SIAM International Conference on Data Mining*. SIAM, 2007, pp. 641–646.

[25] Z. Xu, I. King, M.-T. Lyu, and R. Jin, “Discriminative semi-supervised feature selection via manifold regularization,” *IEEE Transactions on Neural Networks*, vol. 21, no. 7, pp. 1033–1047, 2010.

[26] X. Chang, F. Nie, Y. Yang, and H. Huang, “A convex formulation for semi-supervised multi-label feature selection.” in *AAAI*, 2014, pp. 1171–1177.

[27] Y. Yang, Z. Ma, A. G. Hauptmann, and N. Sebe, “Feature selection for multimedia analysis by sharing information among multiple tasks,” *IEEE Transactions on Multimedia*, vol. 15, no. 3, pp. 661–669, 2013.

[28] G. Roffo, “Ranking to learn and learning to rank: On the role of ranking in pattern recognition applications,” *arXiv preprint arXiv:1706.05933*, 2017.

[29] X. He, D. Cai, and P. Niyogi, “Laplacian score for feature selection,” in *Advances in Neural Information Processing Systems*, 2005, pp. 507–514.

[30] D. Cai, C. Zhang, and X. He, “Unsupervised feature selection for multi-cluster data,” in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*. ACM, 2010, pp. 333–342.

[31] C. M. Bishop *et al.*, *Neural networks for pattern recognition*. Clarendon press Oxford, 1995.

[32] Z. Zhao and H. Liu, “Spectral feature selection for supervised and unsupervised learning,” in *Proceedings of the 24th International Conference on Machine Learning*. ACM, 2007, pp. 1151–1157.

[33] G. Roffo and S. Melzi, “Features selection via eigenvector centrality,” in *Proceedings of New Frontiers in Mining Complex Patterns (NFMCP 2016)*, Oct 2016.

[34] X. Zhu, “Semi-supervised learning literature survey,” *Computer Science, University of Wisconsin-Madison*, vol. 2, p. 3, 2006.

[35] J. Zhao, K. Lu, and X. He, “Locality sensitive semi-supervised feature selection,” *Neurocomputing*, vol. 71, no. 10, pp. 1842–1849, 2008.

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

[37] F. Nie, S. Xiang, Y. Liu, and C. Zhang, “A general graph-based semi-supervised learning with novel class discovery,” *Neural Computing and Applications*, vol. 19, no. 4, pp. 549–555, 2010.

[38] X. Zhu, Z. Ghahramani, J. Lafferty *et al.*, “Semi-supervised learning using gaussian fields and harmonic functions,” in *ICML*, vol. 3, 2003, pp. 912–919.

[39] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, “Learning with local and global consistency,” *Advances in Neural Information Processing Systems*, vol. 16, no. 16, pp. 321–328, 2004.

[40] F. Nie, S. Xiang, Y. Jia, and C. Zhang, “Semi-supervised orthogonal discriminant analysis via label propagation,” *Pattern Recognition*, vol. 42, no. 11, pp. 2615–2627, 2009.

[41] R. Jenatton, J.-Y. Audibert, and F. Bach, “Structured variable selection with sparsity-inducing norms,” *The Journal of Machine Learning Research*, vol. 12, pp. 2777–2824, 2011.

[42] R. Chartrand, “Exact reconstruction of sparse signals via nonconvex minimization,” *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 707–710, 2007.

[43] P. Gong, J. Ye, and C.-s. Zhang, “Multi-stage multi-task feature learning,” in *Advances in Neural Information Processing Systems*, 2012, pp. 1988–1996.

[44] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien, “Lp-norm multiple kernel learning,” *Journal of Machine Learning Research*, vol. 12, pp. 953–997, 2011.

[45] J. M. Borwein and A. S. Lewis, *Convex analysis and nonlinear optimization: theory and examples*. Springer Science & Business Media, 2010.

[46] F. Nie, J. Yuan, and H. Huang, “Optimal mean robust principal component analysis,” in *Proceedings of the 31st International Conference on Machine Learning*. ICML, 2014, pp. 1062–1070.

[47] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

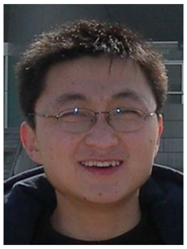
[48] M. Kloft, U. Brefeld, P. Laskov, K.-R. Müller, A. Zien, and S. Sonnenburg, “Efficient and accurate ℓ_p-norm multiple kernel learning,” in *Advances in Neural Information Processing Systems*, 2009, pp. 997–1005.

[49] M. Kloft and G. Blanchard, “On the convergence rate of lp-norm multiple kernel learning,” *Journal of Machine Learning Research*, vol. 13, no. Aug, pp. 2465–2502, 2012.

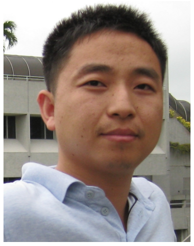
[50] I. Laptev, “On space-time interest points,” *Int. J. Comput. Vision*, vol. 64, no. 2-3, pp. 107–123, Sep. 2005.

[51] T. Fawcett, “An introduction to roc analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

Tingjin Luo is a PhD candidate with the College of Science at the National University of Defense Technology, Changsha, China. He received his Master degree and B.S. degree with the College of Information System and Management at the same university in 2013 and 2011, respectively. His research interests include machine learning, multimedia analysis and computer vision.



Chenping HOU (M’12) received his B.S. degree and the Ph.D. degree both in Applied Mathematics from the National University of Defense Technology, Changsha, China in 2004 and 2009, respectively. He is now an associate professor of College of Science in NUDT. He has published several papers in the following journals and conferences: TNNLS/TNN, TSMCB, TIP, Pattern Recognition, IJCAI. He is also a member of IEEE and ACM. His research interests include pattern recognition, machine learning, data mining, and computer vision.



Feiping NIE received the Ph.D. degree in Computer Science from Tsinghua University in 2009. His research interests are machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing and information retrieval. He has published more than 100 papers in the following top journals and conferences: TPAMI, IJCV, TIP, TNNLS/TNN, TKDE, TKDD, TMM, TSMCB/TC, Machine Learning, Pattern Recognition, Medical Image Analysis, Bioinformatics, ICML, NIPS, KDD, IJCAI, AAAI, ICCV, CVPR and SIGIR. He is now serving as Associate Editor or PC member for several prestigious journals and conferences in the related fields.

Hong TAO is a PhD candidate with the College of Science at the National University of Defense Technology, Changsha, China. She earned her B.S. degree from the same university in 2008. Her research interests include machine learning, systems science and data mining.

Dongyun YI is Professor of College of Science at the National University of Defense Technology. He earned his B.S. degree from Nankai University and the M.S. and Ph.D. degrees from National University of Defense Technology in Changsha, China, respectively. He has worked as a visiting researcher at the University of Warwick in 2008. His research interests include statistics, systems science and data mining.