



Multi-class Semi-supervised Logistic I-RELIEF Feature Selection Based on Nearest Neighbor

Baige Tang¹ and Li Zhang^{1,2}(✉)

¹ School of Computer Science and Technology,
Joint International Research Laboratory of Machine Learning
and Neuromorphic Computing, Soochow University, Suzhou 215006, Jiangsu, China
bgtang@stu.suda.edu.cn

² Provincial Key Laboratory for Computer Information Processing Technology,
Soochow University, Suzhou 215006, Jiangsu, China
zhangliml@suda.edu.cn

Abstract. The multi-class semi-supervised logistic I-RELEIF (MSLIR) algorithm has been proposed and showed its feature selection ability using both labeled and unlabeled samples. Unfortunately, MSLIR is poor when predicting labels for unlabeled samples. To solve this issue, this paper presents a novel multi-class semi-supervised logistic I-RELEIF based on nearest neighbor (MSLIR-NN) for multi-class feature selection tasks. To generate better margin vectors for unlabeled samples, MSLIR-NN uses the nearest neighbor scheme to first predict the labels of unlabeled samples and then calculates their margin vectors according to these estimated labels. Experimental results demonstrate that MSLIR-NN can improve the prediction accuracy of unlabeled data.

Keywords: Logistic I-RELIEF · Feature selection ·
Multi-class classification · Semi-supervised · Nearest neighbor

1 Introduction

In many fields such as data mining and machine learning, we usually need to deal with high-dimensional data which may contain a large number of irrelevant and redundant features. These features would lead to the sparsity of data distribution in the feature space and be a hindrance to data analysis tasks. The rapid growth of data dimension not only increases the computational cost and memory consumption, but also affects the classification performance of classifiers. In order to improve learning performance, a variety of data dimensionality reduction methods have been produced, among which feature selection is one of the most effective techniques for processing high-dimensional data [1, 2].

The main goal of feature selection is to select an optimal feature subset, which contains most useful information in original features and has the greatest correlation with classification tasks. Based on the optimal feature subset, the training

time of classifiers can be effectively shorten, and the learning performance could be enhanced further [3–5]. In recent years, the technology of feature selection is of diversity. A lot of feature methods have been proposed [9–11, 15, 17, 18]. Here, we consider RELIEF-based methods.

Kira and Rendell proposed the famous Relief algorithm in 1992, which uses the Euclidean distance as a metric to select features with great weights [6]. In 1994, Kononenko presented the extended algorithm RELIEF-F to solve the problem of multi-class classification [7]. On the basis of RELIEF, Sun et al. proposed an iterative RELEIF (I-RELIEF) algorithm to alleviate the deficiencies of RELEIF by exploring the framework of the expectation-maximization algorithm [8]. In order to better estimate feature weights, Sun et al. also proposed a logistic I-RELIEF (LIR) algorithm, which optimizes I-RELIEF in the form of logistic regression [16]. All of the above RELIEF-based algorithms are supervised feature selection ones, which can only use data with class labels. However, these methods cannot have a good performance when there exist few labeled data and a large number of unlabeled data. To remedy it, a semi-supervised logistic I-RELEIF (SLIR) method was presented [16]. In SLIR, both labeled and unlabeled data are used to calculate margin vectors of samples. However, SLIR was designed only for binary classification tasks. Tang et al. developed a multi-class semi-supervised logistic I-RELIEF (MSLIR) algorithm [19]. MSLIR designs a novel scheme to find margin vectors of unlabeled samples by calculating all possible candidate margin vectors and picking an optimal one under the condition of current feature weights. However, although MSLIR can implement multi-class feature selection in semi-supervised learning and get better classification performance than LIR, the supervised learning method, the prediction performance of unlabeled samples is unsatisfactory.

In order to solve the above issue, we propose a multi-class semi-supervised logistic I-RELEIF based on nearest neighbor (MSLIR-NN) for multi-class feature selection. In MSLIR-NN, the nearest neighbor scheme is adopted to assign pseudo labels to unlabeled samples according to labeled data in each iteration. In this case, unlabeled samples with pseudo labels could be treated as labeled ones. Thus, the margin vectors of labeled and unlabeled samples could be calculated easily. MSLIR-NN has a smaller computational complexity than MSLIR when calculating margin vectors of unlabeled samples. In experiments, support vector machine (SVM) and nearest neighbor (NN) classifiers are used to ensure the fairness of the classification results, respectively. Experimental results show that MSLIR-NN greatly improves the prediction performance of unlabeled data and enhances the performance of classifiers.

The rest part of this paper is organized as follows. The proposed method is described in detail in Sect. 2. The connections of MSLIR-NN to other related work are also discussed. Section 3 gives and analyzes experimental results. Section 4 concludes this paper.

2 Proposed Method: MSLIR-NN

In this section, we design a novel multi-class semi-supervised feature selection method, MSLIR-NN which adopts the nearest neighbor scheme to assign pseudo labels to unlabeled samples in each iteration. In doing so, unlabeled samples with pseudo labels could be treated as labeled ones. Thus, the margin vectors of labeled and unlabeled samples could be calculated easily. In the following, we describe MSLIR-NN in detail and discuss its connections to MSLIR and SLIR.

2.1 Margin Vectors

Assume that there is a labeled sample set $D_l = \{(\mathbf{x}_i^l, y_i^l)\}_{i=1}^L$ and an unlabeled sample set $D_u = \{\mathbf{x}_i^u\}_{i=1}^U$, where $\mathbf{x}_i^l \in \mathbb{R}^I$, $y_i^l \in \{1, 2, \dots, c\}$, $\mathbf{x}_i^u \in \mathbb{R}^I$, I is the number of original features, c is the class number, L and U represent the number of labeled and unlabeled samples, respectively. Generally, $L \ll U$.

It is well known that one of main differences of RELIEF-based methods is the way of calculating margin vectors of samples. Without loss of generality, let \mathbf{z}_i^l and \mathbf{z}_i^u be margin vectors of the labeled sample \mathbf{x}_i^l and the unlabeled sample \mathbf{x}_i^u , respectively. It is easy to generate the margin vectors of labeled samples in semi-supervised RELIEF-based methods. Similar to MSRIL [19], the margin vector \mathbf{z}_i^l of the labeled sample \mathbf{x}_i^l can be expressed as follows:

$$\begin{aligned} \mathbf{z}_i^l = & \sum_{\mathbf{x}_k^l \in M_i} P(\mathbf{x}_k^l = NM(\mathbf{x}_i^l)|\mathbf{w})|\mathbf{x}_i^l - \mathbf{x}_k^l| \\ & - \sum_{\mathbf{x}_k^l \in H_i} P(\mathbf{x}_k^l = NH(\mathbf{x}_i^l)|\mathbf{w})|\mathbf{x}_i^l - \mathbf{x}_k^l| \end{aligned} \quad (1)$$

where \mathbf{w} is the feature weight vector, the set $M_i = \{\mathbf{x}_k^l | (\mathbf{x}_k^l, y_k^l) \in D_l, y_i^l \neq y_k^l, k = 1, \dots, L, y_k^l \in \{1, \dots, c\}\}$ contains all labeled samples that have different labels from \mathbf{x}_i^l , the set $H_i = \{\mathbf{x}_k^l | (\mathbf{x}_k^l, y_k^l) \in D_l, y_i^l = y_k^l, k = 1, \dots, L, y_k^l \in \{1, \dots, c\}\}$ contains all labeled samples that have the same label as \mathbf{x}_i^l , $P(\mathbf{x}_k^l = NM(\mathbf{x}_i^l)|\mathbf{w})$ and $P(\mathbf{x}_k^l = NH(\mathbf{x}_i^l)|\mathbf{w})$ are the probabilities that the sample \mathbf{x}_k^l is the nearest miss and the nearest hit of \mathbf{x}_i^l , respectively, $NM(\mathbf{x}_i^l)$ represents the nearest miss (the nearest neighbor of sample \mathbf{x}_i^l from a different class) of \mathbf{x}_i^l , and $NH(\mathbf{x}_i^l)$ the nearest hit (the nearest neighbor of sample \mathbf{x}_i^l from the same class) of \mathbf{x}_i^l .

For semi-supervised RELIEF-based methods, it is the key and difficulty that how to define the margin vectors of unlabeled samples. Before calculating them, we first predict the pseudo labels of unlabeled data. According to the information contained in the labeled set D_l , we use the nearest neighbor scheme to predict the pseudo labels of unlabeled data in the set D_u . Note that \mathbf{w} changes as iterations. For any unlabeled sample, its neighborhood is metabolic under different weight conditions. In other words, \mathbf{w} has an effect on the procedure of searching nearest neighbors. Thus, we search nearest neighbors of unlabeled samples in the weighted feature space instead of the original input space. Then, we extend

D_u to the set $\hat{D}_u = \{(\mathbf{x}_i^u, \hat{y}_i^u)\}_{i=1}^U$ with pseudo labels \hat{y}_i^u for \mathbf{x}_i^u . Similar to (1), we define the margin vector \mathbf{z}_i^u of the unlabeled sample \mathbf{x}_i^u :

$$\begin{aligned} \mathbf{z}_i^u &= \sum_{\mathbf{x}_k^l \in M'_i} P(\mathbf{x}_k^l = NM(\mathbf{x}_i^u) | \mathbf{w}) |\mathbf{x}_i^u - \mathbf{x}_k^l| \\ &\quad - \sum_{\mathbf{x}_k^l \in H'_i} P(\mathbf{x}_k^l = NH(\mathbf{x}_i^u) | \mathbf{w}) |\mathbf{x}_i^u - \mathbf{x}_k^l| \end{aligned} \tag{2}$$

where the set $M'_i = \{\mathbf{x}_k^l | (\mathbf{x}_k^l, y_k^l) \in D_l, \hat{y}_i^u \neq y_k^l, k = 1, \dots, L, y_k^l \in \{1, \dots, c\}\}$ contains all labeled samples that have different labels from \mathbf{x}_i^u , and the set $H'_i = \{\mathbf{x}_k^l | (\mathbf{x}_k^l, y_k^l) \in D_l, \hat{y}_i^u = y_k^l, k = 1, \dots, L, y_k^l \in \{1, \dots, c\}\}$ contains all labeled samples that have the same label as \mathbf{x}_i^u .

2.2 Optimization Problem

After obtaining margin vectors of all samples, the optimization of MSLIR-NN can be described as:

$$\begin{aligned} \min_{\mathbf{w}} \|\mathbf{w}\|_1 + \alpha \sum_{i=1}^L \log(1 + \exp(-\mathbf{w}^T \mathbf{z}_i^l)) + \beta \sum_{i=1}^U \log(1 + \exp(-\mathbf{w}^T \mathbf{z}_i^u)) \tag{3} \\ \text{s.t. } \mathbf{w} \geq 0 \end{aligned}$$

where $\|\cdot\|_1$ is the 1-norm, the regularization parameters $\alpha \geq 0$ and $\beta \geq 0$ control the importance of labeled and unlabeled samples, respectively.

To eliminate the constraint of $\mathbf{w} \geq 0$, let $\mathbf{w} = [v_1^2, \dots, v_I^2]^T$ and $\mathbf{v} = [v_1, \dots, v_I]$. Substituting \mathbf{v} into (3), we can make it to an unconstraint optimization problem and have

$$\begin{aligned} \min_{\mathbf{v}} J = \|\mathbf{v}\|_2^2 + \alpha \sum_{i=1}^L \log(1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^l)) \\ + \beta \sum_{i=1}^U \log(1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^u)) \end{aligned} \tag{4}$$

where $\mathbf{z}_i^l = [z_{i1}^l, \dots, z_{iI}^l]$ and $\mathbf{z}_i^u = [z_{i1}^u, \dots, z_{iI}^u]$. (4) can be solved by using the gradient descent method. The derivation of J to \mathbf{v} can be written as follows:

$$\begin{aligned} \frac{\partial J}{\partial v_k} &= 2v_k - \alpha \sum_{i=1}^L \frac{\exp(-\sum_{d=1}^I v_d^2 z_{id}^l)(2v_k z_{ik}^l)}{1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^l)} \\ &\quad - \beta \sum_{i=1}^U \frac{\exp(-\sum_{d=1}^I v_d^2 z_{id}^u)(2v_k z_{ik}^u)}{1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^u)} \end{aligned} \tag{5}$$

Let

$$Q = \alpha \sum_{i=1}^L \frac{\exp(-\sum_{d=1}^I v_d^2 z_{id}^l)(v_k z_{ik}^l)}{1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^l)} + \beta \sum_{i=1}^U \frac{\exp(-\sum_{d=1}^I v_d^2 z_{id}^u)(v_k z_{ik}^u)}{1 + \exp(-\sum_{d=1}^I v_d^2 z_{id}^u)}$$

Then we can update v_k by

$$v_k \leftarrow v_k - \eta(v_k - Q) \quad (6)$$

where $\eta > 0$ is the learning rate.

2.3 Algorithm and Complexity Analysis

The algorithm description of MSLIR-NN is shown in Algorithm 1. Given a labeled dataset D_l and an unlabeled dataset D_u , the weight vector \mathbf{w} is updated iteratively. Note that the weight vector in the t -th iteration is denoted as $\mathbf{w}_{(t-1)}$. First, under the current feature weights, MSLIR-NN computes the weighted labeled samples \mathbf{x}^{l*} and unlabeled \mathbf{x}^{u*} , that is: $\mathbf{x}^{l*} = \mathbf{x}^l \circ \mathbf{w}_{(t)}$ and $\mathbf{x}^{u*} = \mathbf{x}^u \circ \mathbf{w}_{(t-1)}$, where \circ denotes the element-by-element multiplication. The pseudo labels of unlabeled samples are determined in the weighted sample space using the NN scheme. Then, the margin vectors of \mathbf{x}^l and \mathbf{x}^u are calculated by (1) and (2), respectively. Finally, \mathbf{w} is obtained by solving (3). MSLIR-NN alternatively modifies the weight vector until convergence.

The computational complexity of MSLIR-NN mainly includes three parts: the calculation of margin vectors for labeled samples, the calculation of margin vectors for unlabeled samples, and the solution to the optimization problem (3). The computational complexity of calculating margin vectors for labeled samples is identical to that of MSLIR and SLIR, which is $O(dL^2)$ without considering the calculation of probability terms, where d is the dimension of samples, and L is the number of labeled samples. For calculating of margin vectors for unlabeled samples, the computational complexity in MSLIR-NN is about $O(dUL)$, where U is the number of unlabeled samples. For the last part, MSLIR-NN has the same computational complexity as MSLIR and SLIR.

2.4 Connections to Related Work

Four RELEIF-based methods, LIR, SLIR, MSLIR and MSLIR-NN use the logistic regression formulation to optimize the feature weight vector \mathbf{w} . We discuss the connections of MSLIR-NN to LIR, SLIR, MSLIR in the following.

MSLIR-NN is a semi-supervised learning method as well as SLIR and MSLIR, LIR is designed for supervised learning. Base on LIR, SLIR introduces a term about unlabelled samples into the objective function. MSLIR changes the objective function of SLIR, which makes a balance calculation between labeled and unlabeled samples. Although MSLIR-NN has the same optimization function as MSLIR, MSLIR-NN has a different way for computing margin vectors of unlabelled samples.

SLIR, MSLIR and MSLIR-NN all adopt the way of calculating margin vectors of labeled samples in LIR. It is intuitive for SLIR to get margin vectors of unlabeled samples since SLIR deals with only binary classification tasks. For a given unlabeled sample, MSLIR first calculates all possible candidate margin vectors and takes an optimal one in the weighted feature space as its margin

Algorithm 1. MSLIR-NN

Input: Labeled dataset $D_l = \{(\mathbf{x}_i^l, y_i^l)\}_{i=1}^L \subset R^I \times \{1, 2, \dots, c\}$; unlabeled dataset $D_u = \{\mathbf{x}_i^u\}_{i=1}^U \subset R^I$, regularization parameters α and β , the iteration number T , and the stop criterion θ .

Output: Feature weight \mathbf{w} .

```

1 begin
2   Initialization: Set  $\mathbf{w}_{(0)} = [1, 1, \dots, 1]^T$ ,  $t = 1$ , and  $\rho = 1 + \theta$ ;
3   while  $t \leq T$  &&  $\rho > \theta$  do
4     Compute the weighted samples  $\mathbf{x}_i^{l*}$  and  $\mathbf{x}_i^{u*}$  :  $\mathbf{x}_i^{l*} = \mathbf{x}_i^l \circ \mathbf{w}_{(t-1)}$  and
        $\mathbf{x}_i^{u*} = \mathbf{x}_i^u \circ \mathbf{w}_{(t-1)}$ , where  $\circ$  denotes the element-by-element
       multiplication;
5     Predict pseudo labels of unlabeled samples  $\mathbf{x}_i^{u*}$  by using weighted data,
        $i = 1, \dots, U$ ;
6     Compute  $\mathbf{z}_i^l$  by (1) and  $\mathbf{z}_i^u$  by (2);
7     Solve the optimization problem (3) using the gradient descent method
       to find  $\mathbf{v}$ ;
8     Compute  $\mathbf{w}_{(t)} = [v_1^2, \dots, v_I^2]^T$ ;
9     Let  $\rho = \|\mathbf{w}_{(t)} - \mathbf{w}_{(t-1)}\|$ ;
10     $t = t + 1$ ;
11  end
12   $\mathbf{w} = \mathbf{w}_{(t)}$ ;
13  Return  $\mathbf{w}$ .
14 end

```

vector. The computational complexity of MSLIR is $O(cdLU)$ when calculating margin vectors of unlabeled samples. MSLIR-NN first assigns a pseudo label for the unlabeled sample and then directly calculate its margin vector as label samples. Compared to MSLIR, MSLIR-NN has a lower complexity, or $O(dLU)$, which is independent of the class number c and identical to SLIR.

3 Experiments

We conduct extensive experiments to demonstrate the efficiency and effectiveness of MSLIR-NN. Ten UCI datasets [20] including Pendigits, Satimage, Waveform, Wine, Vehicle, Iris, Breast, Heart, Wdbc and Pima are adopted, where the first six datasets are multi-class, and the rest four ones are binary. All datasets are randomly divided into training and test subsets, and the training subsets contain labeled and unlabeled samples. A brief description of datasets is listed in Table 1, where “#Training” and “#Test” represent the number of training and test samples, respectively, “#Labeled” and “#Unlabeled” are the number of labeled and unlabeled samples in a training set, “#Feature” represents the dimension of samples, and “#Class” indicates the number of categories in datasets. For each dataset, we add 100 additional noise features which are independently Gaussian distributed. We normalize all features with the original data.

Table 1. Description of ten UCI datasets.

Data sets	#Training		#Test	#Feature	#Class
	#Labeled	#Unlabeled			
<i>Pendigits</i>	20	7474	3498	16(+100)	10
<i>Satimage</i>	30	4405	2000	36(+100)	6
<i>Waveform</i>	30	470	4500	21(+100)	3
<i>Wine</i>	10	40	128	13(+100)	3
<i>Vehicle</i>	20	180	646	18(+100)	4
<i>Iris</i>	10	40	100	4(+100)	3
<i>Breast</i>	20	179	500	9(+100)	2
<i>Heart</i>	40	130	133	13(+100)	2
<i>Wdbc</i>	20	149	400	30(+100)	2
<i>Pima</i>	30	158	580	8(+100)	2

In our experiments, the compared methods include RELIEF-F, LIR, SLIR, MSLIR and MSLIR-NN. Both RELIEF-F and LIR use only labeled data, while SLIR, MSLIR and MSLIR-NN use both labeled and unlabeled data. The classification performance is tested on the same test subsets. In LIR, the regularization parameter λ and learning rate η are 10 and 0.03, respectively. In MSLIR-NN, MSLIR and SLIR, the parameters α and β are 10 and 0.1, respectively, and the learning rate is the same as that of LIR.

To eliminate the effect of statistical error, each algorithm runs 10 times for each dataset, and takes the average result as the final one. In order to ensure the reliability of experimental results, the nearest neighbor (NN) and support vector machine (SVM) classifiers are used in experiments. Here the Gaussian kernel and regularization parameters in SVM are selected by the grid search method, where both vary from 2^{-10} to 2^{10} .

3.1 Experiments on Multi-class Datasets

For multi-classification tasks, we compare the proposed algorithm with other three methods MSLIR, LIR and RELIEF-F. Experiments are implemented on the Pendigits, Satimage, Waveform, Wine, Vehicle and Iris datasets. The performance of these feature selection algorithms are evaluated by the classification accuracy with selected features, and the final experimental results obtained by SVM are shown in Fig. 1. From Fig. 1, we can observe that the classification accuracy of MSLIR-NN is the best among compared methods, which indicates that features selected by MSLIR-NN have a greater correlation with the label information. In Figs. 1(a) and (c), it can be observed that the classification accuracy of the four algorithms increases as increasing the number of features, which demonstrates that the chosen features are all useful features for classification and the noisy features are excluded. In Figs. 1(b), (d), (e) and (f), the classification

accuracy tends to be steady or decreasing, which indicates that feature selection is useful. In other words, not all original features are related to classification tasks. For example, in the Wine dataset, when the number of selected features is eight, MSLIR-NN achieves the highest classification accuracy of 98%, and the last four selected features are likely to be irrelevant features.

The performance curves obtained by NN are shown in Fig. 2. From these figures, we can have similar conclusions as those from Fig. 1. We give the best average accuracies and the corresponding standard deviations of four methods in Table 2, where the best results are bolded. Compared with the other three methods, MSLIR-NN has a higher classification accuracy and smaller standard deviation, reflecting that our method has better stability than the previous MSLIR.

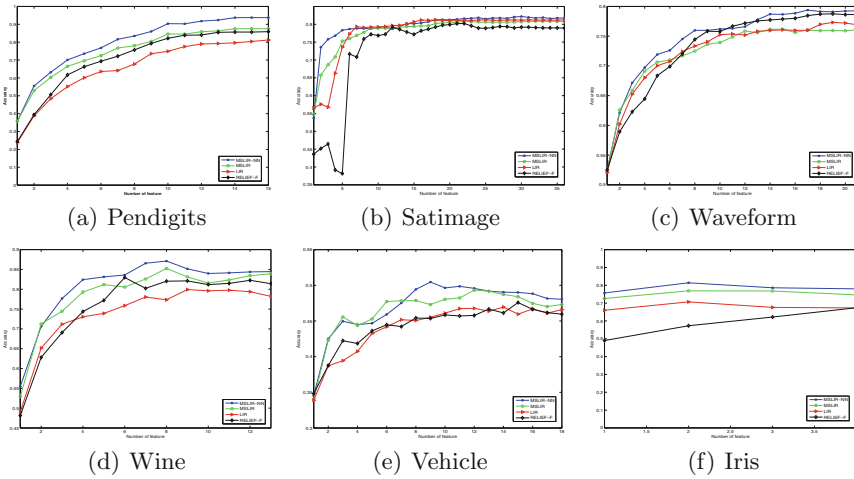


Fig. 1. Classification accuracy of SVM using four feature selection methods on multi-class datasets: (a) Pendigits, (b) Satimage, (c) Waveform, (d) Wine, (e) Vehicle and (f) Iris.

Table 2. Classification accuracy and standard deviations (%) of NN using four feature selection methods on multi-class datasets

Data sets	MSLIR-NN	MSLIR	LIR	RELIEF-F
<i>Pendigits</i>	97.74 ± 0.10	92.22 ± 17.45	90.05 ± 6.13	89.49 ± 4.92
<i>Satimage</i>	88.55 ± 0.00	79.18 ± 4.23	83.07 ± 1.82	84.21 ± 1.25
<i>Waveform</i>	70.85 ± 2.90	68.38 ± 2.81	67.17 ± 4.65	69.74 ± 3.26
<i>Wine</i>	87.11 ± 3.85	85.23 ± 5.37	79.92 ± 8.92	82.97 ± 9.21
<i>Vehicle</i>	51.35 ± 6.93	48.98 ± 7.41	45.74 ± 8.93	47.77 ± 6.99
<i>Iris</i>	92.30 ± 5.23	86.10 ± 19.91	79.70 ± 25.79	81.20 ± 23.88

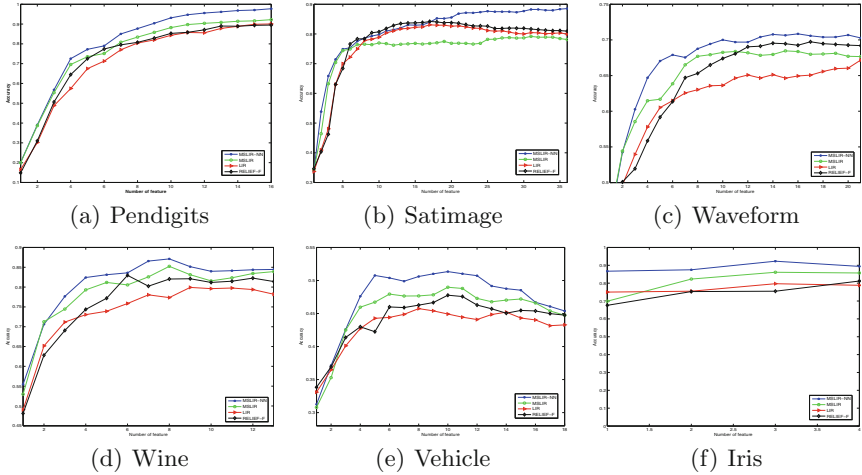


Fig. 2. Classification accuracy of NN using four feature selection methods on multi-class datasets: (a) Pendigits, (b) Satimage, (c) Waveform, (d) Wine, (e) Vehicle and (f) Iris.

3.2 Experiments on Binary Datasets

Similar to MSLIR, MSLIR-NN can also be applied to binary classification tasks. Since SLIR is only applicable to binary classification tasks, we compare MSLIR-NN and MSLIR with it on Breast, Heart, Wdbc and Pima datasets.

The experimental results obtained by SVM are given in Fig. 3. We can see that MSLIR-NN is much better than both MSLIR and SLIR, especially in Figs. 3(a), (c) and (d).

The performance curves obtained by NN are shown in Fig. 4. MSLIR-NN still has an advantage over other two methods. We list the best average classification accuracies and corresponding standard deviations in Table 3. Obviously, MSLIR-NN has the best performance among three methods on four binary datasets, followed by MSLIR. Compared to MSLIR, MSLIR-NN is improved 2.03% accuracy on Heart and 5.73% the accuracy on Pima, respectively.

Table 3. Classification accuracy and standard deviations (%) of NN using three feature selection methods on binary datasets

Data sets	MSLIR-NN	MSLIR	SLIR
<i>Breast</i>	94.74 \pm 1.12	93.86 \pm 1.88	93.00 \pm 3.48
<i>Heart</i>	74.51 \pm 4.48	72.48 \pm 4.12	69.02 \pm 6.70
<i>Wdbc</i>	87.43 \pm 3.65	87.18 \pm 2.58	84.30 \pm 7.34
<i>Pima</i>	67.20 \pm 3.37	61.47 \pm 3.72	60.81 \pm 5.24

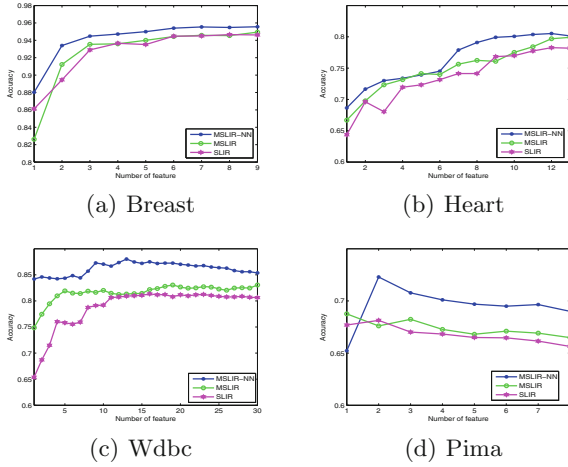


Fig. 3. Classification accuracy of SVM using four feature selection methods on binary datasets: (a) Breast, (b) Heart, (c) Wdbc, (d) Pima.

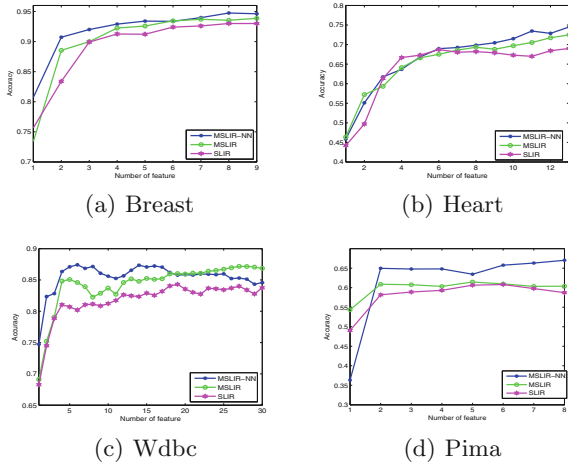


Fig. 4. Classification accuracy of NN using four feature selection methods on binary datasets: (a) Breast, (b) Heart, (c) Wdbc, (d) Pima.

3.3 Comparison of MSLIR-NN and MSLIR

MSLIR-NN and MSLIR have the same objective function, and different ways for constructing margin vectors of unlabeled samples. Here, we compare them in two aspects, the prediction ability on unlabeled samples in training subsets and the running time of feature selection.

Table 4 lists the accuracy on unlabeled samples and running time of feature selection for two methods. We can see that MSLIR-NN is significantly better than MSLIR on the prediction performance, which indicates that our new proposed

Table 4. Comparison of MSLIR-NN and MSLIR algorithms

Data sets	Accuracy (%)		Running time (sec.)	
	MSLIR-NN	MSLIR	MSLIR-NN	MSLIR
<i>Pendigits</i>	30.33	2.98	55.26	64.00
<i>Satimage</i>	48.57	2.77	9.86	31.70
<i>Waveform</i>	63.15	6.00	63.01	59.93
<i>Wine</i>	62.00	5.00	2.87	2.48
<i>Vehicle</i>	35.78	18.00	18.93	10.78
<i>Iris</i>	84.25	6.00	0.84	5.99
<i>Breast</i>	90.61	64.75	1.75	14.68
<i>Heart</i>	67.31	46.08	49.59	53.12
<i>Wdbc</i>	83.22	63.75	1.48	19.42
<i>Pima</i>	57.53	35.70	0.78	44.78

algorithm can use a few labeled data to predict the label of unlabeled data. Thus we can calculate the margin vectors of unlabeled samples more accurately, which can improve the stability of algorithm. The computational complexity of algorithms can be reflected by the running time of feature selection. Obviously, MSLIR-NN is much faster than MSLIR on all ten datasets, which supports our analysis about computational complexity in Sect. 2.3.

4 Conclusions

In this paper, we propose MSLIR-NN based on MSLIR for multi-class semi-supervised feature selection by introducing the nearest neighbor scheme. MSLIR-NN has a less complexity than MSLIR, and can improve the accuracy of label prediction for unlabeled data which contributes to the calculation way of margin vectors of unlabeled samples. Extensive experiments are performed on binary and multi-class classification tasks. Two classical classifiers NN and SVM are used to implement classification after feature selection has finished. On multi-class datasets, MSLIR-NN is superior to supervised methods LIR and RELEIF-F, and the semi-supervised method MSLIR. On the binary datasets, MSLIR-NN performs the best among three semi-supervised methods. In experiments of comparison with MSLIR, MSLIR-NN unfolds its ability in predicting labels of unlabeled samples and speedability. In general, MSLIR-NN can extract useful features and achieve better performance.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Grant No. 61373093, by the Soochow Scholar Project of Soochow University, and by the Six Talent Peak Project of Jiangsu Province of China.

References

1. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *J. Mach. Learn. Res.* **3**(6), 1157–1182 (2003)
2. Zhao, Z., Wang, L., Liu, H., Ye, J.: On similarity preserving feature selection. *IEEE Trans. Knowl. Data Eng.* **25**(3), 619–632 (2013)
3. Benabdeslem, K., Hindawi, M.: Efficient semi-supervised feature selection: constraint, relevance, and redundancy. *IEEE Trans. Knowl. Data Eng.* **26**(5), 1131–114326 (2014)
4. Zhang, D., Chen, S., Zhou, Z.H.: Constraint score: a new filter method for feature selection with pairwise constraints. *Pattern Recogn.* **41**(5), 1440–1451 (2008)
5. Sheikhpour, R., Sarram, M.A., Gharaghani, S., et al.: A survey on semi-supervised feature selection methods. *Pattern Recogn.* **64**(C), 141–158 (2016)
6. Kira, K., Rendell, L.A.: The feature selection problem: traditional methods and a new algorithm. In: Tenth National Conference on Artificial Intelligence, pp. 129–134 (1992)
7. Kononenko, I.: Estimating attributes: analysis and extensions of RELIEF. In: European Conference on Machine Learning on Machine Learning, pp. 171–182 (1994)
8. Sun, Y.: Iterative RELIEF for feature weighting: algorithms, theories, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 1035–1051 (2007)
9. Cheng, Z.D., Zhang, Y.J., Fan, X., Zhu, B.: Study on discriminant matrices of commonly used fisher discriminant functions. *Acta Autom. Sinica* **36**(10), 1361–1370 (2010)
10. Chen, L.F., Liao, H.Y.M., Ko, M.T., Lin, J.C., Yu, G.J.: A new LDA based face recognition system which can solve the small sample size problem. *Pattern Recogn.* **33**(10), 1713–1726 (2000)
11. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Comput. Soc.* **27**(8), 1226 (2005)
12. Mitra, P., Murthy, C.A., Pal, S.K.: Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(3), 301–312 (2002)
13. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: International Conference on Neural Information Processing Systems, vol. 18, pp. 507–514 (2005)
14. Bishop, C.M.: *Pattern Recognition and Machine Learning*. Springer, New York (2006)
15. Zeng, Z., Wang, X.D., Zhang, J., Wu, Q.: Semi-supervised feature selection based on local discriminative information. *Neurocomputing* **173**(P1), 102–109 (2016)
16. Cheng, Y., Cai, Y., Sun, Y., Li, J.: Semi-supervised feature selection under the Logistic I-RELIEF framework. In: International Conference on Pattern Recognition, pp. 1–4 (2008)
17. Zhao, Z., Liu, H.: Semi-supervised feature selection via spectral analysis. In: SIAM International Conference on Data Mining, SIAM 2007, pp. 641–646. SIAM, Minneapolis (2007)
18. Xu, J., Tang, B., He, H., Man, H.: Semi-supervised feature selection based on relevance and redundancy criteria. *IEEE Trans. Neural Netw. Learn. Syst.* **28**(9), 1974–1984 (2016)
19. Tang, B., Zhang, L.: Semi-supervised feature selection based on logistic I-RELIEF for multi-classification. In: Geng, X., Kang, B.-H. (eds.) PRICAI 2018. LNCS (LNAI), vol. 11012, pp. 719–731. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-97304-3_55
20. UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml/datasets.html>