Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Weighted classifier ensemble based on quadratic form

Shasha Mao [a,b,*,1], Licheng Jiao [a], Lin Xiong [a,1], Shuiping Gou [a], Bo Chen [a], Sai-Kit Yeung [b]

[a] Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, PR China
[b] The Vision, Graphics and Computational Design Group at the Information Systems Technology and Design, Singapore University of Technology and Design, Singapore

ARTICLE INFO

Article history:
Received 23 July 2013
Received in revised form
20 August 2014
Accepted 12 October 2014

Keywords:
Ensemble learning
Weighted classifier ensemble
Quadratic form

ABSTRACT

Diversity and accuracy are the two key factors that decide the ensemble generalization error. Constructing a good ensemble method by balancing these two factors is difficult, because increasing diversity is at the cost of reducing accuracy normally. In order to improve the performance of an ensemble while avoiding the difficulty derived of balancing diversity and accuracy, we propose a novel method that weights each classifier in the ensemble by maximizing three different quadratic forms. In this paper, the optimal weight of individual classifiers is obtained by minimizing the ensemble error, rather than analyzing diversity and accuracy. Since it is difficult to minimize the general form of the ensemble error directly, we approximate the error in an objective function subject to two constraints ($\sum w_i = 1$ and $-1 < w_i < 1$). Particularly, we introduce an error term with a weight vector $\mathbf{w}_0$, and subtract this error with the quadratic form to obtain our approximated error. This subtraction makes minimizing the approximation form equivalent to maximizing the original quadratic form. Theoretical analysis finds that when the value of the quadratic form is maximized, the error of an ensemble system with the corresponding optimal weight $\mathbf{w}^*$ will be smallest, especially compared with the ensemble with $\mathbf{w}_0$. Finally, we demonstrate improved classification performance from the experimental results of an artificial dataset, UCI datasets and PolSAR image data.

1. Introduction

Ensemble learning as an active research indeed improves the performance of a single learner by combining multiple learners [1,2]. In recent years, it has been widely used in fields of not only supervised learning but also unsupervised learning [3–5]. Classifier ensemble [6–8] is considered as a classical application that ensemble learning is employed to combine multiple classifiers in supervised learning in order to improve the accuracy and stability of a single classifier. Moreover, it is also named as multiple classifier system [8]. In a classifier ensemble system, a classifier as a learner is called an individual classifier or an individual. At present, many ensemble methods [9–13,37,38] have been proposed and they roughly fall into two basic categories. One lays emphasis on how to construct individual classifiers, and the other lays emphasis on how to combine individual classifiers.

For the former, it concentrates on making different training subsets for individual classifiers, and many classical ensemble strategies have been proposed, such as bagging [2], AdaBoost [14], random forest [7], rotation forest [9] and so on. For the latter, it engages in how to combine the outputs of individual classifiers and is considered as a research hotspot of classifier ensemble recently. From the point of the value of classifiers' coefficients, the existing methods about combining classifiers are roughly divided into three categories: (a) *simple vote strategy* [8,11]: It combines all individual classifiers' outputs with same probability. In other words, all individual classifier are given a same weight coefficient in simple vote strategy. Especially, it is equivalent to majority vote [8,15] as a most popularly used rule; (b) *weighted classifier ensemble* (*WCE*) [8,16,17,39–44]: It combines individual classifiers with different weight coefficients, and the value of each weight coefficient is not equal to zero. In WCE, it indicates that each individual classifier is supposed to have a different contribution for improving the performance; (c) selective or pruning classifier ensemble [18–20,45–48]. It combines individual classifiers with a weight vector including a zero coefficient at least, which indicates that some individual classifiers have negative or insignificant effects on boosting the performance. According to Zhou et al. [21], it demonstrates that an ensemble of partial individual classifiers is better than all. In particular, our work focuses on designing a weighted classifier ensemble method in this paper.

In general, the ensemble generalization error is decided by diversity among individual classifiers and accuracy of them in an ensemble system [22,14,7]. On this basis, many ensemble

* Corresponding author at: The Vision, Graphics and Computational Design Group at the Information Systems Technology and Design, Singapore University of Technology and Design, Singapore. Tel.: +86 15399028270.
E-mail address: skymss0828@gmail.com (S. Mao).
[1] Contributed equally to this work.

http://dx.doi.org/10.1016/j.patcog.2014.10.017
0031-3203/

algorithms [9,19,38] have been proposed based on accuracy of individuals themselves or diversity among individuals. It illustrates that a good ensemble method depends on not only high accuracy of individual classifiers but also high diversity between a pair of individual classifiers. However, according to Krogh and Vedelsby [22], Zhou et al. [23] and Zhang et al. [18], it is known that the more individual classifiers are of high accuracy rates, the less the diversity among them becomes, because the class label of the target sample is uniform. In other words, enhancing diversity among individual classifiers is at the expense of decreasing their accuracy. Thus constructing a good ensemble is difficult based on diversity and accuracy.

On the other hand, some ensemble algorithms increase diversity among individual classifiers by producing different training subsets. Yet perhaps it may be resulted in an unexpected problem that individual classifiers corresponding to different training subsets gain the same outputs. It means that diversity is not always enhanced while creating classifiers by different training subsets. Moreover, the paper [24] illustrates that accuracy of individual classifiers is the leading factor in improving the ensemble performance compared against diversity among individuals. It means that some classifiers with the better performance are more helpful than ones with the higher diversity but poor performance in an ensemble. In brief, it is tough and inconclusive to design an ensemble method via balancing and analyzing diversity and accuracy. In fact, the initial intention of an ensemble of classifiers is the improvement of the classification performance, and the analysis of diversity and accuracy is also to boost the performance of an ensemble system. Consequently, it is fascinating to see whether or not we construct a method by the explicit analysis on the performance of classifier ensemble rather than facing a dilemma of balancing diversity and accuracy.

In this paper, we propose a novel weighted classifier ensemble method based on quadratic forms, which is also named by QFWEC method. In the proposed method, the ensemble error is directly utilized to seek the optimal weight vector of classifiers instead of analyzing diversity and accuracy, whereas it is difficult to obtain the optimal solution via the minimization of the ensemble error, especially for a binary classification problem. Thereby the QFWEC method converts the minimization of the ensemble error into a new optimization problem that contains an approximation form and two constraints. The approximation form is considered as the target function of seeking the optimal weight vector of the classifiers. Furthermore, the approximation form is decomposed into two parts by introducing a given weight vector. The first part is the ensemble error gained by the introduced weight vector, and the second part is a quadratic form. The value of the approximation form is equal to the one by subtracting the second part from the first part. Specifically, the first part is independent of the solving weight vector. Consequently, the process of minimizing the approximation form is transformed into maximizing the quadratic form. Finally, an optimal weight vector is sought by maximizing the quadratic form in the QFWEC method. In addition, it is found that when the value of the quadratic form is larger, the ensemble error gained by the sought weight vector is lower than the one gained by the introduced weight vector. In addition, the experimental results demonstrate that the proposed method obtains a better performance against other ensemble methods.

The organization of this manuscript is as follows. Section 2 introduces several proposed weighted classifier ensemble algorithms. Section 3 introduces the proposed method in detail, including how to change the minimization of the ensemble error into maximizing a real quadratic form and how to seek the optimal weight vector based on three different optimizations. In Section 4, the experimental results of an artificial dataset, UCI datasets and PolSAR image data are shown to illustrate that the proposed method improves the classification performance. Lastly, Section 5 concludes our work and proposes some future works.

2. Related works

As the latter part of classifier ensemble, combining individual classifiers is actually equal to assemble the predictions obtained by individual classifiers, as important as constructing individual classifiers. Suppose a given training sample set $\mathbf{X}$ with $N \times d$, $\mathbf{X} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, where y_n is the true label of $\mathbf{x}_n$ ($\mathbf{x}_n \in \mathbb{R}^d$), $y_n \in \{\omega_1, \ldots, \omega_C\}$, ω_j expresses the jth class, and C is the number of classes. In an ensemble system, Ψ denotes a set of individual classifiers, $\Psi = \{\mathscr{L}_1, \ldots, \mathscr{L}_L\}$, where $\mathscr{L}_i$ $(i = 1, \ldots, L)$ expresses an individual classifier and L is the number of individual classifiers. Then a general form of combining individual classifiers is given as follows:

$$H(\mathbf{x}_T) = \underset{\omega_j \in \{\omega_1, \ldots, \omega_C\}}{\arg\max} \left(\sum_{i=1}^{L} p_{ij}(\mathbf{x}_T) * w_i \right) \tag{1}$$

where $H(\mathbf{x}_T)$ expresses the predictive label of an unlabeled sample $\mathbf{x}_T$ ($\mathbf{x}_T \in \mathbb{R}^d$) given by an ensemble, $p_{ij}(\mathbf{x}_T)$ is the probability of $\mathbf{x}_T$ classified to ω_j by an individual classifier $\mathscr{L}_i$, and w_i denotes the weight coefficient of $\mathscr{L}_i$. When the processed problem is a two-class classification problem $(y_n \in \{-1, +1\})$, Eq. (1) is also presented in the following formula:

$$H(\mathbf{x}_T) = \operatorname{sgn}\left(\sum_{i=1}^{L} f_i(\mathbf{x}_T) * w_i \right) \tag{2}$$

where $f_i(\mathbf{x}_T)$ expresses the predictive label of $\mathbf{x}_T$ given by $\mathscr{L}_i$.

2.1. Simple vote rule

Simple vote rule [8,11] has been widely used to combine the outputs of individual classifiers in many ensemble strategies which focus on constructing different individual classifiers, such as bagging [2], random subspace method [28], rotation forest [9], and so on. In general, each individual classifier is considered to be of an effect as same as others in simple voting rule. In fact, it is equivalent to giving a same coefficient to all individuals in an ensemble system, such as $w_i = 1$ $(i = 1, \ldots, L)$ in Eq.(1). In other words, each individual is important for improving ensemble performance as same as others while simple voting rule is employed to combine classifiers. Particularly, the ensemble predictive label of an unlabeled sample $\mathbf{x}_T$ is given by the formula $H(\mathbf{x}_T) = \operatorname{sgn}\left(\sum_{i=1}^{L} f_i(\mathbf{x}_T) \right)$ for a two-class classification problem.

2.2. Weighted majority vote

Weighted majority vote [8] achieves the final decision of classifier ensemble by giving more power to more individual classifiers. It indicates that an individual has a different effect from others in an ensemble when it is not of the identical classification performance. According to Kuncheva [8] and Rodriguez [15], the probability that a sample is classified into each class is computed by weighted majority vote, shown as follows:

$$p_j^{wmv}(\mathbf{x}_T) = \sum_{i=1}^{L} w_i d_{ij}, \quad j \in \{1, \ldots, C\} \tag{3}$$

where $p_j^{wmv}(\mathbf{x}_T)$ expresses the probability that $\mathbf{x}_T$ is classified into ω_j by weighted majority vote, w_i is the weight coefficient of a classifier $\mathscr{L}_i$ and satisfies for the condition $\sum_{i=1}^{L} w_i = 1$, and if $\mathbf{x}_T$ is classified into ω_j by $\mathscr{L}_i$, $d_{ij} = 1$, otherwise, $d_{ij} = 0$. Finally, the label

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

ω_j with the maximum probability ($p_j^{wmv}(\mathbf{x}_T)$) is as the final ensemble predictive class label of $\mathbf{x}_T$. Additionally, the weight coefficients of individual classifiers are given in [8]

$$w_i = \log \frac{Acc_i}{1-Acc_i} \tag{4}$$

where Acc_i is the accuracy rate that the training samples are correctly classified by $\mathscr{L}_i$.

2.3. Naive Bayes ensemble rule

Naive Bayes ensemble rule [8,11,15] assumes that each individual classifier is mutually independent with others for giving a class label, and thus it is also named for 'independence model' [25], 'simple Bayes' [26]. According to Kuncheva [8], a confusion matrix CM^i with $C \times C$ is produced based on the true labels of the training samples and their prediction labels given by a classifier $\mathscr{L}_i$, therefore, it represents the information of a classifier $\mathscr{L}_i$. In CM^i, its element $cm^i_{j,k}$ $(j,k=1,\ldots,C)$ denotes the number of all samples which belong to the class label ω_j but are classified into ω_k by $\mathscr{L}_i$. Based on CM^i, the prediction label $H(\mathbf{x}_T)$ of a sample $\mathbf{x}_T$ is obtained by the following formula:

$$H(\mathbf{x}_T) = \arg\max_{j=1,\ldots,C} \left(\frac{1}{N_j^{N-1}} \prod_{i=1}^{L} cm^i_{j,k_i} \right) \tag{5}$$

where N_j is the number of all samples with the true class label ω_j in a training set $\mathbf{X}$. Additionally, Titterington et al. [25] gave the modification of the formula of Naive Bayes ensemble rule in order to account for the possible zero occurred in Eq.(5), shown as follows:

$$H(\mathbf{x}_T) = \arg\max_{j=1,\ldots,C} \left(\frac{N_j}{N} \prod_{i=1}^{L} \frac{cm^i_{j,k_i}+1/J}{N_j+1} \right) \tag{6}$$

3. Weighted classifier ensemble based on three quadratic forms

In this section, we introduce a new weighted ensemble method based on the real quadratic form, which seeks an optimal weight vector of individual classifiers by minimizing the ensemble error rate directly. Classifier ensemble aims at improving the classification performance of a single classifier algorithm, which means that the final classification performance will determine the merit of an ensemble algorithm. In supervised learning, the error rate of classification is generally the probability that samples are incorrectly classified. Similarly, the classification error of an ensemble system is equal to the probability of samples whose predictive labels are not the same as their true labels. It is calculated based on the true labels of samples and their prediction labels given by an ensemble.

Suppose a two-class classification problem (classes:$[-1,+1]$) has a training set $\mathbf{X}$ with class labels ($\mathbf{X}=\{(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_N,y_N)\}$) and a testing set $\mathbf{X}_t$ without labels ($\mathbf{X}_t=\{\mathbf{x}_1,\ldots,\mathbf{x}_M\}$), where $\mathbf{x}_n$ ($\mathbf{x}_n \in \mathbb{R}^d$, $n=1,\ldots,N$) expresses a training sample, y_n is the true class label of $\mathbf{x}_n$, and $\mathbf{x}_m$ ($\mathbf{x}_m \in \mathbb{R}^d$, $m=1,\ldots,M$) expresses a testing sample. Then L individual classifiers are produced based on the training subsets from $\mathbf{X}$ in an ensemble system, shown by a set $\Psi=\{\mathscr{L}_1,\ldots,\mathscr{L}_L\}$. In general, the error rate of an ensemble is calculated based on the difference between the final predictive labels and the true labels:

$$R_{enc} = \frac{1}{4N} \sum_{n=1}^{N} \|H_{enc}(\mathbf{x}_n) - y_n\|^2 \tag{7}$$

where R_{enc} expresses the error rate of an ensemble, $H_{enc}(\mathbf{x}_n)$ denotes the predictive label of a sample $\mathbf{x}_n$ gained by an ensemble, and N is the number of all samples. According to Eq.(2), Eq.(7) is also presented by the following formula:

$$R_{enc} = \frac{1}{4N} \sum_{n=1}^{N} \left[\mathrm{sgn}\left(\sum_{i=1}^{L} f_{ni} * w_i\right) - y_n \right]^2 = \frac{1}{4N} \|\mathrm{sgn}(\mathbf{Fw}) - \mathbf{y}\|^2 \tag{8}$$

where $\mathbf{w}$ denotes a weight vector of individual classifiers, $\mathbf{w}=[w_1,\ldots,w_L]^T$, $\mathbf{y}$ is a vector of the true labels, $\mathbf{y}=[y_1,\ldots,y_N]^T$, f_{ni} ($f_{ni} \in [-1,+1]$) expresses a predictive label of $\mathbf{x}_n$ given by $\mathscr{L}_i$, and $\mathbf{F}$ is a matrix with $N \times L$ elements consisting of f_{ni} (f_{ni} is the element in the nth row and the ith column of the matrix $\mathbf{F}$). In fact, $\mathbf{F}$ is a set of predictive labels of all individual classifiers, $\mathbf{F}=[\mathbf{f}_1,\ldots,\mathbf{f}_L]$, where $\mathbf{f}_i$ expresses a vector of predictive labels gained by $\mathscr{L}_i$, $\mathbf{f}_i=[f_{1i},\ldots,f_{Ni}]^T$, $i=1,\ldots,L$. According to Eq. (8), it indicates that constructing a good algorithm should seek an optimal weight vector $\mathbf{w}$ with which the error rate R_{enc} is minimized, specifically shown in the following formula:

$$\min \ \frac{1}{4N} \|\mathrm{sgn}(\mathbf{Fw}) - \mathbf{y}\|^2. \tag{9}$$

However, it is tough to gain the optimal solution from the above formula directly because of the sign function in Eq. (9). Hence, an approximation form is constructed in this paper in order to handle this problem, shown as follows:

$$\begin{aligned} \min \quad & \frac{1}{4N} \|\mathbf{Fw} - \mathbf{y}\|^2 \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{L} w_i = 1 \\ & -1 < w_i < 1 \end{aligned} \tag{10}$$

Compared with Eq. (9), two constraints ($\sum w_i = 1$ and $-1 < w_i < 1$) are added in Eq. (10). In fact, we can show that $\mathbf{Fw}$ is equivalent to an approximation of $\mathrm{sgn}(\mathbf{Fw})$ based on two constraints. Because it is easy to make the range of $\mathbf{F}_n\mathbf{w}$ with $\sum w_i = 1$ and $-1 < w_i < 1$, $-1 \le \mathbf{F}_n\mathbf{w} \le 1$, where $\mathbf{F}_n$ expresses a set of all predictive labels of a sample $\mathbf{x}_n$ obtained by all individual classifiers, $\mathbf{F}_n=[f_{n1},\ldots,f_{nL}]$, additionally, $\mathbf{F}=\left[\mathbf{F}_1^T,\ldots,\mathbf{F}_N^T\right]^T$. Therefore, Eq. (10) is utilized to seek the optimal weight vector instead of Eq. (9), and an approximated error R^*_{enc} is used instead of R_{enc}, where $R^*_{enc} = \|\mathbf{Fw}-\mathbf{y}\|^2/4N$. Especially, R^*_{enc} is considered as the target function of the optimization. From Eq. (10), it is obviously seen that $\|\mathbf{Fw}-\mathbf{y}\|^2$ represents the squared error between $\mathbf{Fw}$ and $\mathbf{y}$. Thus a function S is given to represent the mean squared error, $S = \|\mathbf{Fw}-\mathbf{y}\|^2/N$, and minimizing R^*_{enc} is actually equal to minimizing S, as shown in the follows:

$$\begin{aligned} \min \quad & S \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{L} w_i = 1 \\ & -1 < w_i < 1 \end{aligned} \tag{11}$$

Specifically, a weight vector $\mathbf{w}_0$ is introduced and adopted in the function S, and then a special formula is made based on $\mathbf{w}_0$:

$$\begin{aligned} S &= \frac{1}{N}\|\mathbf{Fw}-\mathbf{y}\|^2 = \frac{1}{N}\|\mathbf{Fw}-\mathbf{y}+\mathbf{Fw}_0-\mathbf{Fw}_0\|^2 \\ &= \frac{1}{N}\left[\|\mathbf{Fw}-\mathbf{Fw}_0\|^2 + 2(\mathbf{Fw}-\mathbf{Fw}_0)^T(\mathbf{Fw}_0-\mathbf{y})\right] + \frac{1}{N}\|\mathbf{Fw}_0-\mathbf{y}\|^2 \end{aligned} \tag{12}$$

In Eq. (12), it is obvious that $\mathbf{Fw}_0$ denotes a set of predictive labels of an ensemble with the weight vector $\mathbf{w}_0$ of individual

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

classifiers. In fact, the weight vector $\mathbf{w}_0$ can be considered as the initial weight vector of the proposed method here. On the one hand, we give the weight vector $\mathbf{w}_0$ by artificial initialization, such as $\mathbf{w}_0 = [1/L, \ldots, 1/L]^T$. On the other hand, the weight vector $\mathbf{w}_0$ also comes from the weight vectors obtained by other ensemble algorithms. But it is worth notice that the weight vector $\mathbf{w}_0$ must satisfy for two constraints $\sum w_{0i} = 1$ and $-1 < w_{0i} < 1$, $i = 1, \ldots, L$. Similar to the function S, a function S_0 is given based on $\mathbf{w}_0$ from Eq. (12), $S_0 = \|\mathbf{F}\mathbf{w}_0 - \mathbf{y}\|^2/N$, and its value represents the mean squared error between $\mathbf{F}\mathbf{w}_0$ and $\mathbf{y}$. In other words, the value of S_0 actually indicates the error rate of an ensemble with the weight vector $\mathbf{w}_0$, naturally, which is independent of the weight vector $\mathbf{w}$. Consequently, an equation is gained by Eq. (12), shown as follows:

$$S = S_0 - \Phi(\mathbf{w}) \tag{13}$$

According to Eqs. (12) and (13), $\Phi(\mathbf{w})$ is given by the following formula:

$$\begin{aligned}\Phi(\mathbf{w}) &= \frac{1}{N}\left[-\|\mathbf{F}\mathbf{w} - \mathbf{F}\mathbf{w}_0\|^2 - 2(\mathbf{F}\mathbf{w} - \mathbf{F}\mathbf{w}_0)^T(\mathbf{F}\mathbf{w}_0 - \mathbf{y})\right] \\ &= \frac{1}{N}\left(-\mathbf{w}^T\mathbf{F}^T\mathbf{F}\mathbf{w} + 2\mathbf{w}^T\mathbf{F}^T\mathbf{y} - 2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y} + \mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0\right)\end{aligned} \tag{14}$$

In fact, the initial motivation of introducing a weight vector $\mathbf{w}_0$ is to seek a weight vector $\mathbf{w}$, and it is made that an ensemble equipped the weight vector $\mathbf{w}$ can obtain a lower classification error than the weight vector $\mathbf{w}_0$. Therefore it is expected that $\Phi(\mathbf{w})$ is more than zero. According to Eq. (13), it is easily seen that $S < S_0$ is given when $\Phi(\mathbf{w}) > 0$, and $S > S_0$ when $\Phi(\mathbf{w}) < 0$. Obviously, it illustrates that the mean squared error gained based on $\mathbf{w}$ is smaller than the one based on $\mathbf{w}_0$ when $\Phi(\mathbf{w}) > 0$. Similarly, it also indicates that the error R^*_{enc} given by $\mathbf{w}$ will be lower than the error R^*_0 given by $\mathbf{w}_0$ if $\Phi(\mathbf{w}) > 0$. Hence, maximizing the function $\Phi(\mathbf{w})$ is crucial for seeking a better weight vector than the initial weight vector $\mathbf{w}_0$. Then the process of minimizing the approximation form R^*_{enc} is transformed into maximizing the function $\Phi(\mathbf{w})$. As follows, three different quadratic forms are respectively constructed to achieve the maximization of $\Phi(\mathbf{w})$ for gaining the optimal weight vector $\mathbf{w}$, and the detailed descriptions are shown in following sections. The algorithms constructed based on the three quadratic forms are named by QFWEC1, QFWEC2 and QFWEC3 methods, respectively.

3.1. QFWEC1 method

According to the above analysis, it demonstrates that maximizing the function $\Phi(\mathbf{w})$ is pivotal to seek an optimal weight vector $\mathbf{w}$ of individual classifiers. Therefore, a quadratic form $Q_1(\mathbf{v})$ is constructed to represent the function $\Phi(\mathbf{w})$, shown as follows:

$$Q_1(\mathbf{v}) = \mathbf{v}^T\mathbf{M}_1\mathbf{v} = \sum_{i=1}^{L+1}\sum_{j=1}^{L+1} m_{ij}v_iv_j \tag{15}$$

where $\mathbf{v}$ is a vector with $(L+1)\times 1$ and $\mathbf{M}_1$ is a matrix with $(L+1)\times(L+1)$. Specially, we give $v_1 = 1$, $v_{k+1} = w_k (k = 1, \ldots, L)$ and a real symmetric matrix $\mathbf{M}_1$, shown as follows:

$$\mathbf{M}_1 = \frac{1}{N}\begin{bmatrix}\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathrm{w}_0 - 2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y} & \mathbf{y}^T\mathbf{F} \\ \mathbf{F}^T\mathbf{y} & -\mathbf{F}^T\mathbf{F}\end{bmatrix} \tag{16}$$

Based on Eqs.(14)–(16), we obtain the following equation easily:

$$\Phi(\mathbf{w}) = Q_1(\mathbf{v}). \tag{17}$$

Thus the optimization problem is converted into maximizing the quadratic form $Q_1(\mathbf{v})$

$$Q_1(\mathbf{v}) = \mathbf{v}^T\mathbf{M}_1\mathbf{v} = \begin{bmatrix}1 \\ \mathbf{w}\end{bmatrix}^T\begin{bmatrix}\frac{\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0 - 2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y}}{N} & \frac{\mathbf{y}^T\mathbf{F}}{N} \\ \frac{\mathbf{F}^T\mathbf{y}}{N} & -\frac{\mathbf{F}^T\mathbf{F}}{N}\end{bmatrix}\begin{bmatrix}1 \\ \mathbf{w}\end{bmatrix} \tag{18}$$

In particular, eigenvalue decomposition is employed to obtain the optimal solution from the above formula, which is rewritten as

$$\mathbf{M}_1\mathbf{v} = Q_1(\mathbf{v})\mathbf{v}. \tag{19}$$

According to Eq. (19), the optimal solution is actually equal to the eigenvector corresponding to the largest eigenvalue of the matrix $\mathbf{M}_1$. It is known that the eigenvector is satisfied for $\mathbf{v}^T\mathbf{v} = 1$ in eigenvalue decomposition, and we can gain $-1 < w_i < 1$ based on $\mathbf{v}^T\mathbf{v} = 1$ and $v_{k+1} = w_k$ $(k = 1, \ldots, L)$, which illustrates that the constraint $-1 < w_i < 1$ of Eq. (10) is satisfied in Eq. (19). Nevertheless, the other equation is also given by $\mathbf{v}^T\mathbf{v} = 1$ because of $v_1 = 1$: $1 + \sum_{i=1}^{L} w_i^2 = 1$, which indicates that $\sum_{i=1}^{L} w_i^2 = 0$ is set up in QFWEC1 method. Unfortunately, $\sum_{i=1}^{L} w_i^2 = 0$ contradicts with a constraint $\sum_{i=1}^{L} w_i = 0$ of Eq. (10). It means that the optimal weight vector gained by QFWEC1 method may satisfy only one constraint $(-1 < w_i < 1)$ of Eq. (10), and QFWEC1 method neglects the other constraint $(\sum_{i=1}^{L} w_i = 1)$ in fact.

3.2. QFWEC2 method

According to Section 3.1, it illustrates that QFWEC1 method is inadequate. Because a constraint of Eq. (10) is not satisfied in the process of maximizing the quadratic form $Q_1(\mathbf{v})$. In order to improve the QFWEC1 method, QFWEC2 method is introduced in this part. In QFWEC2 method, a quadratic form $Q_2(\mathbf{v})$ is given as follows:

$$Q_2(\mathbf{v}) = \mathbf{v}^T\mathbf{M}_2\mathbf{v} = \begin{bmatrix}\sigma \\ \mathbf{w}\end{bmatrix}^T\begin{bmatrix}\frac{\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0 - 2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y}}{N\sigma^2} & \frac{\mathbf{y}^T\mathbf{F}}{N\sigma} \\ \frac{\mathbf{F}^T\mathbf{y}}{N\sigma} & -\frac{\mathbf{F}^T\mathbf{F}}{N}\end{bmatrix}\begin{bmatrix}\sigma \\ \mathbf{w}\end{bmatrix}, \tag{20}$$

where $v = [\sigma \quad \mathbf{w}^T]^T$, $0 < \sigma < 1$. In the quadratic form $Q_2(\mathbf{v})$, we give $v_1 = \sigma$ instead of $v_1 = 1$ of QFWEC1 method, and thus a different symmetric matrix $\mathbf{M}_2$ is gained:

$$\mathbf{M}_2 = \frac{1}{N}\begin{bmatrix}\frac{\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0 - 2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y}}{\sigma^2} & \frac{\mathbf{y}^T\mathbf{F}}{\sigma} \\ \frac{\mathbf{F}^T\mathbf{y}}{\sigma} & -\mathbf{F}^T\mathbf{F}\end{bmatrix} \tag{21}$$

According to Eq. (20), it is obvious that $\Phi(\mathbf{w}) = Q_2(\mathbf{v})$ as well as QFWEC1 method. Then the optimal weight vector of individual classifiers is gained based on the following formula:

$$\begin{aligned}&\max \quad \mathbf{v}^T\mathbf{M}_2\mathbf{v} \\ &\text{s.t.} \quad \sigma^2 + \textstyle\sum_{i=1}^{L} w_i^2 = 1\end{aligned} \tag{22}$$

In fact, the maximization of Eq. (22) is also equivalent to eigenvalue decomposition of the matrix $\mathbf{M}_2$ as similar as QFWEC1 method. The optimal solution is equal to the eigenvector corresponding to the largest eigenvalue of $\mathbf{M}_2$ in QFWEC2 method. In particular, when $v_1 = \sigma$ $(0 < \sigma < 1)$, both $\sum_{i=1}^{L} w_i = 1$ and $\sigma^2 + \sum_{i=1}^{L} w_i^2 = 1$ are potentially set up at the same time, and the illustration about two constraint is shown in detail in Appendix A.

3.3. QFWEC3 method

From Section 3.2, QFWEC2 method has improved QFWEC1 method by giving $v_1 = \sigma$ $(0 < \sigma < 1)$, and it also applies eigenvalue decomposition of $\mathbf{M}_2$ to gain the optimal solution of Eq. (22).

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

However, it is known that only one constraint ($\mathbf{v}^T\mathbf{v}=1$) needs to be satisfied in eigenvalue decomposition essentially. It means that only the constraint ($-1<w_i<1$) is emphasized and utilized in QFWEC2 method, and the constraint ($\sum_{i=1}^{L} w_i=1$) is neglected. Hence, in order to consider directly the constraint ($\sum w_i=1$) which is not mentioned in optimization of QFWEC1 and QFWEC2, we add $\sum w_i=1$ into Eq. (22) of QFWEC2, and then a new formula is given:

$$\begin{array}{ll}\max & \mathbf{v}^T\mathbf{M}_2\mathbf{v} \\ \text{s.t.} & \sigma^2+\sum_{i=1}^{L} w_i^2=1 \\ & \sum_{i=1}^{L} w_i=1\end{array} \tag{23}$$

Specifically, Eq. (23) is recast into the below formula by adding a regularization term:

$$\begin{array}{ll}\max & \mathbf{v}^T\mathbf{M}_2\mathbf{v}+\lambda(\mathbf{v}^T\hat{\mathbf{e}}-1) \\ \text{s.t.} & \sigma^2+\sum_{i=1}^{L} w_i^2=1\end{array} \tag{24}$$

where λ ($\lambda>0$) is a constant that is given artificially, and $\mathbf{e}$ is a vector with $L\times 1$ in which each element is equal to 1, $\hat{\mathbf{e}}=[0 \quad \mathbf{e}]^T$. It is obvious that the term ($\mathbf{v}^T\hat{\mathbf{e}}-1$) is viewed as the regularization term in Eq. (24) and $\mathbf{v}^T\hat{\mathbf{e}}=\sum_{i=1}^{L} w_i$. According to Eq. (24), $Q_3(\mathbf{v})$ is given by the below formula based on $\Phi(\mathbf{w})=Q_3(\mathbf{v})$:

$$Q_3(\mathbf{v}) = \mathbf{v}^T\mathbf{M}_3\mathbf{v}=\begin{bmatrix}\sigma \\ \mathbf{w}\end{bmatrix}^T\begin{bmatrix}\frac{\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0-2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y}-\lambda}{N\sigma^2} & \frac{\mathbf{y}^T\mathbf{F}}{N\sigma} \\ \frac{\mathbf{F}^T\mathbf{y}+\lambda\mathbf{e}}{N\sigma} & -\frac{\mathbf{F}^T\mathbf{F}}{N}\end{bmatrix}\begin{bmatrix}\sigma \\ \mathbf{w}\end{bmatrix}, \tag{25}$$

where $v=[\sigma \quad \mathbf{w}^T]^T$ $(0<\sigma<1)$, and the matrix $\mathbf{M}_3$ is shown:

$$\mathbf{M}_3=\frac{1}{N}\begin{bmatrix}\frac{\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{F}\mathbf{w}_0-2\mathbf{w}_0{}^T\mathbf{F}^T\mathbf{y}-\lambda}{\sigma^2} & \frac{\mathbf{y}^T\mathbf{F}}{\sigma} \\ \frac{\mathbf{F}^T\mathbf{y}+\lambda\mathbf{e}}{\sigma} & -\mathbf{F}^T\mathbf{F}\end{bmatrix}. \tag{26}$$

Similarly, an optimal weight vector $\mathbf{w}$ is sought based on maximizing $Q_3(\mathbf{v})$, named by QFWEC3 method. Certainly, the optimal solution of QFWEC3 method is also obtained by eigenvalue decomposition of $\mathbf{M}_3$ as well as QFWEC1 and QFWEC2 methods. Then the eigenvector corresponding to the largest eigenvalue of $\mathbf{M}_3$ is equivalent to the optimal solution of QFWEC3 method. Finally, the detailed procedure of the proposed method is shown in Algorithm 1.

Algorithm 1. Weighted classifiers ensemble based on quadratic form (QFWEC method)

Input: A training set $\mathbf{X}_{trn}$ with N samples, $\mathbf{X}_{trn}=[\mathbf{x}_1,\ldots,\mathbf{x}_N]^T$, $\mathbf{x}_n\in\mathbb{R}^d$ $(n=1,\ldots,N)$; $\mathbf{y}$ is a set of the true labels of training samples, $\mathbf{y}=[y_1,\ldots,y_N]^T$; a testing set $\mathbf{X}_{tst}$ with M samples without true labels, $\mathbf{X}_{tst}=[\mathbf{x}_1,\ldots,\mathbf{x}_M]^T$, $\mathbf{x}_m\in\mathbb{R}^d$ $(m=1,\ldots,M)$; L is the number of individual classifiers; $\mathbf{w}_0$ is a given weight vector of classifiers; two parameters of QFWEC, σ and λ.

Procedure:

for $i=1,\ldots,L$

1. Obtain a training subset $\mathbf{X}_i$ from $\mathbf{X}_{trn}$ by an ensemble strategy;
2. Produce an individual classifier $\mathscr{L}_i$ by a basic classifier algorithm based on $\mathbf{X}_i$;
3. Gain the predictive labels $\mathbf{f}_i$ and $\mathbf{f}_i^t$ by using $\mathscr{L}_i$ to classify for $\mathbf{X}_{trn}$ and $\mathbf{X}_{tst}$ respectively, $\mathbf{f}_i\in\mathbb{R}^N$, $\mathbf{f}_i^t\in\mathbb{R}^M$;

end for

4. Give the set $\mathbf{F}$ of predictive labels, $\mathbf{F}=[\mathbf{f}_1,\ldots,\mathbf{f}_L]$;
5. Compute the matrix $\mathbf{M}$ based on $\mathbf{F}$, $\mathbf{y}$, $\mathbf{w}_0$, σ and λ by Eqs. (16), (21) or (25), and construct the quadratic form with the weight vector of individual classifiers and the matrix $\mathbf{M}$;
6. Obtain the optimal solution $\mathbf{v}_{opt}$ by Eqs. (19), (22) or (24), $\mathbf{v}_{opt}=[v_{opt1},\ldots,v_{optL}]^T$;
7. Compute the optimal weight vector $\mathbf{w}^*$ based on $\mathbf{v}_{opt}$: $w_i^*=(v_1/v_{opt1})v_{opt(i+1)}$ ($v_1=1$ or σ);
8. Combine individual classifiers with the optimal vector $\mathbf{w}^*$: $\mathbf{f}_{enc}=\mathbf{F}*\mathbf{w}^*$;

Output: The final predictive label of a testing sample $\mathbf{x}_m$ obtained by a classifier ensemble:
$H(\mathbf{x}_m)=\mathrm{sgn}\left(\sum_{i=1}^{L} f_{mi}^t(\mathbf{x}_m)*w_i^*\right)$.

3.4. Computational complexity

In this section, we analyze and discuss the computational complexity of the proposed methods. Normally, for an ensemble system, its computational complexity is consisted of two parts. One is corresponding to the process of producing individual classifiers, and the other is corresponding to the process of combining individual classifiers. Because QFWEC algorithms focus on the latter, we give only the computational complexity about the part of combining classifiers. We assume that the number of training samples is N and the number of individual classifiers is L, $N>L$. First, for the step 5 of Algorithm 1, the computational complexity of computing the matrix $\mathbf{M}$ by Eqs. (16), (21) or (26) is $O(NL^2+(L+1)^2)$. Second, in the step 6, the complexity is equal to the eigenvalue decomposition of the matrix $\mathbf{M}$, $O((L+1)^3)$. Finally, the complexity is $O(L)$ in the step 7 and it is $O(NL)$ in the step 8. Thus the total computational complexity of the proposed methods is $O(NL^2+(L+1)^3)$ in the part of combining individual classifiers.

4. Experiments and analysis

In order to validate the classification performance of the proposed method, three experiments are performed on an artificial dataset, UCI datasets [27] and PolSAR image [32], respectively. The detailed descriptions of experimental datasets are shown in the following parts. In this section, all algorithms are implemented by Matlab R2010b and all numerical experiments are performed on a desktop with HP dc7700 1.86 GHz Intel Core2, 2G memory with Windows XP 32bit operation system.

In the proposed method, it emphasizes on how to combining individual classifiers can obtain better classification performance, rather than how to produce individual classifiers. Consequently, a simple and classical ensemble strategy is employed to construct the training sample subset of each individual in our experiments and it is bagging strategy [2]. But it does not mean that the proposed method is unsuitable for other ensemble strategies, such as random subspace strategy [28], rotation forest [9] and so on. In fact, there are two advantages for using a simple ensemble strategy. First, a simple ensemble strategy is easily implemented and can obtain less computation complexity and space than some complicated strategies. Second, the performance of the proposed method can be better and more fairly exhibited by using a simple ensemble strategy. If a specified excellent ensemble strategy is employed to obtain individual classifiers, the performance's improvement may be attributed to the application of that specified excellent strategy. In our case, decision tree C4.5 [29] with back-pruning is applied as the basic classifier models of ensemble for artificial and UCI datasets, and SVM classifier [30] is used in the experiment of PolSAR image classification. For each dataset, the optimal parameters of basic classifier algorithms are given by 10-fold cross validation in experiments.

Furthermore, in order to elucidate the performance of the proposed method, we compare the classification performance of

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

the weighted classifier ensemble methods based on QFWEC1, QFWEC2 and QFWEC3 with the following methods:

1. *Single classifier algorithm*: The original training sample set is learned by a single classifier, and 'C4.5' or 'SVM' denotes a single classifier algorithm in experimental results;
2. *Simple vote rule* (*SV*) [8,11]: It combines actually individual classifiers with a weight vector in which all elements are same. In our experiments, each element of the weight vector is assigned to $w_i = 1$ $(i = 1, \ldots, L)$;
3. *Weighted majority vote* (*WMV*) [8]: It gains the weight coefficients of individual classifiers based on the probability of correct classification for training samples obtained by each individual.
4. *AdaBoost strategy* [14]: It produces an individual based on samples classified incorrectly in an iteration, and it computes the weight coefficients of individuals according to their classification error.
5. *Evolutionary Ensemble classifiers* (*EVEN*) [17]: It gains the weight coefficients of classifiers by genetic algorithm. In experiments, the parameters of EVEN algorithm are crossover probability (*cp*), mutation probability (*mp*) and selection probability (*sp*), and the values of three parameters are $cp = 0.8$, $mp = 0.01$ and $sp = 0.19$, respectively. Additionally, EVEN is run for 1000 generations with a population size of 250 in experiments.
6. *Naive Bayes combination method* (*NBC*) [8]: It gives a sample a predictive class label with the maximum probability obtained according to the classification probability of each individual for each class.
7. *Combining classifiers by using correspondence analysis* (*SCANN*) [39]: It applies the strategies of stacking and correspondence analysis to model the relationship between the learning samples and their predictions from the combination of learned models, and then the final prediction of a testing sample is given by a nearest neighbor method.

In the proposed methods, the values of parameters σ and λ are decided based on each dataset in our experiments. Generally, we give $\sigma \in \{0.1, 0.2, 0.3\}$ and $\lambda = \tau * N$, where $\tau \in \{0.1, 0.3\}$ and N is the number of training samples. Moreover, we set a weight vector $\mathbf{w}_0$ as the introduced weight vector of QFWEC algorithms, where $w_{0i} = 1/L$ $(i = 1, \ldots, L)$, and it is obvious that the weight vector $\mathbf{w}_0$ is satisfied for $\sum_{i=1}^{L} w_{0i} = 1$ and $-1 < w_{0i} < 1$. The reason of adopting it is just that it is simple and fair to compare the proposed method against other weight ensemble methods. Whereas, it does not mean that other weight vectors are not utilized in the proposed method. But notice that the used weight vector must be satisfied for two constraints ($\sum_{i=1}^{L} w_{0i} = 1$ and $-1 < w_{0i} < 1$). In addition, combining individual classifiers with $\mathbf{w}_0$ $(w_{0i} = 1/L)$ is equivalent to combine classifiers by simple vote rule strategy, because simple vote rule strategy combines individual classifiers with same coefficients. Thus the results of SV are equivalent to the results of combining classifiers with $\mathbf{w}_0$ $(w_{0i} = 1/L)$.

4.1. Experiments on an artificial dataset

Because the low dimension dataset can be visualized, we employ initially an artificial dataset to make an experiment, and this experimental artificial dataset belongs to the hyperbolic distribution. In this experiment, 1000 training samples and 1000 testing samples are produced randomly, and their distributions are respectively shown in Fig. 1. In Fig. 1, the red points indicate positive samples and the blue points indicate negative samples. For training and testing sets, the number of samples that belong to each class is a half of all samples. In the experiment, C4.5 classifier is applied as the basic classifier model and bagging strategy is adopted to obtain training subset of each classifier.

In Table 1, it is shown that the mean and standard deviation of error rates of classification gained by nine ensemble algorithms, which are the results of 50 times ensemble with 20 individual classifiers. From the results, it is seen that QFWEC2 and QFWEC3 obtain the lowest error rates of classification for both training and testing sets compared against other algorithms, and only NBC and

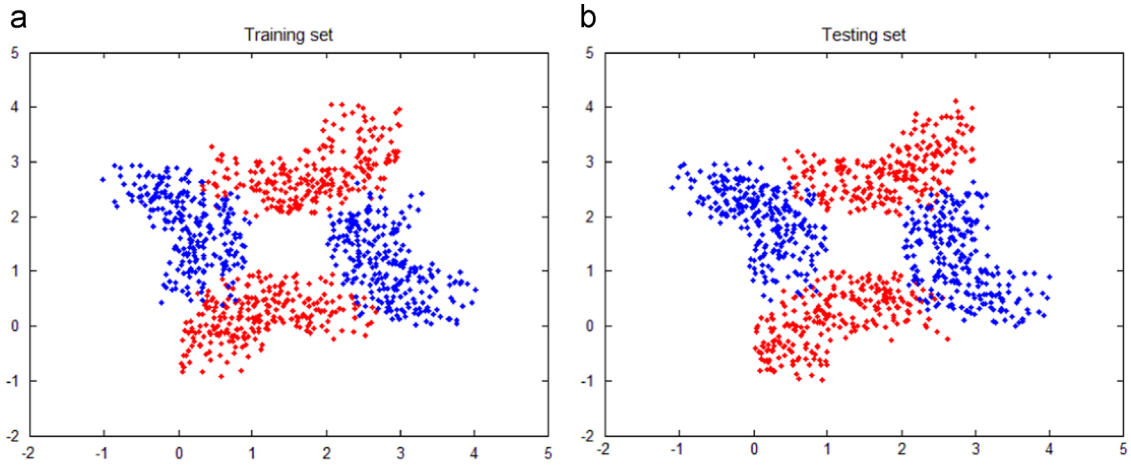

Fig. 1. Distributions of training set and testing set of Hyperbola Dataset. (a) Training set and (b) testing set. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Table 1
Error rates (%) obtained by eight algorithms for Hyperbola Dataset.

Methods	SV	WMV	AdaBoost	NBC	EVEN	SCANN	QFWEC1	QFWEC2	QFWEC3
Training	27.78 ± 6.47	27.60 ± 5.54	27.80 ± 5.31	18.83 ± 4.21	26.45 ± 6.04	18.95 ± 4.29	38.47 ± 7.69	$\mathbf{16.99 \pm 2.13}$	$\mathbf{16.98 \pm 2.13}$
Testing	29.22 ± 6.63	29.00 ± 5.33	29.10 ± 5.22	20.42 ± 3.98	27.70 ± 5.89	20.67 ± 4.08	38.54 ± 7.43	$\mathbf{18.87 \pm 1.87}$	$\mathbf{18.83 \pm 1.88}$

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

SCANN obtain the better performance in all compared algorithms. It indicates that the weight vector obtained by the proposed method represents effects of individual classifiers better than other algorithms. Statistically speaking, the boxplots are shown in Fig. 2 in order to show the performance of nine algorithms. The horizontal-axis expresses the ensemble algorithm and the vertical-axis expresses the error rate of classification. In Fig. 2 (a) and (b) are results of the training set and the testing set, respectively. From the results, it illustrates obviously that the proposed methods obtain the best performance in all methods.

Because the artificial dataset has only two dimensions, the distribution of samples is easily shown by a two-dimensional picture. Hence, we give distribution diagrams of the testing set with predictive labels obtained by nine algorithms in one ensemble, shown in Fig. 3. In particular, the misclassified samples are labeled in cyan. In Fig. 3, it is easily found that the points classified incorrectly by QFWEC2 and QFWEC3 are less than other algorithms. In addition, the error rates obtained by nine algorithms are 23.8% (SV), 32.6% (WMV), 32.6% (AdaBoost), 21.1% (NBC), 21.8% (EVEN), 23% (SCANN), 38.6% (QFWEC1), 16.8% (QFWEC2) and 16.8% (QFWEC3) in this ensemble, respectively. In summary, it demonstrates that the proposed methods are superior to other algorithms in the classification performance for the hyperbola dataset according to all experimental results.

However, it is found that QFWEC1 obtains the worse performance than others from the results. By the analysis of the experimental results, it is the reason that some classifiers with the poor performance are given higher weight coefficients by QFWEC1. In addition, the sums of weight vectors obtained by three proposed methods are shown in Fig. 4, and Fig. 4 shows the results of 10 times ensemble. The horizontal-axis expresses each ensemble and the vertical-axis expresses the sum of the weight vector obtained by QFWEC algorithms in each ensemble. From the results, it is obviously seen that QFWEC1 gains the large sum of weight coefficients and the sum is far greater than one. But the sums of weight vectors obtained by both QFWEC2 and QFWEC3 are almost close to 1, which indicates that two constraints ($-1 \le w_i \le 1$ and $\sum w_i = 1$) are satisfied by QFWEC2 and QFWEC3. Thus QFWEC2 and QFWEC3 make the better performance than QFWEC1.

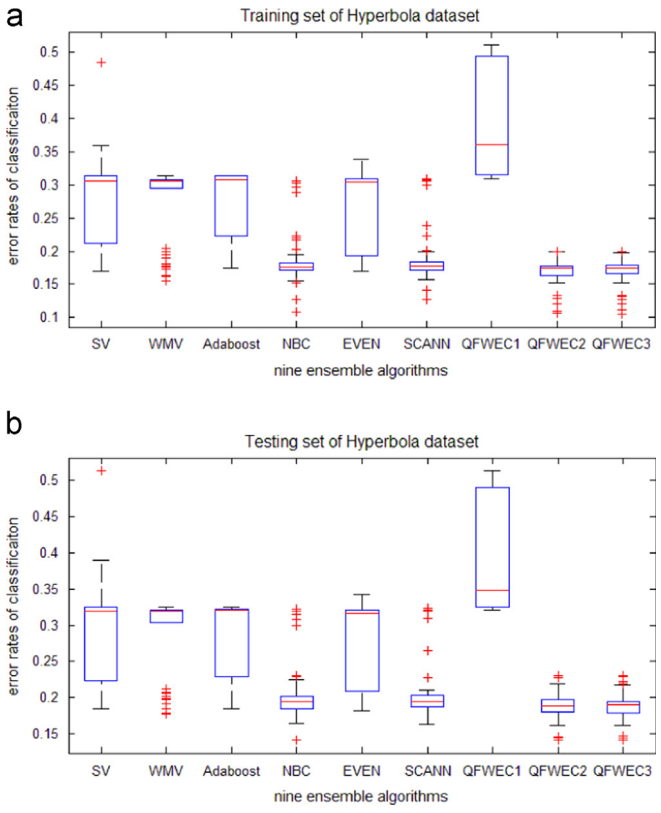

Fig. 2. Error rates of classification obtained by nine ensemble algorithms. (a) Training set and (b) testing set.

4.2. Experiments on UCI datasets

In this section, we use 16 UCI datasets in our experiment and the detailed descriptions of the datasets are shown in Table 2. Table 2 presents some attributes of 16 UCI datasets, where *N* is the sample number of a dataset, *Feature* is the feature number of a dataset, and *class* is the class number of a dataset. In this experiment, C4.5 classifier is used as the basic classifier model and bagging is employed to obtain the training subsets, as similar as Section 4.1. In Tables 2, 'C4.5' indicates the error rate of classification of each dataset gained by a single C4.5 classifier based on 10-fold cross validation. In addition, we implement the experiments with 20, 50 and 200 individual classifiers ensemble, and the experimental results are shown in Tables 3–5, respectively. Because the proposed methods are introduced for the classification problem with two classes, we employ the one-against-one rule [31] to deal with the multiclass datasets.

4.2.1. Classification performance

In this section, the experimental results of 16 UCI datasets are averaged over 10-fold cross validation performed 10 times. Table 3 shows the experimental results of ensemble 20 classifiers by the proposed methods and several compared ensemble algorithms, Table 4 shows the results of ensemble 50 classifiers, and Table 5 shows the results of ensemble 200 individuals. The result of each dataset is shown by the mean and standard deviation of classification error rates (%) on the test sets. In tables, QFWEC1, QFWEC2 and QFWEC3 are three proposed algorithms, respectively, and SV, WMV, AdaBoost, NBC, EVEN and SCANN are six comparing ensemble algorithms. Additionally, a result is bolded in each column of the table when it is the lowest error rate in results of nine algorithms.

From the results of Tables 3–5, it is seen easily that QFWEC2 and QFWEC3 outperform other ensemble algorithms at the classification performance, and they can gain the lowest error rate for most datasets in 16 datasets. In detail, it is found that the proposed methods gain the lowest error rate for 10, 12 and 14 datasets for combining 20, 50 and 200 individual classifiers, respectively. In Table 3, we see that the comparing algorithms WMV, AdaBoost and NBC are respectively superior to the proposed method only on one datasets, and only SCANN algorithm outperforms for two datasets. Unfortunately, SV and EVEN algorithms are inferior to the proposed methods for all datasets. Similarly to Table 3, the results of Table 4 show that AdaBoost, EVEN and SCANN algorithm outperform for only one datasets, respectively, and none of SV, WMV and NBC algorithms win for any datasets. For Table 5, only AdaBoost and SCANN gains two better results than the proposed method, and other algorithms are inferior to our method. In short, according to the experimental results of classification error, it illustrates the proposed method which obtains the optimal weight vector of individual classifiers based on minimizing the ensemble error can improve effectively the classification performance of an ensemble than other ensemble algorithms. Specially, it also indicates that our methods performed a better performance when more individuals are combined in an ensemble system according to all results shown in three tables. In addition, the average values

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

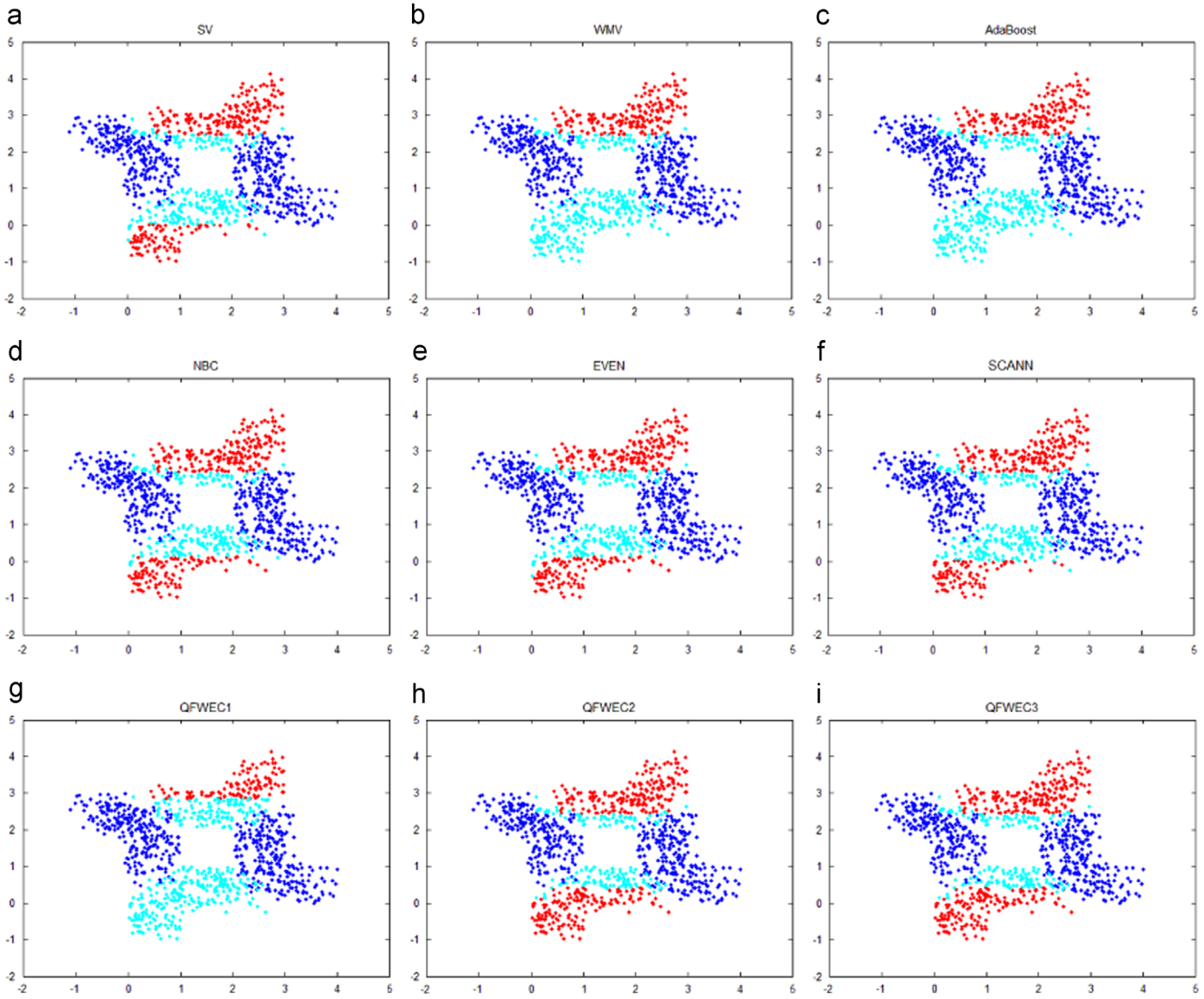

Fig. 3. Classification results of nine ensemble algorithms for Hyperbola Dataset in one ensemble. (a) SV, (b) WMV, (c) AdaBoost, (d) NBC, (e) EVEN, (f) SCANN, (g) QFWEC1, (h) QFWEC2, and (i) QFWEC3. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

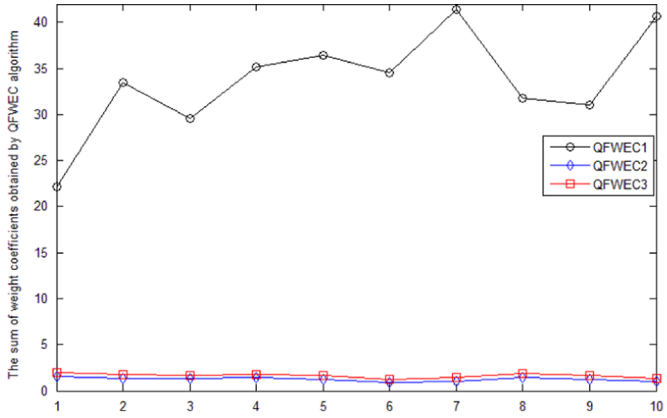

Fig. 4. Sum of weight vectors obtained by three proposed methods.

of error rates of 16 datasets obtained by nine algorithms are shown in Fig. 5. From Fig. 5, it is also obvious that QFWEC2 and QFWEC3 gain the lowest errors compared with other methods.

In order to demonstrate statistically the performance of nine algorithms, we use the *t*-test method to make a comparison with the proposed methods and others. In Table 6, it is shown that the win/tie/loss results based on *t*-test, computed at the 5% significance level. According to the results, we find that QFWEC2 exceeds SV, WMV, AdaBoost, NBC, EVEN and SCANN on 14, 14, 12, 14, 14 and 9 datasets, respectively, and QFWEC3 exceeds SV, WMV, AdaBoost, NBC, EVEN and SCANN on 13, 15, 12, 13, 14 and 9 datasets, respectively. Hence, the results demonstrate that our algorithms are superior to other compared algorithms.

In Tables 3–5, they show only the mean and standard deviation of error rates of 10 times ensemble. Hence, in order to show statistically the results of the proposed method, Fig. 6 shows the boxplot diagrams of error rates of 10 times ensemble with 200 individuals obtained by nine algorithms for eight datasets, and eight datasets are Australian, Breast, Dna, Glass, Heart, Liver, Pima and Wdbc, respectively. In Fig. 6, the horizontal-axis expresses the ensemble algorithm and the vertical-axis is the error rate of classification. From the results of Fig. 6, it is obviously seen that the boxplots of QFWEC2 and QFWEC3 are located below other methods, which indicates the proposed methods outperform

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

others. According to the points of both minimum and median of error rates, it illustrates that QFWEC2 or QFWEC3 has the lowest error rate in nine methods. In brief, it demonstrates the proposed method can obtain the better classification performance compared with other algorithms according to all experimental results of classification performance.

Table 2
Descriptions of UCI datasets used in experiments.

Datasets	*N*	*Feature*	*Class*	Error rate (%) C4.5
Air	1089	64	3	17.17
Australian	690	14	2	16.91
Breast	699	9	2	31.48
Dna	3186	180	3	9.81
Glass	214	9	7	40.00
Heart	270	13	2	23.70
HeartC	303	13	2	24.14
Ionosphere	351	32	2	6.18
Iris	150	4	3	4.67
Liver	345	6	2	38.24
Pima	768	8	2	27.37
Sonar	208	60	2	30.50
Vehicle	864	18	4	33.17
Vote	435	16	2	4.52
Wdbc	569	30	2	7.68
Wine	178	13	3	10.00

4.2.2. *Weight vector of individual classifiers*

In order to illustrate visually the performance of the proposed method, we make an experiment to observe the weight vector obtained by several ensemble methods. In this experiment, we use breast, heart and pima datasets as the observation datasets. Because the training and testing sets are changed in each group in 10-fold cross validation, it is difficult to decide that the result of which group is selected to show. Consequently, the training set and the testing set are separated with a half of all samples for each dataset in this experiment. The parameters of all algorithms are similar to the previous experiments, and the number of individual classifiers is 20.

In Fig. 7, it shows weight coefficients of individual classifiers obtained by five algorithms for three datasets in one ensemble, respectively. In (a)–(c) of Fig. 7, the left-upper sub-graph is the accuracy rate obtained by each individual classifier for training samples, and the rest of sub-graphs are the weight vectors obtained by WMV, EVEN and three QFWEC algorithms, respectively. From Fig. 7, it is seen that the left-upper sub-graphs which

Table 3
Error rates (%) of classification on test sets of 16 UCI datasets by 20 classifiers ensemble.

Datasets	SV	WMV	AdaBoost	NBC	EVEN	SCANN	QFWEC1	QFWEC2	QFWEC3
Air	16.94 ± 1.05	16.37 ± 0.92	**8.73 ± 0.67**	16.21 ± 1.24	17.36 ± 1.18	16.24 ± 1.30	16.78 ± 1.09	13.21 ± 0.92	13.46 ± 1.14
Australian	15.84 ± 0.31	15.24 ± 0.32	15.24 ± 0.50	15.06 ± 0.33	15.34 ± 0.53	15.10 ± 0.44	15.09 ± 0.38	**15.01 ± 0.36**	**15.03 ± 0.30**
Breast	30.56 ± 0.91	30.04 ± 0.93	30.15 ± 1.06	32.11 ± 1.18	29.70 ± 0.97	29.59 ± 1.46	30.26 ± 1.21	**28.30 ± 1.17**	**27.96 ± 1.60**
Dna	7.11 ± 0.08	7.11 ± 0.08	9.78 ± 0.21	7.11 ± 0.10	7.14 ± 0.11	7.15 ± 0.07	7.13 ± 0.06	**6.00 ± 0.25**	**5.98 ± 0.25**
Glass	34.78 ± 1.51	35.94 ± 1.46	34.22 ± 2.39	37.50 ± 1.96	35.50 ± 2.30	41.28 ± 5.49	36.72 ± 1.54	**34.00 ± 2.17**	**34.17 ± 2.08**
Heart	23.30 ± 0.95	22.59 ± 0.91	23.04 ± 1.28	22.67 ± 0.90	22.00 ± 0.93	21.59 ± 0.94	22.70 ± 1.00	**20.93 ± 1.34**	**20.81 ± 1.37**
HeartC	21.07 ± 1.88	20.38 ± 1.44	21.69 ± 1.84	20.62 ± 1.26	20.34 ± 1.50	**19.97 ± 1.42**	20.38 ± 1.71	20.34 ± 1.79	20.24 ± 1.21
Ionosphere	6.82 ± 0.43	6.56 ± 0.28	6.47 ± 0.34	6.56 ± 0.48	6.59 ± 0.35	6.56 ± 0.31	6.59 ± 0.28	**6.44 ± 0.32**	**6.38 ± 0.31**
Iris	3.93 ± 0.66	**3.47 ± 0.28**	4.60 ± 0.21	3.80 ± 0.45	4.13 ± 0.69	3.87 ± 0.61	3.73 ± 0.47	3.60 ± 0.47	3.60 ± 0.47
Liver	38.18 ± 0.27	38.03 ± 0.14	37.85 ± 0.39	37.79 ± 0.35	37.97 ± 0.22	**35.97 ± 0.57**	38.03 ± 0.24	37.82 ± 0.32	37.24 ± 0.35
Pima	27.21 ± 0.42	26.55 ± 0.43	26.61 ± 0.30	25.82 ± 0.44	26.76 ± 0.31	25.71 ± 0.32	26.87 ± 0.52	**25.62 ± 0.45**	**25.62 ± 0.43**
Sonar	30.00 ± 1.00	28.25 ± 1.36	28.55 ± 1.54	27.70 ± 1.78	27.80 ± 0.95	27.90 ± 1.52	28.25 ± 1.59	**27.35 ± 1.13**	**27.35 ± 1.29**
Vehicle	29.11 ± 0.63	29.12 ± 0.64	28.61 ± 0.49	29.13 ± 0.59	29.28 ± 0.70	29.12 ± 0.62	29.29 ± 0.53	**28.60 ± 0.69**	**28.59 ± 0.65**
Vote	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00
Wdbc	7.39 ± 0.41	7.07 ± 0.46	7.48 ± 0.30	7.23 ± 0.27	7.00 ± 0.38	7.04 ± 0.33	7.14 ± 0.42	**6.84 ± 0.40**	7.00 ± 0.43
Wine	9.06 ± 2.38	10.81 ± 1.06	8.31 ± 2.78	**7.56 ± 1.04**	8.81 ± 2.37	8.69 ± 2.13	8.81 ± 2.27	8.87 ± 1.17	8.87 ± 1.17

Note that the result is shown by (A ± B), A and B express the mean of classification error rate (%) and the standard deviation of classification error rate (%) on the test set of each dataset, respectively.

Table 4
Error rates (%) of classification on test sets of 16 UCI datasets by 50 classifiers ensemble.

Datasets	SV	WMV	AdaBoost	NBC	EVEN	SCANN	QFWEC1	QFWEC2	QFWEC3
Air	16.78 ± 0.85	16.14 ± 0.82	**7.96 ± 0.46**	15.88 ± 0.73	16.85 ± 0.60	15.81 ± 0.79	16.80 ± 0.79	11.45 ± 0.75	11.80 ± 0.73
Australian	15.90 ± 0.47	15.54 ± 0.49	15.56 ± 0.32	15.68 ± 0.46	15.51 ± 0.42	15.57 ± 0.46	15.54 ± 0.43	**14.90 ± 0.67**	**14.93 ± 0.71**
Breast	30.40 ± 0.76	30.15 ± 0.78	29.81 ± 1.47	31.07 ± 0.44	30.26 ± 1.10	30.33 ± 0.96	30.30 ± 0.90	**28.67 ± 1.06**	**28.07 ± 0.85**
Dna	7.07 ± 0.03	7.07 ± 0.03	9.73 ± 0.23	7.07 ± 0.03	7.08 ± 0.04	7.04 ± 0.04	7.08 ± 0.03	**5.81 ± 0.20**	**5.81 ± 0.20**
Glass	33.22 ± 1.30	32.94 ± 1.14	36.50 ± 0.95	34.94 ± 1.77	33.39 ± 1.00	35.61 ± 1.15	33.44 ± 1.25	**32.39 ± 1.36**	**32.33 ± 1.41**
Heart	21.90 ± 0.41	21.81 ± 0.48	21.89 ± 0.69	21.70 ± 0.63	21.63 ± 0.43	20.89 ± 0.68	22.11 ± 0.61	**20.19 ± 1.52**	**19.85 ± 1.65**
HeartC	20.80 ± 0.89	20.48 ± 1.06	21.34 ± 0.90	20.45 ± 1.35	20.10 ± 1.12	20.86 ± 1.93	20.52 ± 0.88	**18.93 ± 1.48**	**19.10 ± 1.02**
Ionosphere	6.47 ± 0.34	6.38 ± 0.24	6.32 ± 0.21	6.26 ± 0.31	6.44 ± 0.35	6.35 ± 0.32	6.41 ± 0.27	**6.18 ± 0.28**	**6.15 ± 0.32**
Iris	3.73 ± 0.56	3.67 ± 0.57	4.67 ± 0.00	4.13 ± 0.53	3.80 ± 0.55	3.73 ± 0.56	3.80 ± 0.71	**3.33 ± 0.00**	**3.33 ± 0.00**
Liver	38.20 ± 0.24	38.06 ± 0.21	38.03 ± 0.14	37.79 ± 0.49	38.03 ± 0.20	**36.15 ± 0.45**	38.03 ± 0.24	37.65 ± 0.24	37.74 ± 0.24
Pima	27.10 ± 0.34	26.67 ± 0.35	26.72 ± 0.40	25.66 ± 0.16	26.99 ± 0.25	25.39 ± 0.42	27.00 ± 0.40	**25.16 ± 0.40**	**25.05 ± 0.36**
Sonar	29.50 ± 0.85	29.15 ± 0.91	29.85 ± 1.31	29.15 ± 1.03	**28.15 ± 1.03**	28.50 ± 0.78	28.75 ± 1.27	28.45 ± 1.77	28.20 ± 1.14
Vehicle	29.04 ± 0.70	29.01 ± 0.74	28.11 ± 0.49	29.20 ± 0.81	29.05 ± 0.44	29.00 ± 0.61	29.05 ± 0.88	**28.01 ± 0.88**	**27.97 ± 0.82**
Vote	4.52 ± 0.00	4.52 ± 0.00	4.55 ± 0.08	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	**4.50 ± 0.00**
Wdbc	7.73 ± 0.49	7.57 ± 0.53	7.66 ± 0.34	7.79 ± 0.36	7.66 ± 0.43	7.54 ± ± 0.43	7.57 ± 0.49	**7.07 ± 0.36**	**7.09 ± 0.38**
Wine	8.19 ± 1.49	11.31 ± 1.27	8.44 ± 2.72	10.19 ± 1.29	**7.50 ± 0.88**	8.06 ± 1.60	8.00 ± 1.55	**7.50 ± 1.02**	7.55 ± 1.02

Note that the result is shown by (A ± B), A and B express the mean of classification error rate (%) and the standard deviation of classification error rate (%) on the test set of each dataset, respectively.

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

Table 5
Error rates (%) of classification on test sets of 16 UCI datasets by 200 classifiers ensemble.

Datasets	SV	WMV	AdaBoost	NBC	EVEN	SCANN	QFWEC1	QFWEC2	QFWEC3
Air	16.44 ± 0.43	15.70 ± 0.42	**7.64 ± 0.46**	15.33 ± 0.47	16.40 ± 0.52	15.12 ± 0.43	16.38 ± 0.42	8.82 ± 0.39	9.17 ± 0.48
Australian	15.66 ± 0.37	15.57 ± 0.34	15.54 ± 0.29	15.72 ± 0.40	15.69 ± 0.44	15.41 ± 0.28	15.57 ± 0.36	**14.90 ± 0.72**	**14.85 ± 0.56**
Breast	29.67 ± 0.62	29.63 ± 0.70	29.78 ± 1.04	31.07 ± 0.44	29.33 ± 0.78	30.04 ± 0.51	30.04 ± 0.48	**28.00 ± 0.61**	**28.00 ± 0.74**
Dna	7.15 ± 0.02	7.15 ± 0.02	9.71 ± 0.09	7.15 ± 0.02	7.16 ± 0.02	7.13 ± 0.03	7.15 ± 0.02	**5.28 ± 0.10**	**5.27 ± 0.12**
Glass	33.89 ± 1.05	33.83 ± 1.18	35.11 ± 0.35	37.67 ± 1.83	33.67 ± 0.84	38.22 ± 2.01	33.78 ± 1.01	**31.67 ± 1.01**	**31.67 ± 0.98**
Heart	21.81 ± 0.48	21.89 ± 0.41	21.74 ± 0.50	21.85 ± 0.60	21.74 ± 0.63	20.33 ± 0.66	22.00 ± 0.40	**19.96 ± 0.71**	**19.63 ± 1.05**
HeartC	19.38 ± 0.72	20.48 ± 0.54	21.59 ± 2.16	19.69 ± 0.77	18.90 ± 0.67	19.59 ± 1.85	20.31 ± 0.57	**18.48 ± 0.92**	19.17 ± 0.98
Ionosphere	6.38 ± 0.14	6.35 ± 0.15	6.47 ± 0.20	**6.03 ± 0.21**	6.35 ± 0.21	6.32 ± 0.16	6.35 ± 0.15	**6.03 ± 0.16**	6.06 ± 0.21
Iris	3.80 ± 0.55	**3.33 ± 0.00**	4.67 ± 0.00	4.20 ± 0.32	3.73 ± 0.56	4.07 ± 0.66	3.80 ± 0.55	**3.33 ± 0.00**	**3.33 ± 0.00**
Liver	37.94 ± 0.00	37.94 ± 0.00	37.94 ± 0.00	37.76 ± 0.46	37.97 ± 0.09	36.50 ± 0.61	37.94 ± 0.00	**36.38 ± 0.28**	36.50 ± 0.26
Pima	26.87 ± 0.22	26.53 ± 0.26	26.54 ± 0.26	25.66 ± 0.16	26.80 ± 0.22	25.24 ± 0.28	26.96 ± 0.25	**24.47 ± 0.63**	**24.30 ± 0.47**
Sonar	29.30 ± 0.42	29.20 ± 0.82	29.75 ± 0.82	29.30 ± 1.48	28.90 ± 0.77	27.65 ± 0.82	29.15 ± 0.78	**27.60 ± 1.02**	**27.05 ± 1.26**
Vehicle	29.27 ± 0.31	29.12 ± 0.36	28.06 ± 0.34	29.20 ± 0.51	28.98 ± 0.45	29.09 ± 0.23	29.05 ± 0.53	**28.02 ± 0.81**	28.12 ± 0.65
Vote	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	4.52 ± 0.00	**4.48 ± 0.15**
Wdbc	7.80 ± 0.28	7.77 ± 0.27	7.77 ± 0.19	7.62 ± 0.28	7.84 ± 0.23	7.71 ± 0.18	7.77 ± ± 0.27	**7.12 ± 0.46**	**7.05 ± 0.48**
Wine	7.06 ± 0.51	11.38 ± 1.38	8.38 ± 1.56	13.37 ± 0.60	7.00 ± 0.40	**6.94 ± 0.46**	7.00 ± 0.57	7.94 ± 0.30	7.81 ± 0.44

Note that the result is shown by (A ± B), A and B express the mean of classification error rate (%) and the standard deviation of classification error rate (%) on the test set of each dataset, respectively.

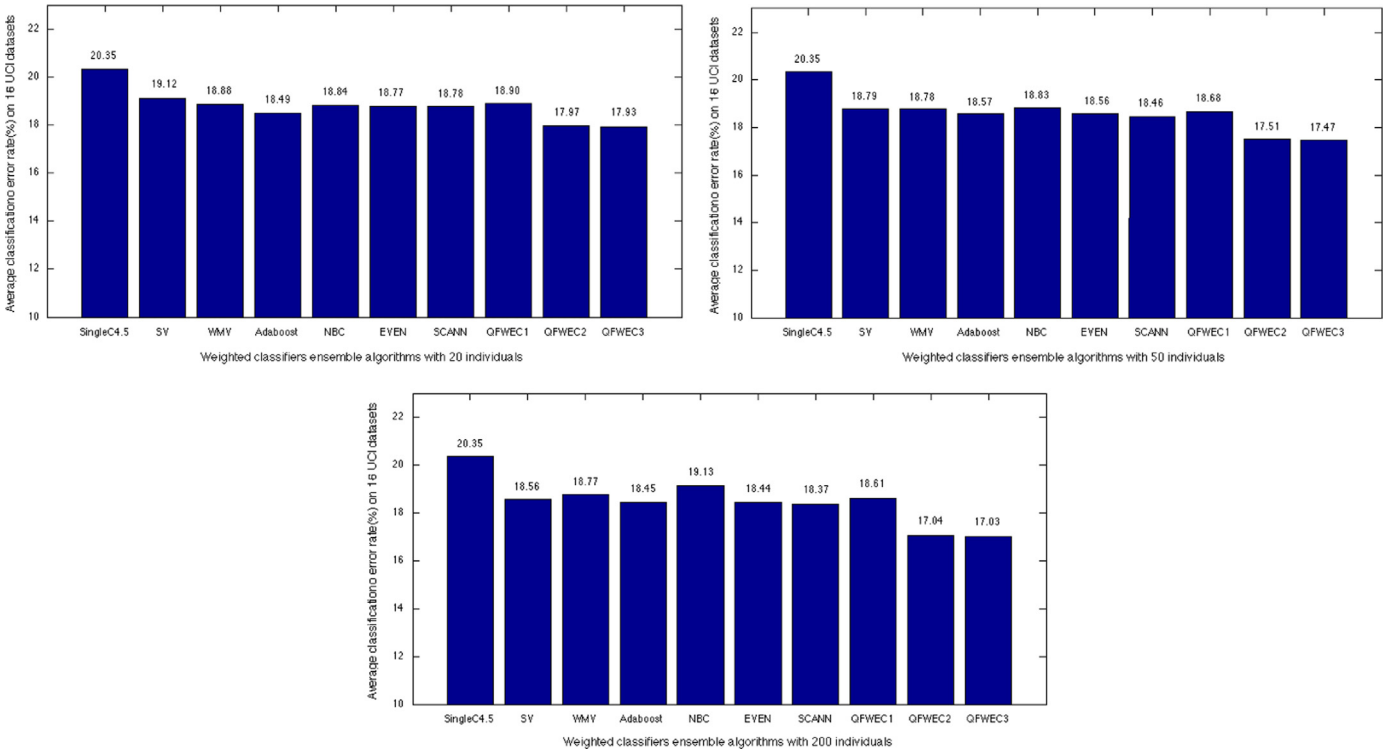

Fig. 5. Average of classification error rates (%) on 16 UCI datasets by ten algorithms.

Table 6
Win/tie/loss counts of nine combination rules with 200 individuals based on *t*-test.

Methods	SV	WMV	AdaBoost	NBC	EVEN	SCANN	QFWEC1	QFWEC2	QFWEC3
SV	–	–	–	–	–	–	–	–	–
WMV	4/10/2	–	–	–	–	–	–	–	–
AdaBoost	3/7/6	3/8/5	–	–	–	–	–	–	–
NBC	4/7/5	2/10/4	5/5/6	–	–	–	–	–	–
EVEN	2/13/1	2/10/4	6/7/3	6/6/4	–	–	–	–	–
SCANN	7/8/1	7/6/3	8/4/4	8/7/1	6/7/3	–	–	–	–
QFWEC1	2/12/2	1/11/4	7/5/4	5/6/5	1/11/4	1/7/8	–	–	–
QFWEC2	14/1/1	14/2/0	12/3/1	14/2/0	14/1/1	9/6/1	14/1/1	–	–
QFWEC3	13/2/1	15/1/0	12/3/1	13/3/0	14/1/1	9/6/1	14/1/1	1/12/3	–

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

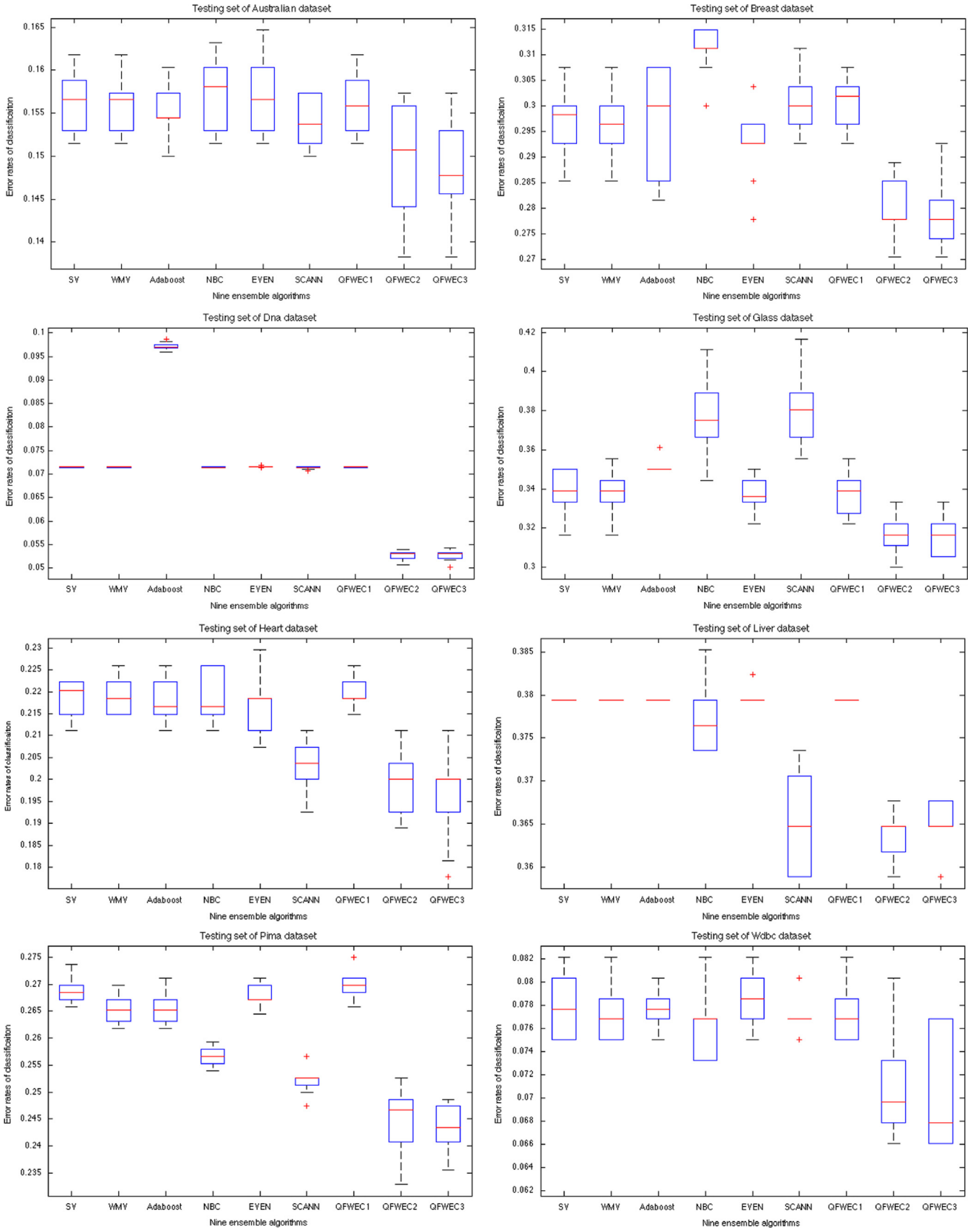

Fig. 6. Error rates of classification obtained by nine ensemble algorithms combining 200 individuals for eight UCI datasets.

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

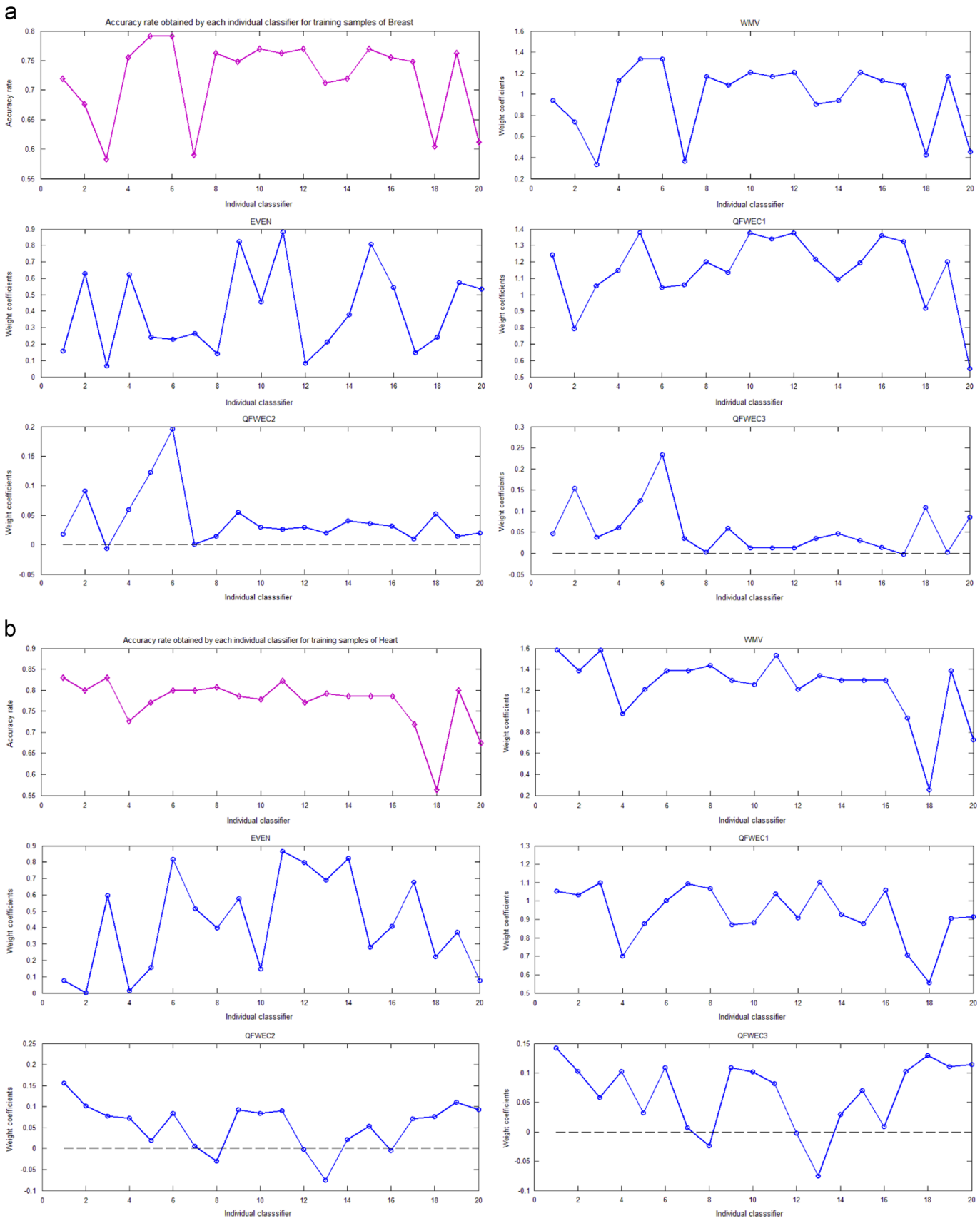

Fig. 7. The curves of weight coefficients of individual classifiers obtained by six algorithms. (a) Breast dataset, (b) Heart dataset, and (c) Pima dataset.

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

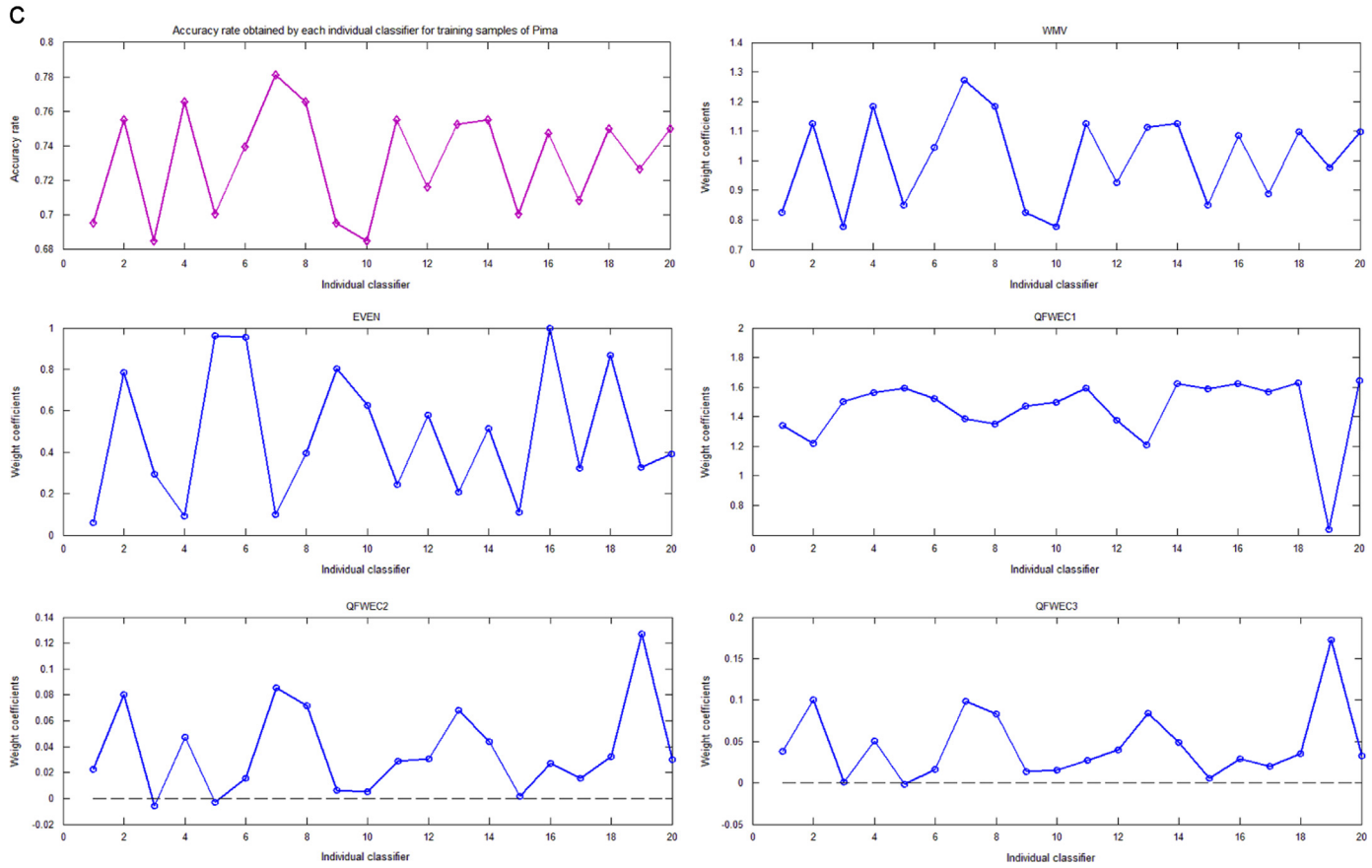

Fig. 7. (*continued*)

are the curves of accuracy rates of the training set are almost the same as the curves of weight coefficients given by WMV algorithm for three datasets. It is the reason that WMV algorithm obtains the weight coefficient of each classifier based on the classification performance of each classifier. Specially, the curve of QFWEC3 is similar to QFWEC2, and only just there are difference in size of coefficients. Thus we focus on the analysis of experimental results of QFWEC2 algorithm in the following parts. In Fig. 7(a), we find that two classifiers (3rd and 7th) are given weight coefficients close to zero value in QFWEC2. In particular, the 3rd and 7th classifiers obtained the worse performance than others according to the curve of accuracy rate. But other methods do not give special values to the 3rd and 7th classifiers. It indicates that QFWEC2 obtains the weight vector which represents better the effects of individual classifiers for the ensemble performance compared against others. On the other hand, it also illustrates that QFWEC2 and QFWEC3 obtain the lowest error according to the shown error rate of each method in Table 7. In Fig. 7(b), we find easily that three classifiers are also given weight coefficients close to zero value by QFWEC2, but it is different from Fig. 7(a). From the curve of accuracy rate of Fig. 7(b), it is found that three classifiers (7th, 12th and 16th) have same accuracy rates as other classifiers (6th, 5th and 15th), respectively, which indicates that the diversity among classifiers is boosted by giving zero-coefficients to some similar classifiers in the proposed methods, and some classifiers are given negative coefficients in QFWEC2 in addition. Similar to Fig. 7(a), it is seen in Fig. 7(c) that some classifiers with the worse performance are given near-zero coefficients in the proposed methods. In summary, the results of the curves of weight vectors illustrate that the proposed methods obtain the weight coefficients which show the effects of classifiers for improving the ensemble performance better and enhance the diversity among classifiers.

Table 7
Error rates (%) of classification by five ensemble methods.

Methods	WMV	EVEN	QFWEC1	QFWEC2	QFWEC3
Breast	29.71	28.26	29.71	26.81	26.81
Heart	27.41	25.93	27.41	22.96	22.96
Pima	28.13	28.39	28.91	26.56	26.56

Moreover, we give the error rates obtained by five methods for three datasets in this experiment, shown in Table 7. From the results of error rates, it is easily found that the proposed methods gain the better performance based on the weight vectors shown in Fig. 7 than other algorithms. Notice that AdaBoost, NBC and SCANN methods are not shown in this experiment. Because individual classifiers produced by AdaBoost method are different from these classifiers shown in Fig. 7, NBC method obtains the weight coefficients of individual classifiers based on the predictive labels of each sample obtained by individual classifiers, which indicates that the weight coefficients gained by NBC method are different for different samples, similarly, and SCANN method obtains the predictive result of an ensemble based on every sample.

4.2.3. *Performance analysis of QFWEC method based on different initial weight vector*

In the proposed method, the function S is equal to the difference between S_0 and a function $\mathbf{F}(\mathbf{w})$ by introducing a given weight vector $\mathbf{w}_0$, detailedly shown in Eqs. (12)–(14), where $\mathbf{w}_0$ is

considered as the initial weight vector of QFWEC method. In particular, this initial weight vector $\mathbf{w}_0$ can be given by artificial initialization with $\mathbf{w}_0 = [1/L, \ldots, 1/L]^T$, random initialization or the weight vector obtained by one kind of ensemble algorithms in our experiments. Hence, in order to verify the effectiveness of QFWEC method for other weight vectors, we implement an experiment that 16 UCI datasets are classified by QFWEC method with three different weight vectors in this section, where three weight vectors are gained by two weighted ensemble methods (WMV and EVEN algorithms) and produced randomly, respectively. The experimental results are shown as follows, and all results are averaged over 10-fold cross validation performed 10 times ensemble, where the number of individual classifiers is 50.

In this experiment, it is noteworthy that the used weight vector $\mathbf{w}_0$ must be satisfied for two constraints ($\sum_{i=1}^{L} w_i = 1$ and $-1 < w_i < 1$) . Therefore, in order to ensure that three weight vectors are bound to satisfy two constraints, we make a simple process for weight vectors: $w_{0i} = w_{0i}^* / \sum_{i=1}^{L} w_{0i}^*$, where w_{0i}^* expresses the weight coefficient of ith individual classifier in three given weight vectors. In fact, this process does not affect the performance of combining individual classifiers with the weight vectors obtained by WMV and EVEN and the randomly produced weight vector, because an ensemble with the weight vector $\mathbf{w}_0$ is equivalent to one with $\mathbf{w}_0^*$.

Table 8 shows the classification errors gained by QFWEC algorithms with the three different initial weight vectors for the testing set, shown by (mean (%) ± standard deviation (%)) of the error rate, where 'RW' expresses the method of combining individual classifiers with a randomly produced weight vector. If the proposed methods outperform the compared method (WMV, EVEN or RW), its results are bolded and vice versa. From the experimental results of Table 8, it is visibly seen that QFWEC2 and QFWEC3 algorithms can obtain lower error than the compared methods when three different weight vectors are respectively employed as QFWEC method's input (the initial weight vector $\mathbf{w}_0$). It illustrates that QFWEC method improve effectively the performance of an ensemble corresponding to the initial weight vector $\mathbf{w}_0$, whatever this weight vector is given randomly or gained by other ensemble methods. In fact, we also find that the proposed methods are superior to SV algorithm corresponding to the weight vector ($\mathbf{w}_0 = [1/L, \ldots, 1/L]^T$) according to the results of Section 4.2.1. In summary, according to results of Tables 3–5 and 8, it demonstrates that QFWEC method can increase the performance of other ensemble algorithms corresponding to its initial weight vector $\mathbf{w}_0$ by maximizing the quadratic form.

Table 8
Error rates (%) of QFWEC algorithms with three different initial weight vectors.

Datasets	QFWEC with the weight vector given by WMV				QFWEC with the weight vector given by EVEN				QFWEC with the weight vector given randomly			
	WMV	QFWEC1	QFWEC2	QFWEC3	EVEN	QFWEC1	QFWEC2	QFWEC3	RW	QFWEC1	QFWEC2	QFWEC3
Air	15.80 ± 0.52	16.53 ± 0.49	**10.75 ± 0.97**	**10.97 ± 0.66**	16.65 ± 0.42	16.53 ± 0.49	**10.73 ± 0.93**	**11.05 ± 0.54**	16.92 ± 0.39	16.53 ± 0.49	**10.78 ± 0.95**	**11.09 ± 0.56**
Australian	15.21 ± 0.56	15.21 ± 0.38	**14.81 ± 0.55**	**14.72 ± 0.59**	15.29 ± 0.41	**15.21 ± 0.38**	**14.81 ± 0.50**	**14.76 ± 0.56**	15.06 ± 0.42	15.21 ± 0.38	**14.82 ± 0.50**	**14.78 ± 0.58**
Breast	29.81 ± 0.68	30.11 ± 0.82	**27.70 ± 0.72**	**27.37 ± 0.95**	30.07 ± 0.89	30.11 ± 0.82	**27.67 ± 0.92**	**27.30 ± 0.84**	29.44 ± 0.86	30.11 ± 0.82	**27.70 ± 0.78**	**27.30 ± 0.91**
Dna	7.09 ± 0.06	7.12 ± 0.06	**5.54 ± 0.15**	**5.55 ± 0.15**	7.09 ± 0.06	7.12 ± 0.06	**5.51 ± 0.14**	**5.51 ± 0.14**	7.11 ± 0.08	7.12 ± 0.06	**5.51 ± 0.15**	**5.51 ± 0.15**
Glass	31.94 ± 0.95	32.67 ± 1.22	**31.83 ± 1.36**	**31.83 ± 1.36**	33.11 ± 1.62	32.67 ± 1.22	**31.44 ± 1.96**	**31.44 ± 1.96**	33.44 ± 1.69	**32.67 ± 1.22**	**31.44 ± 1.98**	**31.44 ± 1.98**
Heart	20.81 ± 0.67	21.89 ± 0.44	**18.89 ± 0.94**	**18.89 ± 1.03**	21.11 ± 0.68	21.89 ± 0.44	**19.00 ± 1.13**	**18.89 ± 1.14**	20.89 ± 0.96	21.89 ± 0.44	**18.96 ± 1.06**	**18.85 ± 1.08**
HeartC	20.03 ± 0.66	20.10 ± 1.05	**19.72 ± 1.19**	**19.79 ± 1.47**	20.14 ± 1.43	20.10 ± 1.05	**19.34 ± 1.08**	**19.72 ± 1.33**	20.28 ± 1.36	**20.10 ± 1.05**	**19.38 ± 1.15**	**19.52 ± 1.25**
Ionosphere	6.94 ± 0.32	7.00 ± 0.30	**6.44 ± 0.32**	**6.44 ± 0.35**	7.00 ± 0.27	7.00 ± 0.30	**6.47 ± 0.37**	**6.44 ± 0.35**	6.85 ± 0.31	7.00 ± 0.30	**6.47 ± 0.37**	**6.44 ± 0.35**
Iris	**4.00 ± 0.70**	4.47 ± 0.45	**4.00 ± 0.70**	**4.00 ± 0.70**	4.33 ± 0.65	4.47 ± 0.45	**3.40 ± 0.21**	**3.47 ± 0.42**	4.47 ± 0.71	4.47 ± 0.45	**3.40 ± 0.21**	**3.47 ± 0.42**
Liver	37.97 ± 0.22	38.12 ± 0.28	**37.26 ± 0.34**	**37.15 ± 0.48**	37.82 ± 0.28	38.12 ± 0.28	**37.18 ± 0.48**	**37.18 ± 0.48**	38.00 ± 0.39	38.12 ± 0.28	**37.21 ± 0.42**	**37.18 ± 0.54**
Pima	26.33 ± 0.46	26.72 ± 0.34	**24.58 ± 0.41**	**24.54 ± 0.47**	26.57 ± 0.49	26.72 ± 0.34	**24.55 ± 0.40**	**24.61 ± 0.46**	26.59 ± 0.44	26.72 ± 0.34	**24.63 ± 0.42**	**24.63 ± 0.34**
Sonar	28.25 ± 1.42	28.40 ± 1.24	28.70 ± 2.15	**27.90 ± 1.88**	**27.50 ± 1.83**	28.40 ± 1.24	28.85 ± 2.21	28.00 ± 1.72	28.20 ± 1.55	28.40 ± 1.24	28.85 ± 2.35	**28.10 ± 1.87**
Vehicle	29.15 ± 0.48	29.38 ± 0.43	**27.93 ± 0.87**	**27.93 ± 0.93**	28.94 ± 0.46	29.38 ± 0.43	**27.90 ± 0.67**	**27.87 ± 0.72**	28.91 ± 0.78	29.38 ± 0.43	**27.91 ± 0.75**	**27.91 ± 0.80**
Vote	4.52 ± 0.00	4.52 ± 0.00	**4.50 ± 0.18**	**4.43 ± 0.17**	4.52 ± 0.00	4.52 ± 0.00	**4.48 ± 0.19**	**4.45 ± 0.16**	4.52 ± 0.00	4.52 ± 0.00	**4.48 ± 0.19**	**4.48 ± 0.22**
Wdbc	6.89 ± 0.38	**6.82 ± 0.42**	**6.59 ± 0.36**	**6.73 ± 0.40**	6.91 ± 0.35	6.82 ± 0.42	**6.61 ± 0.37**	**6.73 ± 0.40**	7.04 ± 0.45	**6.82 ± 0.42**	**6.59 ± 0.36**	**6.71 ± 0.39**
Wine	8.63 ± 1.01	**7.75 ± 1.45**	**7.44 ± 2.01**	**7.69 ± 1.91**	8.44 ± 1.19	**7.75 ± 1.45**	**6.25 ± 1.14**	**6.31 ± 1.00**	7.94 ± 1.02	**7.75 ± 1.45**	**6.31 ± 1.04**	**6.44 ± 1.14**

4.2.4. Analysis of parameters σ and λ

In QFWEC algorithms, there are two important parameters (σ and λ). Thus, an experiment about using different parameters is implemented in this section. In the experiment, two datasets (Breast and Heart) are classified by QFWEC2 and QFWEC3 with different parameters. The parameter σ is given {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9} and τ is {0.1, 0.2, 0.3, 0.4, 0.5, 0.6}, $\lambda = \tau N$, where N is the number of training samples of a dataset. All results of various parameters are shown in Fig. 8. In Fig. 8, two left figures express the trend of accuracy rates of classification obtained by QFWEC2 over different values of σ, and the right ones express the results obtained by QFWEC3 with different values of σ and τ. Specially, each line is the trend based on a value of τ in right figures, and each value of τ is marked near to each line.

From the results of Fig. 8, it is seen that there is a slight fluctuation of performance based on the values of σ (0.3–0.9) for breast dataset and (0.1–0.9) for heart dataset for QFWEC2, and even the discrepancy is less than 0.004. But the curve of breast dataset indicates that its accuracy rate is increased when the value

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

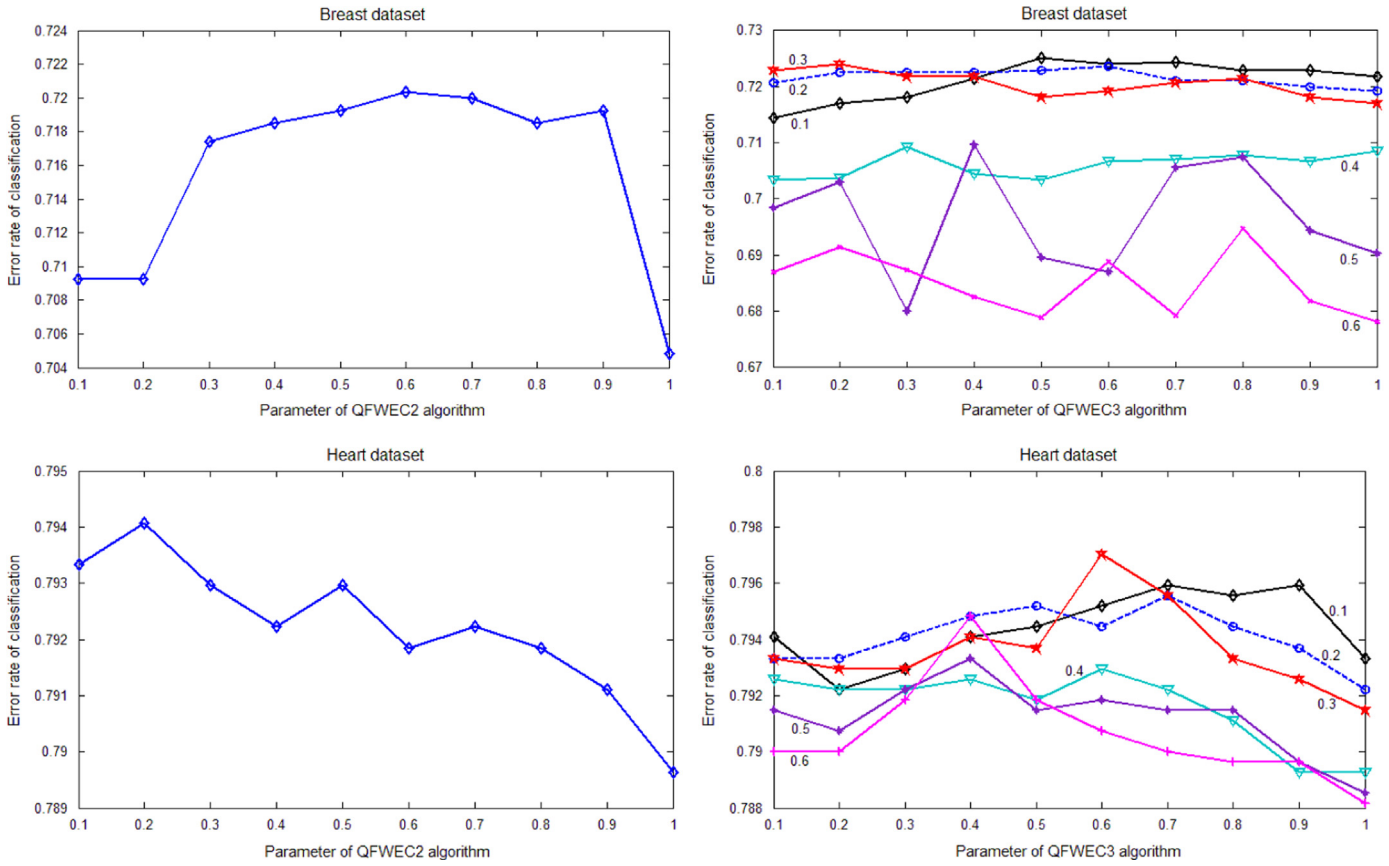

Fig. 8. Diagrams of accuracy rates obtained by QFWEC2 and QFWEC3 with different parameters.

Fig. 9. NASA/JPL PolSAR image of San Francisco and entropy H [33], respectively, shown in Table 9. Thus a pixel with ten features is considered as a sample of the classification of PolSAR image in our experiment. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

of σ becomes larger, on the contrary, the curve of heart dataset indicates that its accuracy rate is decreased when the value of σ becomes larger. The results of heart dataset illustrates obviously that when the value of σ exceeds the upper bound of σ in the analysis of Appendix A, Eq. (28) has not a feasible solution, and thus its accuracy rate is decreased when σ becomes large. Note that QFWEC2 algorithm degenerates to QFWEC1 algorithm when $\sigma = 1$. For QFWEC3 algorithm, the curves of breast and heart datasets indicate that the performance corresponding to $\tau = 0.1$, 0.2 or 0.3 is more stable and better than $\tau = 0.4$, 0.5 and 0.6, and the performance is reduced when the value of τ becomes larger. In summary, there is a little difference in the error rates of QFWEC2 method while changing the value of the parameter σ. Nevertheless, as increasing the values of the parameter σ and λ, the difference in the error rates of QFWEC3 is larger than that of QFWEC2. Furthermore, QFWEC3 may obtain the more stable and higher accuracy rate when the values of τ are 0.1, 0.2, and 0.3.

4.3. Experiments on PolSAR data

Synthetic aperture radar (SAR) imaging [32] is a well-developed coherent and microwave remote sensing technique for providing large-scaled two-dimensional (2D) high spatial resolution images of the Earth's surface reflectivity. The first fully polarimetric radar system was the L-band AIRSAR [32] which was started in 1980s by the Jet Propulsion Laboratory. Polarimetric SAR (PolSAR) [32,33,34] implements the polarization measurement and obtains more abundant information of targets. Recently, PolSAR image classification [32,35,36] has been an important research for SAR image processing. Hence, the NASA/JPL AIRSAR L-band data of San Francisco[2] is experimented as a real data in order to illustrate the performance of QFWEC algorithms, and its original image is shown in Fig. 9. The original data is four-look data with 900×1024, and three classes are considered in our experiments, including ocean, forest and city. Ocean areas are shown by blue, forest areas are shown by green and the rests are city areas.

[2] http://earth.eo.esa.int/polsarpro/default.html.

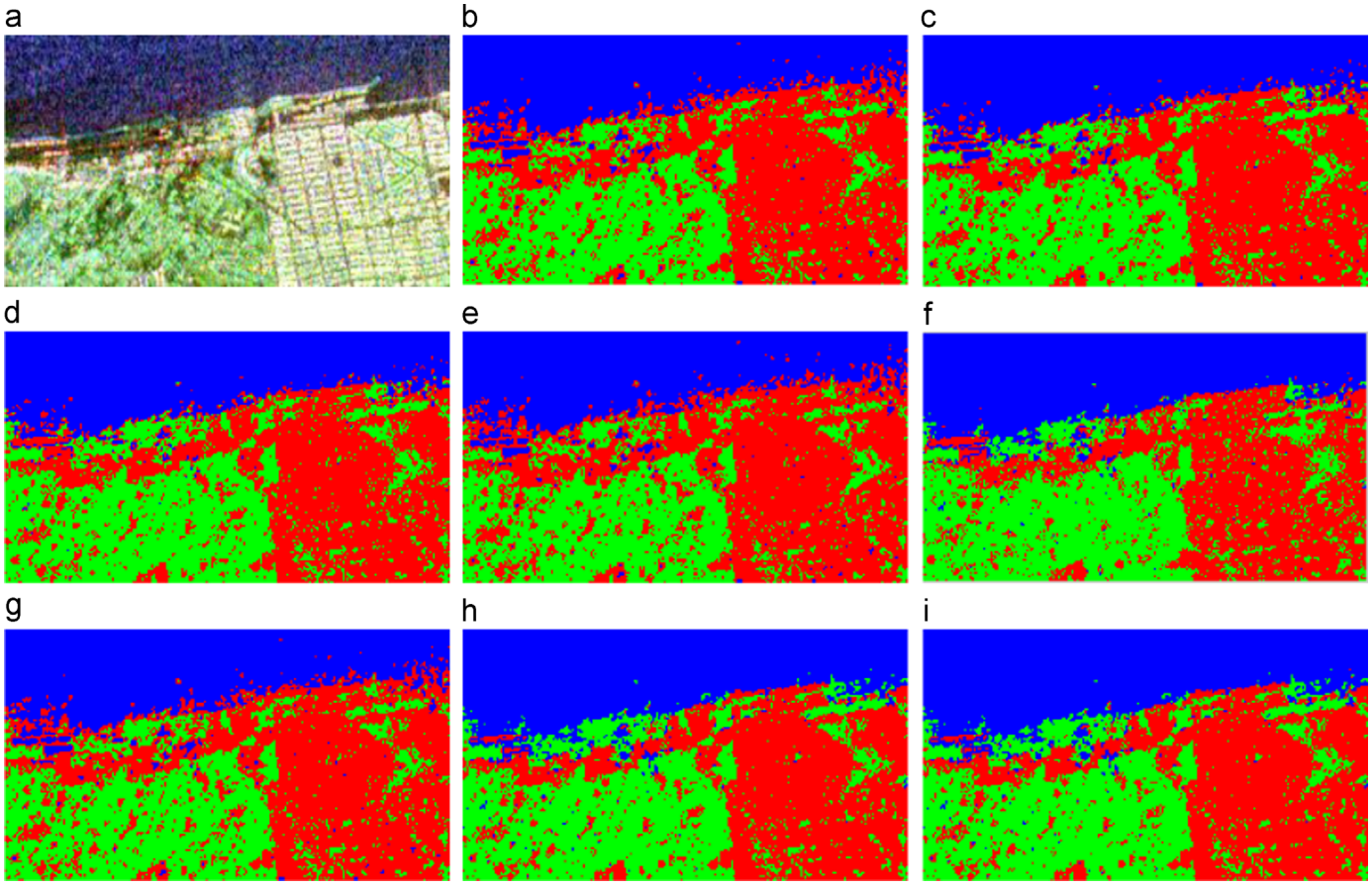

Fig. 10. Classification results obtained by seven ensemble algorithms for Zone 1 (134 × 237). (a) Original image, (b) SV, (c) WMV, (d) NBC, (e) EVEN, (f) SCANN, (g) QFWEC1, (h) QFWEC2, and (i) QFWEC3. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

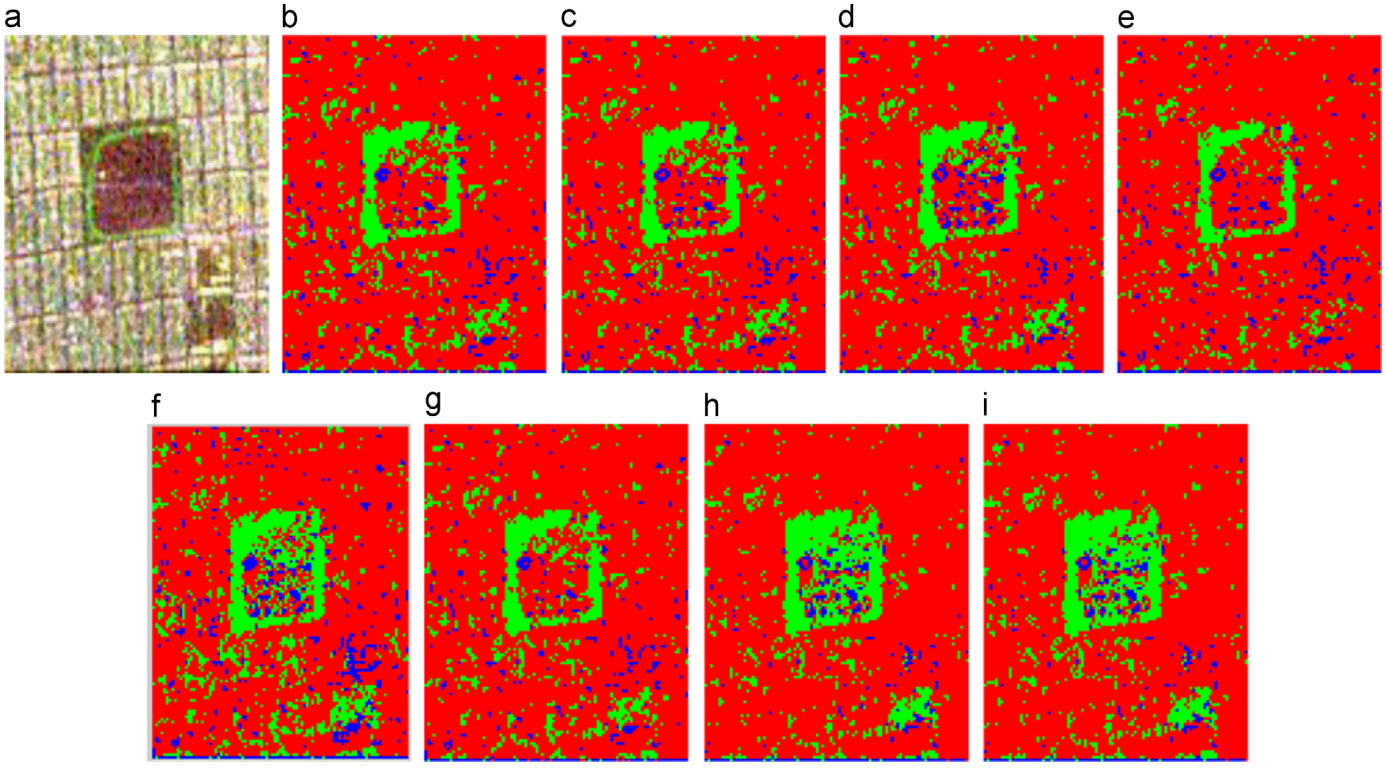

Fig. 11. Classification results obtained by seven ensemble algorithms for Zone 2 (122 × 95). (a) Original image, (b) SV, (c) WMV, (d) NBC, (e) EVEN, (f) SCANN, (g) QFWEC1, (h) QFWEC2, and (i) QFWEC3. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

In this experiment, we choose 3000 pixels from original image data randomly, and each class has 1000 pixels. For PolSAR data, a pixel is represented by a coherency matrix T [32,34]. Then two areas (Zone 1 and Zone 2) are selected from NASA/JPL PolSAR image as two test sets in the experiment, labeled in a black box in Fig. 9. The sizes of Zone 1 and Zone 2 are 134×237 and 122×95, respectively, and their original images are shown in Fig. 10(a) and Fig. 11(a). In order to cover more information of PolSAR data, we used ten features for each pixel. Ten features are three diagonal elements of the coherency matrix T, the real parts of the three complex upper triangular elements of the coherency matrix T, three eigenvalues of the coherency matrix T and entropy H [33], respectively, shown in Table 9. Thus a pixel with ten features is considered as a sample of the classification of PolSAR image in our experiment.

In experiments, SVM classifier is employed as a basic classifier model. For SVM, the RBF kernel function is applied with the kernel parameter ($p = 0.5$) and the penalty factor applies to $C = \mathrm{Inf}$. In order to reduce the training computational complexity of individual classifiers, the training subset of each individual classifier contains 300 samples which are selected randomly from original training samples, and the number of samples that belong to each class is 100, respectively. In addition, the number of individual classifiers is 15. In particular, AdaBoost algorithm is not experimented in this section, because all training samples (3000 samples) are learned by a SVM classifier at first iteration when AdaBoost algorithm is executed, and it means that AdaBoost algorithm has very high computational complexity compared against other methods. As follows, the results of classification for Zones 1 and 2 are shown in Figs. 10 and 11.

In Fig. 10 and Fig. 11, (b)–(i) denote the classification results of SV, WMV, NBC, EVEN, SCANN, QFWEC1, QFWEC2 and QFWEC3, respectively. In figures, the samples belonged to the ocean area are shown in blue, the samples belonged to the forest area are shown in green and the ones belonged to the city area are shown in red. From Fig. 10, it is seen that QFWEC2 and QFWEC3 algorithms obtain better classification results than other algorithms. Especially, less samples located on the boundary of ocean and forest (or ocean and city) are misclassified to city by the proposed methods than others. In Fig. 11, it shows the classification results concentrating on two classes (forest and city). According to the results, it is obvious that QFWEC2 and QFWEC3 algorithms gain larger areas of forest, but other methods gain only a boundary of forest and misclassify a lot of samples belonged to forest areas into city areas. In addition, it is found that some samples are misclassified into ocean according to results of Fig. 11, because we still employ all training samples with three classes to classify for Zone 2.

5. Conclusions

In this paper, in order to avoid the dilemma problem generated by constructing an ensemble algorithm based on diversity and accuracy, a general form of the ensemble error is utilized to seek an optimal weight vector of classifiers. Nevertheless, it is difficult to perform the minimization of the general form of the ensemble error, directly. Hence, the general form of the ensemble error is replaced by an approximation form of the ensemble error with two constraints. Moreover, by introducing a weight vector, the approximation form is decomposed into two parts: a quadratic form and an error term based on the introduced weight vector. Consequently, we propose a new method to combine classifiers with an optimal weight vector acquired by maximizing three different quadratic forms. Particularly, a quadratic form is given from the approximation form based on the introduced weight vector. According to the theoretical analysis, when the value of the quadratic form is larger than zero, the ensemble error gained by the weight vector corresponding to it is lower than the one given by the introduced weight vector. Furthermore, the experimental results demonstrate that QFWEC algorithms improve the ensemble performance compared against other combination methods. It is desired that QFWEC3 is superior to QFWEC2, whereas we find that QFWEC3 almost ties QFWEC2 from our experimental results. Therefore we will apply other methods to solve Eq. (23) in our further work. The experimental results also illustrate our method increases the classification performance based on different initial weight vectors, thus we will expand this method into selective ensemble or sparse ensemble. In addition, we will consider how to seek the optimal weight vector directly from multi-class problems and extend the proposed method in regression.

Table 9
Ten features applied in the experiment.

Diagonal elements	Upper triangular elements	Eigenvalues	Entropy
$T_{11}T_{22}T_{33}$	$T_{12}T_{13}T_{23}$	$\lambda_1\lambda_2\lambda_3$	H

Acknowledgments

The authors would like to thank Editor, Associate Editor and five anonymous reviewers for their valuable comments and suggestions. This work is supported in part by the National Basic Research Program (973 Program) of China (No. 2013CB329402), the National Natural Science Foundation of China No. 61003198 and No. 61272282. This work is also supported by SUTD Start-Up Grant ISTD 2011 016, SUTD-MIT International Design Center Grant IDG31300106, Singapore MOE Academic Research Fund MOE2013-T2-1-159 and SUTD-ZJU Collaboration Research Grant 2012 SUTD-ZJU/RES/03/2012.

Appendix A. The illustration of two constraint constraint

Based on two constraints $\sum_{i=1}^{L} w_i = 1$ and $\sigma^2 + \sum_{i=1}^{L} w_i^2 = 1$, we give the illustration about the range of the parameter σ, on which there is a solution at least to satisfy two constraint meanwhile. Firstly, we give an analysis for $L = 2$, and then two constraints are shown when $L = 2$:

$$\begin{cases} w_1 + w_2 = 1 \\ w_1^2 + w_2^2 = 1 - \sigma^2 \end{cases} \tag{27}$$

In fact, it is easy to gain the range of σ by Eq. (27). Based on Eq. (27), the range is equal to $0 \le \sigma \le \sqrt{2}/2 \approx 0.7071$, and it means that Eq. (27) has the feasible solution when $0 \le \sigma \le 0.7071$. As follows, we give the pictures corresponding to the maximum of σ, the minimum of σ and no solution of Eq.(27) in Fig. 12. In Fig. 12, the red curve denotes a quarter of the circle about $w_1^2 + w_2^2 = 1 - \sigma^2$, the blue line denotes the line segment about $w_1 + w_2 = 1$, and the black dotted line expresses the radius (r) of the circle. From Fig. 12, it is obvious that Eq.(27) has two feasible solutions when $0 \le \sigma \le 0.7071$ (shown in Fig. 12(a)), Eq.(27) has only one solution when $\sigma = 0.7071$ (shown in Fig. 12(b)), and there is no solution for Eq.(27) when $0.7071 < \sigma < 1$ (shown in Fig. 12(c)).

For $L > 2$, the problem is a high dimensional problems about the weight vector $\mathbf{w}$, shown as follows:

$$\begin{cases} \sum_{i=1}^{L} w_i = 1 \\ \sum_{i=1}^{L} w_i^2 = 1 - \sigma^2 \end{cases} \tag{28}$$

For high dimensional problems, the 3-dimensional picture can be easily shown. Thus we make an analysis about the condition $L = 3$. For $L = 3$, we gain a spherical surface S and a plane P based

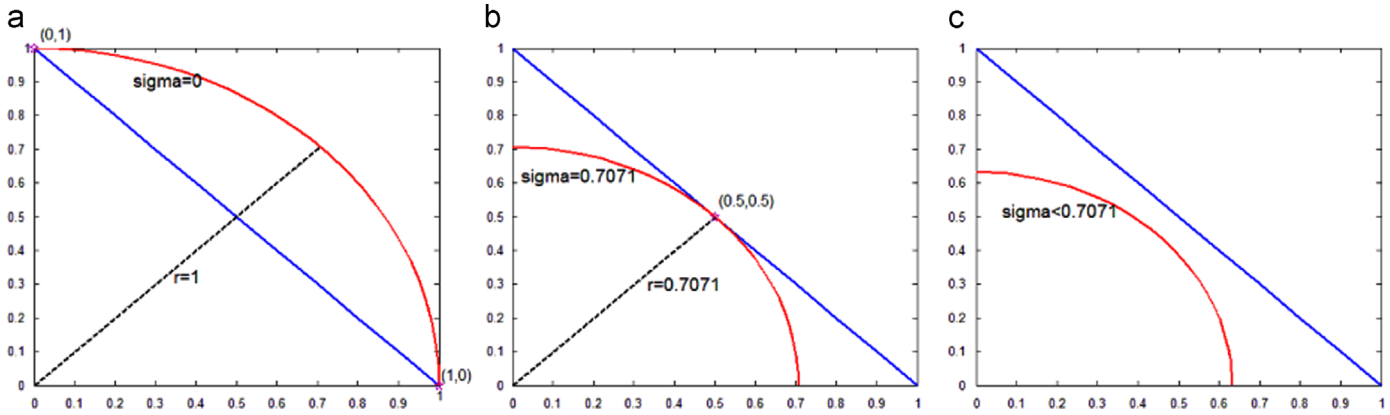

Fig. 12. Diagrams of the maximum and minimum of σ and no solution for $L=2$.

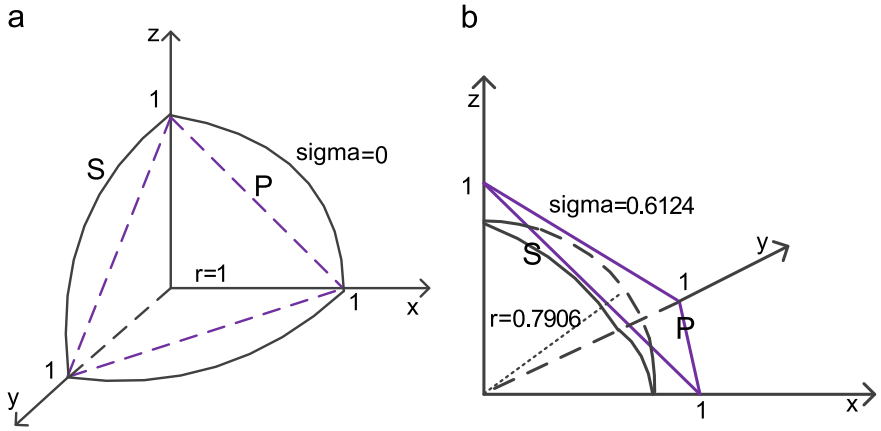

Fig. 13. Diagrams of the maximum and the minimum of σ for $L=3$.

on Eq. (28). The spherical surface S is depicted as $\sum_{i=1}^{3} w_i^2 = 1-\sigma^2$ and the plane P is indicated as $\sum_{i=1}^{3} w_i = 1$. Then two special cases are shown in Fig. 13, respectively corresponding to the minimum and the maximum of σ. In Fig. 13, the area surrounded by three purple lines expresses the plane P, the area surrounded by three black curves expresses the spherical surface S, r denotes the radius of the sphere, and 'sigma' expresses the parameter σ. When $\sigma=0$, a spherical surface S with the largest radius is given and it intersects P at three points, shown in Fig. 13(a). It indicates that Eq. (28) has three feasible solutions for $L=3$ when $\sigma=0$. When $\sigma=\sqrt{6/4}\approx 0.6124$, a spherical surface S with the smallest radius is given and it is tangent to the plane P, shown in Fig. 13(b). It illustrates that Eq. (28) has only one feasible solution when $\sigma=0.6124$ for $L=3$, and it means that Eq. (28) has no solution when $\sigma > 0.6124$ for $L=3$. Hence, Eq. (28) has the feasible solution when $0\le\sigma\le\sqrt{6/4}$ for $L=3$.

In summary, according to the analysis for $L=2$ and 3, it is found that the upper bound of σ for $L=2$ is higher than $L=3$, which indicates that the upper bound of σ may decrease when the number of individual classifiers increases. Additionally, it is easily seen based on Fig. 12(a) and Fig. 13(a) that the number of feasible solutions of Eq. (28) is relation to the number of individual classifiers and it is equal to L.

References

[1] Thomas G. Dietterich, Ensemble learning, in: M.A. Arbib (Ed.), In The Handbook of Brain Theory and Networks, Second edition,The MIT Press, Cambridge, MA, 2002, pp. 405–408.
[2] L. Breiman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.
[3] A. Patra, S. Das, Enhancing decision combination of face and fingerprint by exploitation of individual classifier space: an approach to multi-modal biometry, Pattern Recognit. 41 (2008) 2298–2308.
[4] Santanu Ghorai, Anirban Mukherjee, Sanghamitra Sengupta, Pranab K Dutta, Cancer classification from gene expression data by NPPC ensemble, IEEE/ACM Trans. Comput. Biol. Bioinform. 8 (2011) 659–671.
[5] Li Liu, Ling Shao, Peter Rockett, Boosted key-frame selection and correlated pyramidal motion-feature representation for human action recognition, Pattern Recognit. 46 (2013) 1810–1818.
[6] Lars Kai Hansen, Peter Salamon, Neural network ensembles, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990) 993–1001.
[7] L. Breiman, Random forest, Mach. Learn. 45 (2001) 5–32.
[8] L.I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Published by John Wiley & Sons, Inc., Hoboken, NJ, 2004.
[9] J.J. Rodriguez, L.I. Kuncheva, C.J. Alonso, Rotation forest: a new classifier ensemble method, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 1619–1630.
[10] N. Garcia-Pedrajas, Constructing ensembles of classifiers by means of weighted instance selection, IEEE Trans. Neural Networks 20 (2009) 258–277.
[11] L. Zhang, W.D. Zhou, Sparse ensemble using weighted combination methods based on linear programming, Pattern Recognit. 44 (2011) 97–106.
[12] Hanning Yuan, Meng Fang, Xingquan Zhu, Hierarchical sampling for multi-instance ensemble learning, IEEE Trans. Knowl. Data Eng. December (2012), doi: 10.1109/TKDE.2012.245.
[13] Daniel Hernandez-Lobato, Gonzalo Martinez-Munoz, Alberto Suarez, How large should ensembles of classifiers be? Pattern Recognit. 46 (2013) 1323–1336.
[14] D.D. Margineantu, T.G. Dietterich, Pruning adaptive boosting, in: Proceedings of the 14th International Conference on Machine Learning, 1997, pp. 211–218.
[15] Ludmila I. Kuncheva, Juan J. Rodriguez, A weighted voting framework for classifiers ensemble, Knowl. Inf. Syst. 38 (2) (2014) 259–275.
[16] Naonori Ueda, Optimal linear combination of neural networks for improving classification performance, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 207–215.
[17] N.V. Chawla, J. Sylvester, Exploiting diversity in ensembles: improving the performance on unbalanced datasets, Mult. Classif. Syst. Lect. Notes Comput. Sci. 4472 (2007) 397–406.
[18] Y. Zhang, S. Burer, W.N. Street, Ensemble pruning via semi-definite programming, J. Mach. Learn. Res. 7 (2006) 1315–1338.

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017

[19] S.S. Mao, L.C. Jiao, L. Xiong, S.P. Gou, Greedy optimization classifiers ensemble based on diversity, Pattern Recognit. 44 (2011) 1245–1261.
[20] N. Li, Y. Yu, Z.H. Zhou, Diversity regularized ensemble pruning, Mach. Learn. Knowl. Discov. Databases Lect. Notes Comput. Sci. 7523 (2012) 330–345.
[21] Zhihua Zhou, Jianxin Wu, Wei Tang, Ensembling neural networks: many could be better than all, Artif. Intell. 137 (2002) 239–263.
[22] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation, and active learning, in: G. Tesauro, D. Touretzky, T. Leen (Eds.), Advances in Neural Information Processing Systems, 7, The MIT Press, 1995, pp. 231–238.
[23] Z.H. Zhou, J.X. Wu, Y. Jiang, S.F. Chen, Genetic algorithm based selective neural network ensemble, in: 17th International Joint Conference on Artificial Intelligence, 2001, pp. 797–802.
[24] L.I. Kuncheva, A bound on kappa-error diagrams for analysis of classifier ensembles, IEEE Trans. Knowl. Data Eng. 25 (2013) 494–501.
[25] D.M. Titterington, G.D. Murray, L.S. Murray, D.J. Spiegelhalter, A.M. Skene, J.D. F. Habbema, G.J. Gelpke, Comparison of discrimination techniques applied to a complex data set of head injured patients, J. R. Stat. Soc. Ser. A (Gen.) 144 (1981) 145–175.
[26] P. Domingos, M. Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss, Mach. Learn. 29 (1997) 103–130.
[27] C. Blake, C.J. Merz, UCI Repository of Machine Learning Databases, 1998, ⟨http://www.ics.uci.Edu/mlearn/MLRepository.html⟩.
[28] Tin Kam Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 832–844.
[29] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, 1993.
[30] Christopher J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discov. 2 (1998) 121–167.
[31] J. Weston, C. Watkins, Multi-class Support Vector Machines, Technical Report CSD-TR-98-04, Royal Holloway University of London, 1998.
[32] Jong-Sen Lee, Eric Pottier, Polarimetric Radar Imaging: From Basics to Applications, CRC Press, Boca Raton, FL, 2009.
[33] S.R. Cloude, E Pottier., A review of target decomposition theorems in radar polarimetry, IEEE Trans. Geosci. Remote Sens. 34 (2) (1996) 498–518.
[34] Fang Cao, Wen Hong, Yirong Wu, Eric Pottier, An unsupervised segmentation with an adaptive number of clusters using the *SPAN*/*H*/α/*A* space and the complex wishart clustering of fully polarimetric SAR data analysis, IEEE Trans. Geosci. Remote Sens. 45 (11) (2007) 3454–3467.
[35] Shuang Zhang, Shuang Wang, Bo Chen, Shasha Mao, Classification method for fully PolSAR data based on three novel parameters, IEEE Geosci. Remote Sens. Lett. 11 (1) (2014) 39–43.
[36] Bo Chen, Shuang Wang, Licheng Jiao, Shuang Zhang, Unsupervised polarimetric SAR image classification using fisher linear discriminant, in: Proceedings of the 2nd AsianPacific Conference on Synthetic Aperture Radar, 2009, pp. 738–741.
[37] Giorgio Fumera, Fabio Roli, A theoretical and experimental analysis of linear combiners for multiple classifier systems, IEEE Trans. Pattern Anal. Mach. Intell. 27 (6) (2005) 942–956.
[38] Yun Li, Suyan Gao, Songcan Chen, Ensemble feature weighting based on local learning and diversity, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2012, pp. 1019–1025.
[39] C.J. Merz, Using correspondence analysis to combine classifiers, Mach. Learn. 36 (1999) 33–58.
[40] L. Rokach, Ensemble-based classifiers, Artif. Intell. Rev. 33 (2010) 1–39.
[41] S. Džeroski, B. Ženko., Is combining classifiers with stacking better than selecting the best one? Mach. Learn. 54 (3) (2004) 255–273.
[42] S.Y. Sohn, H. Choi. Ensemble based on data envelopment analysis, in: ECML Meta Learning Workshop, 2001.
[43] P. Derbeko, R. El-Yaniv, R. Meir, Variance optimized bagging. Machine learning: ECML, Lect. Notes Comput. Sci. 2430 (2002) 60–72.
[44] ShaSha Mao, Lin Xiong, Licheng Jiao, Shuang Zhang, Bo Chen, Weighted ensemble based on 0–1 matrix decomposition, Electron. Lett. 49 (2) (2013) 116–118.
[45] Xingquan Zhu, Xindong Wu, Ying Yang. Dynamic classifier selection for effective mining from noisy data streams, in: Proceeding of the Fourth IEEE International Conference on Data Mining (ICDM'04), 2004, pp. 305–312.
[46] A. Tsymbal, M. Pechenizkiy, S. Puuronen, D.W. Patterson, Dynamic integration of classifiers in the space of principal components, Adv. Databases Inf. Syst. Lect. Notes Comput. Sci. 2798 (2003) 278–292.
[47] Qun Dai, A novel ensemble pruning algorithm based on randomized greedy selective strategy and ballot, Neurocomputing 122 (2013) 258–265.
[48] Huanhuan Chen, Peter Tino, Xin Yao, Predictive ensemble pruning by expectation propagation, IEEE Trans. Knowl. Data Eng. 21 (7) (2009) 999–1013.

Shasha Mao received the B.S. degree in measuring and controlling technology and instrument and the Ph.D. degree in circuits and systems from Xidian University, Xi'an, China, in 2006 and 2014, respectively. She is currently working as a postdoctoral research fellow of VGD group at Singapore University of Technology and Design, Singapore. Her current research interests include ensemble learning, data mining, city understanding, and computer vision.

Licheng Jiao (M'87–SM'91) received the B.S. degree from Shanghai Jiaotong University, China, in 1982 and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively. From 1984 to 1986, he was an Assistant Professor with the Civil Aviation Institute of China, Tianjing, China. During 1990 and 1991, he was a Postdoctoral Fellowwith the Key Lab for Radar Signal Processing, Xidian University, Xi'an, China. He is currently Professor and Dean of the School of Electronic Engineering in Xidian University. His current research interests include signal and image processing, nonlinear circuits and systems theory, learning theory and algorithms, optimization problems, wavelet theory, machine learning and data mining. He is the author of four books: Theory of Neural Network Systems, Theory and Application on Nonlinear Transformation Functions, Neural Computing, and Applications and Implementations of Neural Networks (1990, 1992, 1993, and 1996, respectively, Xidian University Press: Xi'an). He is the author or coauthor of more than 150 scientific papers.

Lin Xiong received the B.S. degree in material forming and control engineering from Shaanxi University of Science & Technology, Xi'an, China, in 2003. He worked for Foxconn Co. from 2003 to 2005. Since 2006, he has been working toward the M.S. degree in pattern recognition and intelligent system at Xidian University, Xi'an, China. He is currently pursing the Ph.D. degree with the key lab of intelligent perception and image understanding of ministry of education, School of Electronic Engineering, Xidian University, Xi'an, China. His current research interests include Riemannian manifold optimization, low-rank and sparse matrix factorization, background modeling, ensemble learning and active learning.

Shuiping Gou received the B.S. and M.S. degrees in Computer Science and Technology from Xidian University, Xi'an, China, in 2000 and 2003 respectively, and the Ph.D. degree in Pattern Recognition and Intelligent System from Xidian University, Xi'an, China, in 2008. She is currently an associate professor with Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China at Xidian University. Her current research interests include machine learning, evolutionary computation, image understand and interpretation, and data mining.

Bo Chen was born in Henan, China, on March 14, 1985. She received the B.S. degree from Xidian University, Xi'an, China, in 2007. Since 2007, she has been working toward the M.S. and Ph.D. degrees in circuit and system at Xidian University. Her research interests include POLSAR image processing and classification and multiple classifier system.

Sai-Kit Yeung received the B.E. degree in computer engineering, the M.S. degree in bioengineering, the Ph.D degree in electronic and computer engineering from HKUST, in 2003, 2005, and 2009, respectively. He is currently an Assistant Professor with the Pillar of Information Systems Technology and Design, Singapore University of Technology and Design, Singapore. He has authored or co-authored more than 20 papers in journals and conferences at top venues including the IEEE T-PAMI, TOG, SIGGRAPH, CVPR and ECCV. His current research interests include computer vision, computer graphics and image processing.

Please cite this article as: S. Mao, et al., Weighted classifier ensemble based on quadratic form, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.10.017