

A Type-2 Fuzzy Logic Ensemble SVM Classifier based on Feature Weighting

Weiping Huang¹, Ziyang Wang¹, Qinghua Chi¹ and Jun Liang¹

¹State Key Lab of Industrial Control Technology Department of Control Science & Engineering, Zhejiang University, Hangzhou 310027, China
weipingh@126.com

Abstract

Support vector machine (SVM) has been gaining popularity in classification, for its structural risk minimization principle and the usage of kernel function. Traditional SVM as well as many improved algorithms treats all features in training data with equal weights, even though different features may have different impacts on the final classification. To overcome this issue, a new method which combined feature weighting with classification was proposed in this paper. Compared with other similar research, the major advantage of this method is that it brings no change to the algorithm structure. To further improve the performance of SVM classifier, a type-2 fuzzy logic based ensemble SVM was proposed. Type-2 fuzzy logic outperforms type-1 fuzzy logic greatly in terms of handling uncertainty in the ensemble process. Experiment results validated the effectiveness of proposed methods.

Keywords: support vector machine; feature weighting; type-2 fuzzy logic; ensemble classifier

1. Introduction

Support vector machine (SVM) has been widely used for various classification application with a high level of classification accuracy as well as great flexibility, especially for high-dimensional data classification. Since introduced by Cortes and Vapnik[1], much research has been conducted to improve the traditional SVM and make it feasible to different application. Lin and Wang applied a fuzzy membership to each input point of SVM and reformulated SVM into fuzzy SVM(FSVM), in order to attain the goal that different input points make different contribution to the learning of decision surface[2].Hui Xue et al. applied the structural information of data and developed a novel algorithm, termed as Structural Support Vector Machine (SSVM), by directly embedding the structural information into the SVM objective function[3]. An interval type-2 fuzzy weighted support vector machine (IT2FW-SVM)was proposed in [4], by applying type-2 fuzzy membership in the computation of penalty coefficient. However, traditional SVM and many improved SVM algorithms set the same weight on all features of the original data, even though different features may have different impacts on the final classification. Thus it is supposed to get more reasonable classification results after assigning different weights to different features according to their influence on the classification process.

Some research has been conducted trying to introduce feature weighting to classification or clustering [5-7]. However, those methods integrate feature weighting into the classification or clustering algorithms. In other words, the algorithms have to be changed, which seriously affects their expandability. Therefore it is desirable to find a method to adopt feature weighting in classification or clustering with least change to the algorithm.

To address this issue, the strategy of data scaling could be introduced. Data scaling is a necessary step before data classification. Part 2 of Sarle's Neural Networks FAQ[8] explains the importance of data scaling in neural networks and most of these considerations also apply to classification algorithm. The main purpose of data scaling is to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges, especially in some classification algorithms based on statistics distance[9]. Usually each attribute is scaled to the range $[-1, +1]$ or $[0,1]$.

Since main purpose of data scaling before classification is to limit the dominance of some attributes, it is supposed to improve the classifier's performance if the dominance relationship in training data was controlled consciously. Thus a new method was proposed in this paper which combined the advantages of feature weighting and data scaling. The weights of different features were calculated and multiplied to corresponding features of training data which has been scaled to the range $[-1;+1]$ or $[0;1]$. Thus, those features with larger effect on classification process were scaled to larger ranges, and less important features were scaled to smaller ranges or even zero. Actually, the original dataset was transformed for better classification without changing classification algorithm itself. To calculate different features' weights, two feature weighing algorithms, compactness and separation coefficient (CSC) algorithm and relief feature weighing algorithm were used.

Except for features contributing discriminatively in classification, how to select appropriate kernel functions and parameters is another practical difficulty, when applying SVMs to solve classification problems. One obvious way is to experiment with different kernels and parameters and then choose the one working best. Generally, it is time-consuming if the training data has a large scale. Ensemble classifier is a good way to solve this problem. SVMs with different kernels and parameters could be selected to construct an ensemble SVM. The resulting SVM is expected to outperform each of those single SVMs since different classifiers could realize mutual complementation.

Bagging [10] and Boosting [11] are two popular ensemble algorithms. Voting combination is a widely used approach of bagging, in which every single SVM's weights are identical. However it may not be desirable since different SVMs usually perform discriminatively and are supposed to have different weights in the ensemble SVM. To overcome the drawbacks of voting combination, type-1 fuzzy logic is introduced to the ensemble classifier [12], which assigns different weights to different single classifiers according to classification accuracies. Considering that the SVM classification accuracy is easily affected by outliers and noise in training data, thus there exists uncertainty in relationships between each single SVM's classification accuracy and its corresponding weight in the ensemble SVM. The type-1 fuzzy logic handles the uncertainty by using a crisp membership function (MF) and mapping the accuracy to a crisp value. However, once the MF is determined, the uncertainty disappears. Type-2 fuzzy logic outperforms type-1 fuzzy logic greatly in terms of handling uncertainty [13, 14]. In this paper, a new method which applied type-2 fuzzy logic in the ensemble algorithm is proposed. A type-2 fuzzy MF between accuracies and weights mapped an accuracy to a fuzzy set instead of a crisp value, but still contains uncertainty. A similar research has been carried out in [15], but it needs to construct a type-2 fuzzy logic system, which would increase the computational complexity in contrast with the method proposed in this paper.

The remainder of the paper is organized as follows. Section 2 introduces the new method combining feature weighting algorithms with classification. Section 3 proposes interval type-2 fuzzy logic and its extended approach. Section 4 describes the way to use interval type-2 fuzzy logic in the ensemble classifier. Section 5 is devoted to presenting the experiments and the results. Finally, Section 6 contains a short discussion and conclusion.

2. Feature Weighting Algorithms and Application

Feature weighting usually tries to measure the discriminating ability of a specific feature to distinguish the different class labels of the original data[16]. All the algorithms proposed for feature weighting could be divided into five categories: distance, information (or uncertainty), dependence, consistency, and classifier error rate[17]. As for distance, *e.g.*, in a binary classification problem, if feature X induces a greater difference between the two classes than feature Y , then it is considered that feature X is preferred to feature Y . Information measures typically mean the information gain from a feature. The information gain from a feature X is defined as the deviation between the prior uncertainty and expected posterior uncertainty after using feature X . Feature X is preferred to feature Y if the information gain from feature X is greater than that from feature Y [18]. Two feature weighing algorithms involved in this paper are both based on distance strategy.

2.1. Relief Feature Weighting Algorithm

Relief algorithm was first proposed by Kenji Kira [19]. It requires linear time in the number of given features and the number of training instances regardless of the target concept to be learned. In a binary classification problem, an instance x_i is represented by a vector composed of p feature value. $X=\{x_1, x_2, \dots, x_n\}$ denotes a set of training instances with size n . λ is a $p \times 1$ vector denoting the weight of each dimensional feature. In relief algorithm, for each instance x_i , L instances which have the closest Euclid distance to x_i of the same class, are selected as its Near-hit instances [19], denoted by $h_j, j=1, 2, \dots, L$. L instances which have the closest Euclid distance to x_i of the different class, are selected as its Near-miss instances, denoted by $m_j, j=1, 2, \dots, L$. $diff_near_hit$ is a $p \times 1$ vector which represents the difference between h_j and x_i .

$$diff_near_hit = \sum_{j=1}^L \frac{|x_i - h_j|}{nu} \quad (1)$$

Where nu is a normalization unit to normalize the values into the interval $[0, 1]$, usually $nu = \max(x) - \min(x)$, where $\max(x)$ ($\min(x)$) denotes the maximum (minimum) element in X . $diff_near_miss$ is a $p \times 1$ vector which represents the difference between m_j and x_i .

$$diff_near_miss = \sum_{j=1}^L \frac{|x_i - m_j|}{nu} \quad (2)$$

Then update the feature weight vector λ for each instance. The update formulation is given by

$$\lambda_{new} = \lambda_{old} - diff_near_hit / L + diff_near_miss / L \quad (3)$$

2.2. CSC Feature Weighing Algorithm

CSC algorithm is introduced in[5], it is based on the assumption that important feature should have smaller within-class difference and bigger between-class difference, and vice versa, which is similar to relief algorithm to some extent.

A binary classification problem is assumed as well to explain CSC algorithm. An instance x_i is represented by a vector composed of p feature value. $X=\{x_1, x_2, \dots, x_n\}$ denotes a set of training instances with size n . λ is a $p \times 1$ vector denoting the weights of

each dimensional feature. D_w is a $N \times p$ matrix, which denotes the difference of instances in the same class.

$$D_w(i, k) = \frac{1}{\alpha_m - 1} \sum_{j=1}^{\alpha_m} (x_i^m(k) - x_j^m(k))^2 \quad (4)$$

$$i = 1, 2, \dots, n \quad k = 1, 2, \dots, p \quad m = 1, 2$$

α_m denotes the number of instances in class m and x_i^m denotes the i th instance in class m .

Thus the within-class difference could be measured by $D_in(k)$

$$D_in(k) = \frac{1}{n} \sum_{i=1}^n D_w(i, k)$$

$$= \frac{1}{2} \sum_{m=1}^2 \frac{1}{\alpha_m} \sum_{i=1}^{\alpha_m} D_w(i, k) \quad (5)$$

$$= \frac{1}{2} \sum_{m=1}^2 \frac{1}{\alpha_m} \sum_{i=1}^{\alpha_m} \frac{1}{\alpha_m - 1} \sum_{j=1}^{\alpha_m} (x_i^m(k) - x_j^m(k))^2$$

D_b is an $N \times p$ matrix, which denotes the difference of each instance with other instances in other classes.

$$D_w(i, k) = \sum_{\substack{n=1 \\ n \neq m}}^2 \frac{1}{\alpha_n} \sum_{j=1}^{\alpha_n} (x_i^m(k) - x_j^n(k))^2 \quad (6)$$

x_i^m is supposed to be the i th instance in class m , and x_j^n stands for the j th instance of class n .

Then the between-class difference could be measured by

$$D_between(k) = \frac{1}{n} \sum_{i=1}^n D_b(i, k)$$

$$= \frac{1}{2} \sum_{m=1}^2 \frac{1}{\alpha_m} \sum_{i=1}^{\alpha_m} D_b(i, k) \quad (7)$$

$$= \frac{1}{2} \sum_{m=1}^2 \frac{1}{\alpha_m} \sum_{i=1}^{\alpha_m} \sum_{\substack{n=1 \\ n \neq m}}^2 \frac{1}{\alpha_n} \sum_{j=1}^{\alpha_n} (x_i^m(k) - x_j^n(k))^2$$

Finally the weight of the k th feature is denoted by

$$\lambda(k) = \frac{D_between(k)}{D_in(k)} \quad k = 1, 2, \dots, p \quad (8)$$

2.3 Application of weighing algorithm to classification

SVM is the basic classification algorithm used in this study. A full introduction to SVM could be found in [1, 20]. Suppose the training data set S consist of n vectors, each vector $x_i \in R^d$ belongs to either of two classes and is given a label $y_i \in \{-1, +1\}$ for $i = 1, 2, \dots, n$.

$$S = [(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)] \quad (9)$$

SVM tries to find a separating hyperplane $\omega \cdot x + b = 0$ that maximizes the margin between two classes. Maximizing the margin is a quadratic programming (QP) problem:

$$\min_{\omega, b} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n \xi_i \quad (10)$$

subject to $y_i(\omega \cdot x_i + b) \geq 1 - \xi_i, \xi_i \geq 0$ for $i = 1, 2, \dots, n$

where C is a constant which determining the tradeoff between maximizing margin and minimizing the number of misclassified instances. In practice, instead of solving the primal form Eq.(10), we usually solve the follow dual form:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j x_i \cdot x_j - \sum_{i=1}^n \xi_i \quad (11)$$

subject to $\sum_{i=1}^n y_i \alpha_i = 0, 0 \leq \xi_i \leq C$ for $i = 1, 2, \dots, n$

where $\alpha_i, i = 1, 2, \dots, n$ are Lagrange multipliers. And then the decision function is given by:

$$f(x) = \text{sgn}\left(\sum_{i=1}^n \alpha_i y_i x_i \cdot x + b\right) \quad (12)$$

To solve nonlinear classification problems, a kernel function is introduced to replace the inner product in Eq.(11)(12). Therefore, the SVM classifier can be represented as

$$f(x) = \text{sgn}\left(\sum_{i=1}^n \alpha_i y_i K(x_i \cdot x) + b\right) \quad (13)$$

where $K(x_i \cdot x) = \Phi(x_i)\Phi(x)$ is the kernel function which satisfies Mercer's theorem. Commonly used kernel functions are polynomials and Gaussian radial basic functions, as follows:

$$K(x \cdot y) = (x^T y + 1)^d \quad (14)$$

$$K(x \cdot y) = \exp(-\gamma \|x - y\|^2) \quad (15)$$

d is the order of the polynomials and γ denotes the scaling factor in the radial basis function kernel.

As for the way to introduce the weights of different features into SVM classification, a new method was proposed in this paper which takes advantage of the strategy of data scaling. The weights of each features, obtained by algorithms mentioned above or other appropriate algorithms according to different training data, are multiplied to corresponding features of data which has been scaled to the range $[-1, +1]$. Therefore features with larger contribution to the classification are scaled to a larger range to strengthen their influence in the process of classification, and features with smaller contribution to the classification are scaled to a smaller range or even zero to weaken their influence. The principle that different feature ranges result in different influence to the classification is easy to understand. The final decision values ($\sum_{i=1}^n \alpha_i y_i x_i \cdot x + b$ in Eq.(12)

and $\sum_{i=1}^n \alpha_i y_i K(x_i \cdot x) + b$ in Eq.(13)) in SVM depend on the inner products or kernel values of feature vectors, and large attribute (feature) values would dominate those smaller attribute values in the computational process. The major advantage of this method is that it introduces feature weighting into classification, while bringing no change to the algorithm structure at the same time. Thus it could be applied to many other classification algorithms as well.

3. Interval Type-2 Fuzzy Logic and its Extended Approach

Type-2 fuzzy logic was first introduced by Karnik *et. al.*, [13]. Aiming at the problem of computation complexity increase in comparison with type-1 fuzzy logic, Liang and Mendel proposed interval type-2 fuzzy logic [21]. In interval type-2 fuzzy logic, the secondary membership functions are interval sets, which simplify the computation complexity remarkably. Interval type-2 fuzzy logic is adopted in the ensemble classifier of this paper.

Type-2 fuzzy logic performs much better than type-1 in handling uncertainty. Typically, interval type-2 fuzzy logic is adopted for simplifying the computation. There exist two memberships in an interval type-2 fuzzy set: the primary membership J_x and secondary membership $f_x(u)$ with all secondary grades of the primary memberships equaling to one.

Figure 1 illustrates an example of an interval type-2 fuzzy set where the gray shaded region denotes the footprint of uncertainty (FOU). The primary membership of an instance x' is denoted by an interval with an upper bound $\bar{\mu}(x')$ and a lower bound $\underline{\mu}(x')$. By contrast, the membership in a type-1 fuzzy set is a crisp value instead. The vertical slice of x' shows that the secondary grades of the primary membership is equal to one.

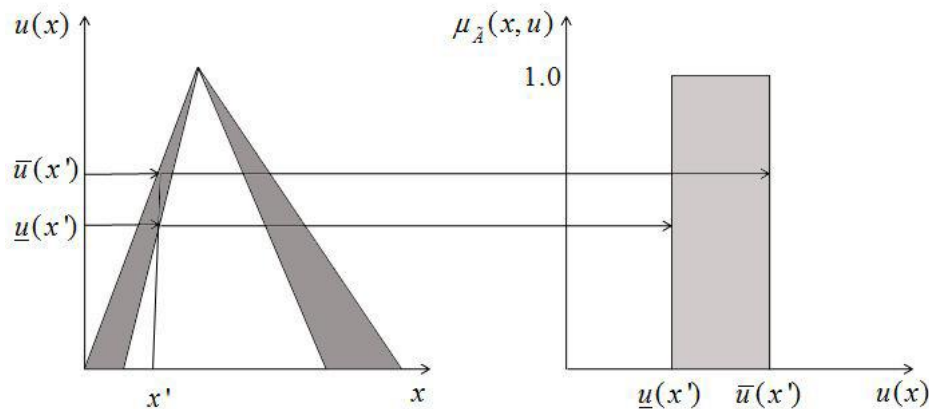


Figure 1 (a) an interval type-2 fuzzy set (b) the vertical slice of instance x'

Thus an interval type-2 fuzzy set \tilde{A} could be represented as

$$\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)) \mid \forall x \in A, \forall u \in J_x \subseteq [0, 1], \mu_{\tilde{A}}(x, u) = 1\} \quad (16)$$

In the ensemble classifier, with the application of type-2 fuzzy logic, the weight of each single classifier w_i is no longer a crisp value but an interval set $\tilde{w}_i = [\bar{w}_i, \underline{w}_i]$, and then the final decision y could be represented as

$$y = \frac{\sum_{i=1}^M \tilde{w}_i y_i}{\sum_{i=1}^M \tilde{w}_i} \quad (17)$$

where y_i is the result of i th single classifier and M denotes the total number of single classifiers.

The final result y is a type-2 fuzzy set and must be reduced to a type-1 fuzzy set so that typical defuzzification could be applied to generate a crisp output, which is called type reduction. Type reduction is an additional step different from type-1 fuzzy. There exists many kinds of type reduction[22], such as centroid, center-of-set, height and modified height. In this paper, center-of sets type reduction was used. To compute the type-2 output fuzzy set y , it's sufficient to compute its upper bound y_r and lower bound y_l , thus the output y can be expressed as $y = [y_l, y_r]$. Figure 2 displays Karnik-Mendel iterative algorithm to compute the upper bound y_r . The lower bound could be calculated in the similar way except in step 4: set $w_i = \bar{w}_i$ for $i \leq R$ and $w_i = \underline{w}_i$ for $i > R$. It has been proved that this iterative procedure can converge in at most M iterations to find y_r and y_l [23].

Defuzzification is applied after type reduction, the final output is set to be the average of y_r and y_l

$$y = \frac{y_r + y_l}{2} \quad (18)$$

1. Arrange y_i in ascending order, i.e. $y_1 \leq y_2 \leq \dots \leq y_M$;
2. Set $w_i = (\bar{w}_i + \underline{w}_i) / 2$ for $i = 1, 2, \dots, M$ and compute y' by $y' = \frac{\sum_{i=1}^M w_i y_i}{\sum_{i=1}^M w_i}$;
3. Find $R \in [1, M-1]$ such that $y_R \leq y' \leq y_{R+1}$;
4. Set $w_i = \underline{w}_i$ for $i \leq R$ and $w_i = \bar{w}_i$ for $i > R$, compute y'' using

$$y'' = \frac{\sum_{i=1}^M w_i y_i}{\sum_{i=1}^M w_i} ;$$

Figure 2. Karnik-Mendel Iterative Procedure to Calculate y_r

4. Application of Interval Type-2 Fuzzy Logic in Ensemble SVM

Different single SVMs with different kernels or parameters may generate different separating hyperplanes, and thus may bring about different classification results for the

same instance. The strategy of ensemble SVM is based on the assumption that the classification results given by several single SVMs have a higher reliability in contrast with the single SVM. Just like the diagnosis report based on several doctors' judgements is usually more reliable than the conclusion draw by a single doctor. The strategy of assigning different weight to each single classifier is supposed to improve the reliability of the final classification results, *i.e.*, high accuracy classifiers own high weight and vice versa.

To handle the uncertainty in the mapping relationship between weight and accuracy, type-2 fuzzy logic was introduced into the ensemble process in this paper. The distance of a testing instance to a single SVM's hyperplane was adopted as this single classifier's output instead of its final decision. It is reasonable because the distance contains information of final decision as well as its confidence coefficient. Therefore an instance in positive class holds a positive distance and vice versa. Figure 3 illustrates the structure of ensemble SVM with application of interval type-2 fuzzy logic.

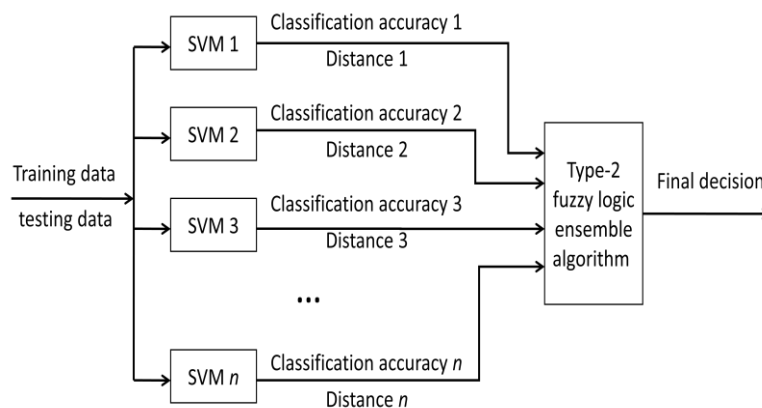


Figure 3. Structure of Ensemble SVM

In this paper, six single SVMs were combined using type-2 fuzzy logical, and this process could be easily extended to arbitrary number of single SVMs.

In the process above, the mapping relationship (*i.e.*, the membership function (MF)) between classification accuracies and weights determines the ensemble classifier's final accuracy. The Gaussian MF of traditional type-1 fuzzy ensemble classifier is shown in Figure 4 (a)

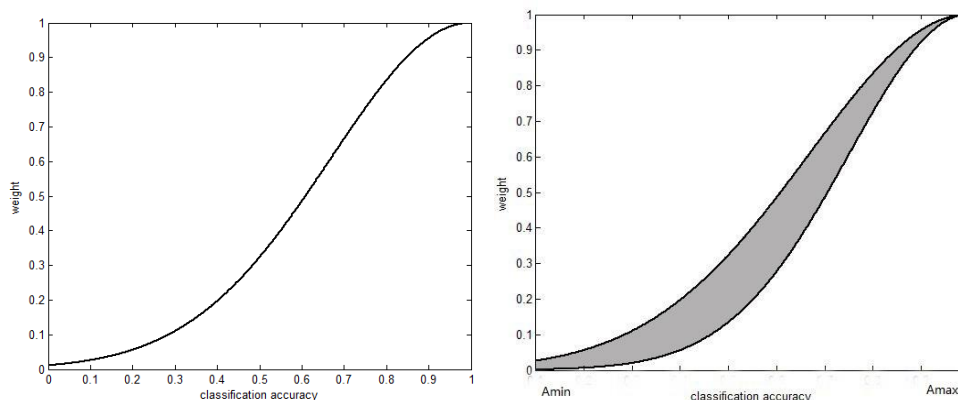


Figure 4. (a) Type-1 Fuzzy MF (b) Type-2 Fuzzy MF

As shown in Figure 4(a), once the MF is determined, the uncertainty in mapping

relationship disappears. For the sake of maintaining the uncertainty, type-2 fuzzy MF was applied as shown in Figure 4(b). Generally speaking, classification accuracies of all single SVMs have a scale from 0% to 100%. In order to make the MF more sensitive to the changes of accuracies, the minimum and maximum accuracies were valued as the two bounds of the domains of the accuracies.

The MF shown in Figure 4(b) is a Gaussian primary MF with uncertain standard deviation, i.e. the Gaussian primary MF has a fixed mean A_{\max} and an uncertain standard deviation which ranges in $[\sigma_{\min}, \sigma_{\max}]$. In this paper two bounds were set as

$$\sigma_{\min} = \frac{A_{\max} - A_{\min}}{3} \quad \sigma_{\max} = \frac{A_{\max} - A_{\min}}{2} \quad (19)$$

thus

$$\mu(a_k) = \exp\left[-\frac{1}{2}\left(\frac{a_k - A_{\max}}{\sigma}\right)^2\right], \quad \sigma \in [\sigma_{\min}, \sigma_{\max}] \quad k = 1, 2, \dots, 6 \quad (20)$$

where a_k denotes the classification accuracy of k th single SVM; $\mu(a_k)$ denotes the weight of k th single SVM. Therefore the upper MF $\bar{\mu}(a_k)$ is

$$\bar{\mu}(a_k) = N(A_{\max}, \sigma_{\max}, a_k) \quad (21)$$

and the lower MF $\underline{\mu}(a_k)$ is

$$\underline{\mu}(a_k) = N(A_{\max}, \sigma_{\min}, a_k) \quad (22)$$

5. Experiments and results

5.1. Dataset description

Australian Credit Approval data from UCI machine learning repository was used to test the performance of SVM based on feature weighting and ensemble SVM based on type-2 fuzzy logic proposed in this study. This dataset includes information of credit card applications with all attribute names and values have been changed to meaningless symbols to protect confidentiality of the data. It is interesting because there is a good mix of attributes -- continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are 6 numerical and 8 categorical attributes. It's well-known that there exist many factors which influence the credit approval, like personal income, property, ages and so on. And it's acceptable to believe that different factors hold their own weights affecting the final process of approval respectively. This dataset was chosen in this study to highlight the advantage of the method proposed in this paper.

5.2. (a) Experiment Design for SVM based on Feature Weighting

In this experiment, data scaling was applied to the original data to scale each attribute to the range $[-1, +1]$. Then the data was classified in ten-fold cross-validation, for each training dataset, two feature weighting algorithms introduced in Section 2 were applied in addition.

For convenience, the "original data" all refers to the data which has been scaled to $[-1, +1]$ in the following, and the following abbreviations are used: "SVM" refers to the

traditional support vector machine, “Relief-SVM” refers to SVM based on relief feature weighting algorithm and “CSC-SVM” refers to SVM based on CSC feature weighting algorithm.

The training datasets applied to SVM, Relief-SVM and CSC-SVM were different because the original data was multiplied by different weights in the latter two methods. Therefore, different combinations of parameters were experimented for these three kinds of SVMs in order to show their best classification performance. Since Gaussian radial basis kernel function (RBF) was used here, to reliably optimize γ and C, a cross-validation work with γ ranging from 2^{-2} to 2^6 and C ranging from 2^{-4} to 2^5 , both with steps of 2^1 , was carried out to obtain the best performance of each classifier. When it comes to Relief-SVM, another parameter L should be taken into consideration, which denotes the number of instances selected as the Near-hit instances and Near-miss instances. The appropriate value of L was selected by comparing classification accuracies with different L.

5.2. (b) Result and Discussion of SVM based on Feature Weighting.

Table 1 shows the weights of 14 attributes in the training data calculated by relief and CSC feature weighting algorithms.

Table 1. Weights of 14 Attributes

		Weights of attributes													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Relief	1	6.5	1.2	1.1	1.8	1.5	1.5	1.2	11.	4.9	1.4	10.	1.7	1.2	1.1
		2	1	4	9	2	7	5	1	3	5	3	4	3	2
	2	5.0	1.3	1.2	1.8	1.4	1.4	1.3	12.	4.1	1.3	15.	2.2	1.1	1.1
		2	0	5	9	0	7	3	3	5	8	4	6	6	1
	3	5.1	1.3	1.2	2.0	1.4	1.5	1.3	12.	4.4	1.3	8.1	1.8	1.1	1.0
		6	1	2	4	8	3	1	5	3	8	3	2	9	6
	4	4.9	1.2	1.1	1.9	1.4	1.6	1.2	10.	4.8	1.3	7.6	1.7	1.1	1.0
		8	5	9	9	8	2	1	5	6	3	1	9	8	4
	5	6.3	1.2	1.2	1.7	1.4	1.5	1.2	10.	4.5	1.4	10.	1.8	1.2	1.1
		5	7	1	4	6	3	3	4	9	0	2	3	2	1
	6	4.4	1.2	1.2	2.4	1.5	1.4	1.2	10.	3.9	1.3	8.1	2.2	1.2	1.0
		6	9	5	0	1	6	1	4	5	1	4	0	0	8
	7	7.9	1.2	1.2	1.7	1.4	1.5	1.3	11.	6.7	1.5	11.	1.8	1.2	1.0
		7	3	4	9	6	5	8	5	6	3	1	2	2	3
8	5.8	1.2	1.2	1.6	1.4	1.4	1.3	13.	6.0	1.4	8.7	1.7	1.2	1.0	
	6	3	8	8	2	1	1	1	8	3	1	0	1	2	
9	5.4	1.2	1.2	1.8	1.5	1.6	1.3	10.	4.4	1.3	9.2	1.9	1.1	1.1	
	0	8	2	3	5	1	2	6	3	5	1	0	8	0	
10	6.1	1.2	1.2	1.6	1.4	1.4	1.2	10.	4.4	1.4	11.	1.7	1.1	1.0	
	0	0	0	5	2	3	3	5	1	6	3	0	7	5	6
CSC	1	1	1.0	1.0	1.1	1.3	1.1	1.2	3.0	1.5	1.5	1	1.0	1.0	1.0
		3	6	2	1	7	4	8	4	1	6	1	5		

2	1	1.0	1.1	1.0	1.2	1.1	1.2	3.0	1.4	1.3	0.9	1.0	1.0	1.0
		7	0	9	4	0	4	8	8	2	9	3	3	5
3	1	1.0	1.1	1.1	1.3	1.0	1.2	3.5	1.4	1.2		1.0	1.0	1.1
		9	2	0	6	9	3	7	3	9	1	4	2	1
4	1	1.0	1.0	1.0	1.3	1.1	1.1	3.0	1.6	1.4		1.0	1.0	1.0
		4	5	9	4	8	8	3	0	3	1	3	2	5
5	1	1.0	1.1	1.0	1.2	1.1	1.1	3.3	1.4	1.2		1.0	1.0	1.0
		4	4	6	8	2	6	3	7	7	1	4	1	4
6	1	1.0	1.0	1.0	1.3	1.1	1.1	2.9	1.6	1.3	0.9	1.0	1.0	1.0
		7	6	8	1	1	6	2	7	5	9	7	5	6
7	1.0	1.0	1.0	1.0	1.3	1.1	1.2	3.5	1.4	1.5	1.0	1.0	1.0	1.0
	1	8	6	7	8	9	9	6	9	6	1	3	1	5
8	0.9	1.1	1.1	1.0	1.2	1.0	1.2	2.9	1.3	1.3		1.0	1.0	1.0
	9	0	2	9	5	9	5	2	9	1	1	1	2	6
9	1	1.0	1.0	1.1	1.4	1.1	1.2	3.4	1.5	1.3		1.0	1.0	1.0
		4	5	0	0	0	3	7	2	8	1	1	1	5
1	0.9	1.0	1.0	1.0	1.2	1.1	1.1	2.9	1.4	1.4		1.0	1.0	1.0
	9	3	6	5	8	4	9	7	2	5	1	4	1	4

The weights of 14 attributes based on ten-fold cross-validations show that different features or attributes make different contribution to the final classification indeed. In the comprehensive analysis of weights calculated respectively by relief and CSC algorithm, two roughly similar results were obtained by these two feature weighting algorithms, i.e. they both draw a conclusion that feature 8, 9 are of relative importance while feature 2,3,6,7,13,14 are relatively insignificant or even ignorable in the classification. However, these two algorithms' opinions varied on feature 1,4,5,10,11,12 which is acceptable since these two are based on different computation process. CSC calculates the difference in the same class (between different classes) by going through all the instances in the same class (different classes), while relief algorithm just goes through L instances for the purpose of simplifying the computation complexity. Thus the relief algorithm might lose some data distribution information in exchange.

Table 2 shows the classification accuracies of three methods (SVM, relief-SVM, CSC-SVM) based on ten-fold cross-validation.

Table 2. Classification Accuracies of Three Methods

		Classifier		
		SVM	Relief-SVM	CSC-SVM
Classification accuracy(%)	1	87.77	88.49	89.21
	2	86.33	86.33	89.21
	3	84.89	85.61	87.77
	4	83.45	84.17	84.17
	5	84.89	85.61	86.33
	6	89.93	89.21	90.65
	7	83.45	84.89	84.89

8	86.33	88.49	90.65
9	86.33	87.05	89.21
10	88.49	88.49	90.65

Overall, classification accuracies based on ten-fold cross-validation in Table 2 show that feature weighting SVMs, either relief-SVM or CSC-SVM, outperformed the traditional SVM without feature weighting. In comparison with relief-SVM, CSC-SVM showed an average increase in overall accuracies of around 3%, outperforming relief-SVM's 1%. It's reasonable because relief algorithm lost some data distribution information, which is mentioned before. The result illustrates that the weights calculated by CSC algorithm matches the practice better than relief algorithm. Thus one can draw a conclusion that the selection of feature weighting algorithm is of great significance when applying the method proposed in this paper.

5.3. (a) Experiment design for ensemble SVM based on type-2 fuzzy logic

To verify the effectiveness of adding type-2 fuzzy logic into ensemble SVM, another experiment was designed as well. The fundamental framework of this experiment is shown in Figure 3, six single SVMs were combined based on different weights which have a type-2 fuzzy relationship with corresponding classification accuracies.

To obtain the single SVMs' classification accuracies, the first choice is to use the training accuracies. However, it may not be as good as expected since the classification hyperplanes were all based on training data, thus there exists over-fitting more or less and leads to a high training accuracy but low testing accuracy. Considering that the original data was classified in n-fold cross-validation in this classification experiment, it should be an acceptable choice to further divided each n-fold training dataset to m-fold to obtain the final classification accuracies of single SVMs, as adopted in[15]. Thus, in this experiment each single SVM's classification accuracy was calculated in this manner: the original data was classified in six-fold cross-validation, and for each six-fold training dataset another six-fold cross-validation was applied to obtain the average classification accuracy of each single SVM. Finally the average accuracy will be used for weighting.

Another point which needs attention is the output of each single SVM. As mention in Section 4, the distance of a testing instance to a single SVM hyperplane was adopted as its output instead of its final decision in this experiment. The result of first experiment shows that SVMs based on CSC feature weighting algorithm outperformed the other two methods, therefore, CSC-SVMs were chosen as the single classifiers in this experiment. As for the selection of single SVMs, there are two choices: one is to select several SVMs with same kernel function but different parameters, and the other is to select several SVMs with different kernel functions and different parameters. In order to determine which kind of combination would be better, two attempts (denoted as Attempt 1 and Attempt 2) based on these two choices were made. For Attempt 1, six combinations of parameter were randomly selected for experiment. Table 3 shows the classification accuracies (six-fold cross-validation) of the six RBF-based SVMs selected in attempt one. For Attempt 2, both polynomial kernel function and radial basis function were used with different random parameters. Table 4 shows the parameters and classification accuracies (six-fold cross-validation) of single SVMs based on different kernel function. For the sake of comparison, ensemble SVM based on type-1 fuzzy logic was implemented as well. The mapping relationship between classification accuracies and weights has been introduced in Section 4.

It should be noted that the mapping relationships between classification accuracies and weights, both in type-1 and type-2 fuzzy, were based on practical experience, and there

might exist some other mapping relationships more suitable. Yet the mapping relationships given out is already qualified to verify type-2 fuzzy logic's capability to handle uncertainty in the ensemble SVM.

Table 3. Parameters and Classification Accuracies of Single SVMs based on RBF

	Parameter		Classification accuracy (%)						
	γ	C	1	2	3	4	5	6	Average
SVM1	1	0.25	86.96	84.06	86.86	84.78	84.78	82.61	85.01
SVM2	0.25	0.5	84.78	82.61	84.06	82.61	81.16	81.16	82.73
SVM3	0.25	1	83.33	82.61	84.06	81.88	79.71	80.43	82.00
SVM4	2	2	86.23	84.06	86.96	84.78	84.78	82.61	84.90
SVM5	4	4	84.78	84.06	85.51	84.06	84.06	81.88	84.06
SVM6	8	16	84.78	84.06	85.51	84.06	85.51	81.88	84.30

Table 4. Parameters and Classification Accuracies of Single SVMs based on Different Kernel

	Kernel	Parameter		Classification accuracy (%)						
		γ/d	C	1	2	3	4	5	6	Average
SVM1	RBF	1	0.25	84.06	87.68	86.96	87.68	86.23	86.96	86.60
SVM2	RBF	0.25	0.5	81.88	81.88	82.61	55.07	84.78	81.88	78.02
SVM3	RBF	8	16	84.78	89.86	86.23	86.96	86.96	87.68	87.08
SVM4	Polynomial	1	0.25	84.78	89.86	85.51	86.96	86.96	87.68	86.96
SVM5	Polynomial	2	0.5	85.51	89.13	85.51	60.14	85.51	84.06	81.64
SVM6	Polynomial	2	1	85.51	89.13	86.23	58.7	85.51	84.06	81.52

5.3. (b) Results and discussion of ensemble SVM based on type-2 fuzzy logic

Classification accuracies based on six-fold cross-validation (Table 5, Table 6) showed that both ensemble SVMs based on type-1 fuzzy logic and type-2 fuzzy logic outperformed the average accuracies of six single SVMs. The analysis of ensemble SVM based on type-1 and type-2 showed that the ensemble SVM based on type-2 outperformed type-1. Another important result is that the type-2 based SVM outperformed any of the six single SVMs in most cases.

Further detailed analysis of the results showed that in Table 4, type-2 based SVM outperformed the best single SVM in four tests while in test 1 the result is opposite. In Table 5, the performance of type-2 based SVM was better, the possible reason is that single classifiers with different kernel functions may complement each other better, in comparison with classifiers with same kernel functions. What is more, when some of the single SVMs performed much badly such as test 1 and test 2 in Table 6, the performance of type-1 based SVM was also seriously affected. On the contrary, the type-2 based SVM performed well, which means stronger robustness.

Finally, it can draw a conclusion that type-2 based ensemble SVM has two main advantages in comparison with single SVM. Firstly, an increase in classification accuracy is obviously obtained. Secondly, it only needs to select several single classifiers to construct a type-2 based ensemble classifier to obtain high classification accuracy in

practical classification, instead of trying out many different combinations of parameters.

Table 5. Classification Accuracies of Six Single SVMs based on RBF, Ensemble SVMs based on Type-1 and Type-2 Fuzzy

Test	Classification accuracy(%)									
	SVM1	SVM2	SVM3	SVM4	SVM5	SVM6	Average	Maximum	Type-1	Type-2
1	86.96	89.57	86.96	86.96	86.96	86.96	87.40	89.57	88.7	88.7
2	90.43	87.83	88.7	89.57	90.43	89.57	89.42	90.43	90.43	91.3
3	82.61	83.48	81.74	82.61	81.74	81.74	82.32	83.48	81.74	83.48
4	86.96	81.74	86.96	86.96	85.22	85.22	85.51	86.96	86.96	87.83
5	81.74	80.87	81.74	82.61	81.74	81.74	81.74	82.61	81.74	83.48
6	86.09	83.48	88.7	86.96	86.96	70.43	83.77	88.7	86.96	89.57

Table 6. Classification Accuracies of Six Single SVMs based on Different Kernel, Ensemble SVMs based on Type-1 and Type-2 Fuzzy

Test	Classification accuracy(%)									
	SVM1	SVM2	SVM3	SVM4	SVM5	SVM6	Average	Maximum	Type-1	Type-2
1	83.48	80	86.96	86.96	83.48	55.65	79.42	86.96	86.09	87.83
2	53.91	89.57	90.43	90.43	88.7	53.91	77.83	90.43	62.61	90.43
3	84.35	80	85.22	85.22	82.61	74.78	82.03	85.22	82.61	86.09
4	86.96	81.74	86.96	86.96	85.22	85.22	85.51	86.96	86.96	87.83
5	81.74	80.87	81.74	82.61	81.74	81.74	81.74	82.61	81.74	83.48
6	91.3	86.96	91.3	91.3	90.43	91.3	90.43	91.3	92.17	92.17

6. Conclusion

In this paper, two methods to improve the performance of SVM classifier were proposed. The first method is based on the strategy of feature weighting. The original data would be transformed by multiplied corresponding weights for better classification. The major advantage of this method is to improve the performance of SVM classifier without any change to the original classification algorithm. Therefore, it can be regard as a special kind of data preprocessing technique. The second method referred to ensemble classifier based on interval type-2 fuzzy logic. With the merit of interval type-2 fuzzy logic, the final ensemble SVM could achieve a more reasonable classification result. In short, ensemble SVM based on type-2 logic outperformed the traditional SVMs as well as the type-1 based ensemble SVM.

Acknowledgments

This study was supported by the National Natural Science Foundation of China(61174114), the Research Fund for the Doctoral Program of Higher Education in China(20120101130016) and the Zhejiang Provincial Science and Technology Planning Projects of China (2014C31019)

References

- [1] C. Cortes and V. Vapnik, "Support-vector networks", *Machine Learning*, vol. 20, no. 3, (1995), pp. 273-297.
- [2] C. F. Lin and S. D. Wang, "Fuzzy support vector machines", *IEEE Transactions on Neural Networks*, vol. 13, no. 2, (2002), pp. 464-471.
- [3] H. Xue, S. E. Chen and Q. Yang, "Structural Support Vector Machine", *Advances In Neural Networks*, Berlin, Germany, (2008), pp. 501-511.
- [4] L. Wang, Z. Liu, C. P. Chen, and Y. Zhang, "Interval type-2 fuzzy weighted support vector machine learning for energy efficient biped walking", *Applied Intelligence*, vol. 40, no. 3, (2013), pp.453-463.
- [5] B. Qi, C. H. Zhao and G. S. Yin, "Feature weighting algorithms for classification of hyperspectral images using a support vector machine", *Applied Optics*, vol. 53, no. 13, (2014), pp. 2839-2846.
- [6] Qi, B.,Zhao C.H., "Use of weighting algorithms to improve traditional support vector machine based classifications of reflectance data", *Optics Express*, vol. 19,no. 27, (2011), pp.26816-26826.
- [7] Fu, H. and A.M. Elmisery. "A new feature weighted fuzzy c-means clustering algorithm", *IADIS european conference on Data Mining*, pp. 11-18, (2009), Portugal.
- [8] Sarle, W.S., *Neural network FAQ*, <ftp://ftp.sas.com/pub/neural/FAQ.html>, (1997).
- [9] Hsu, C.-W., C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification", www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf, (2003).
- [10] Breiman, L., "Bagging predictors", *Machine learning*, vol. 24,no. 2, (1996), pp.123-140.
- [11] Freund, Y. and R.E. Schapire, "Experiments with a new boosting algorithm", *Machine Learning: Proceedings of the Thirteenth International Conference*, pp. 148-156, (1996).
- [12] Kittler, J., Hatef, M., Duin, R.P., and Matas, J., "On combining classifiers", *IEEE Transactions on Pattern Analysis And Machine Intelligence*, vol. 20,no. 3, (1998), pp.226-239.
- [13] Karnik, N.N., J.M. Mendel, and Q.L. Liang, "Type-2 fuzzy logic systems", *IEEE Transactions on Fuzzy Systems*, vol. 7,no. 6, (1999), pp. 643-658.
- [14] Hwang, C. and F.-H. Rhee, "Uncertain fuzzy clustering: interval type-2 fuzzy approach to C-means", *IEEE Transactions on Fuzzy Systems*, vol. 15,no. 1, (2007), pp.107-120.
- [15] Chen, X., Li, Y., Harrison, R. and Zhang, Y. Q., "Type-2 fuzzy logic-based classifier fusion for support vector machines", *Applied Soft Computing*, vol. 8,no. 3, (2008), pp.1222-1231.
- [16] Guyon, I. and A. Elisseeff, "An introduction to variable and feature selection", *The Journal of Machine Learning Research*, vol. 3, (2003), pp.1157-1182.
- [17] Dash, M. and H. Liu, "Feature selection for classification", *Intelligent data analysis*, vol. 1,no.3, (1997), pp.131-156.
- [18] Rodriguez-Lujan, I., Bailador, G., Sanchez-Avila, C., Herrero, A., Vidal-de-Miguel,G., "Analysis of pattern recognition and dimensionality reduction techniques for odor biometrics", *Knowledge-Based Systems*, vol. 52, (2013), pp.279-289.
- [19] Kira, K. and L.A. Rendell. "A practical approach to feature selection", *Proceedings of the ninth international workshop on Machine learning*, pp. 249-256, (1992).
- [20] Vapnik, V., "The nature of statistical learning theory", *springer*, (2000).
- [21] Liang, Q.L. and J.M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design", *IEEE Transactions on Fuzzy Systems*, vol. 8,no.5, (2000), pp.535-550.
- [22] Karnik, N.N. and J.M. Mendel. "Type-2 fuzzy logic systems: type-reduction", *1998 IEEE International Conference on, IEEE*, pp. 2046-2051, (1998).
- [23] Mendel, J.M., "Uncertain rule-based fuzzy logic system: introduction and new directions", (2001).

Authors



Weiping Huang, he is studying for a doctor's degree in department of control science and engineering, Zhejiang University. His current research interests include pattern recognition.