

Received July 26, 2017, accepted August 10, 2017, date of publication August 23, 2017, date of current version September 6, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2740420

Hybrid Structure-Adaptive RBF-ELM Network Classifier

HUI WEN^{1,2}, HONGGUANG FAN¹, WEIXIN XIE¹, AND JIHONG PEI¹, (Member, IEEE)

¹ ATR Key Laboratory, Shenzhen University, Shenzhen 518060, China

² Hubei Normal University, Huangshi 435002, China

Corresponding author: Jihong Pei (jhpei@szu.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61331021 and in part by the Shenzhen Science and Technology Plan Project Under Grant JCYJ20130408173025036.

ABSTRACT

In this paper, a hybrid structure-adaptive radial basis function-extreme learning machine (HSARBF-ELM) network classifier is presented. HSARBF-ELM consists of a structure-adaptive radial basis function (SARBF) network and an extreme learning machine (ELM) network of cascade, where the output of the SARBF network hidden layer is used as the input layer of the ELM network. In the HSARBF-ELM network classifier, the SARBF network is utilized to achieve adaptively localizing kernel mapping of input vectors, after that step, the ELM network is utilized to implement global classification of mapping samples in the kernel space. HSARBF-ELM indicates the combination of localized kernel mapping learning and the global nonlinear classification, which combines the advantages of the SARBF network and the ELM network. The quantitative conditions for the separability enhancement and the corresponding theoretical explanation for the HSARBF-ELM network are given, which demonstrate that when input vectors go through the SARBF network, adaptively adjusting the RBF kernel parameters can boost the separability of the original sample space. Thus, the classification performance of the HSARBF-ELM network can be guaranteed theoretically. An appropriate learning algorithm for the HSARBF-ELM network is subsequently presented, which effectively combines the methods of density clustering with a potential function, center-oriented unidirectional repulsive force and the existing ELM algorithm, and the optimized complementary HSARBF-ELM network can be constructed. The experimental results show that the classification performance of the HSARBF-ELM network clearly outperforms the ELM network, and outperforms other classifiers on most classification problems.

INDEX TERMS Extreme learning machine (ELM), radial basis function (RBF), density clustering, hybrid, structure-adaptive, separability analysis, neural network.

I. INTRODUCTION

In recent years, with the rapid development of intelligent information processing technology, various optimization algorithms have been proposed to solve different problems, such as k-nearest neighbors [1]–[4], fuzzy clustering [5], [6], support vector machines (SVM) [7]–[10] and neural networks based on adaptive learning [11]–[15]. Extreme learning machine (ELM) [16] has been researched intensively in the past several years. Compared to traditional gradient descent algorithms, ELM has better classification and generalization performance, which can overcome such problems as a slow convergence rate, becoming trapped in local minima, and overfitting. In ELM, the input weights with the hidden node parameters are randomly generated, and the output layer weights are later optimized by the regularized least-squares

algorithm directly. In this manner, ELM can achieve universal approximation capability, as well as very high running efficiency. In practice, because of the complexity of various problems, several algorithms for optimizing ELM hidden nodes are proposed to obtain a network of suitable size [17]–[20]. Several algorithms are employed to identify better network parameters to optimize ELM. In [21], a differential evolution algorithm is utilized to adjust ELM input parameters, and the corresponding E-ELM algorithm is proposed. In [22], an effective learning algorithm, known as self-adaptive evolutionary ELM (SaE-ELM), is presented to adjust the control parameters of E-ELM adaptively, which improves the network performance further. Other algorithms try to extend the basic ELM to make it more efficient and more suitable for specific problems [23]–[26].

It is noteworthy that for ELM and its variants, all of the algorithms are composed of two stages: 1) random feature mapping and 2) output weight optimization [26]. However, for complex classification problems, the effect of using random feature mapping to boost the separability of the original sample space is often limited, which increases the dependence on subsequent output weight optimization, and this approach may not be able to achieve the best approximation in the ELM theoretical model. Moreover, most of the current variants of ELM are based on the existing ELM network model and there are few methods for combining the ELM network structure with other network structure adjustments, except for the deep learning network.

To further improve the ELM network performance, this paper takes the improvement of network structure as the starting point and presents a hybrid structure-adaptive RBF-ELM (HSARBF-ELM) network classifier. Unlike previous methods, the proposed method is not restricted by the single ELM model framework, which is independent of the adjustment of the ELM network parameters, and can be considered an indirect effective improvement of the ELM network performance. The presented HSARBF-ELM network consists of a structurally adaptive radial basis function (SARBF) network, which is presented in our previous work [27], and an ELM network of cascade, which can fully mine the spatial distribution information of the classification data set and construct optimized RBF hidden nodes to achieve optimization coverage of the original sample space. In this paper, HSARBF-ELM effectively combines the SARBF network with the ELM network, through the cascade adjustment of the SARBF network and the ELM network. The classification performance of the single ELM network can be notably improved. Specifically, the output of the SARBF network hidden layer is used as the input layer of the ELM network, and an optimized complementary network structure model can be constructed. As the RBF network hidden nodes have good local response characteristics; and the ELM network hidden nodes can adopt a unified sigmoid function, which has global response characteristics, the effective combination of these two different network structures can be considered as the adaptive localization kernel mapping of the original samples. Then, the ELM network is used to complete the global classification of the mapping samples in the kernel space. Thus, HSARBF-ELM indicates the combination of localized kernel mapping learning and the global nonlinear classification. In this way, HSARBF-ELM combines the advantages of the SARBF network and the ELM network, and can improve the ELM network performance effectively.

To quantify and explain the classification performance improvement of the single ELM network due to the presented HSARBF-ELM network classifier, in this paper, the quantitative conditions for the separability enhancement and the corresponding theoretical proof in the SARBF network are given, and the superiority of the HSARBF-ELM network is explained theoretically. This finding demonstrates that when samples go through the SARBF network, adaptively adjusting

the RBF kernel parameters can boost the separability of the original sample space and is beneficial to improving the classification performance of the resulting ELM network. In this way, the classification performance of the HSARBF-ELM network can be guaranteed theoretically. We also briefly analyze several other methods for optimizing the hidden node parameters in RBF networks to further illustrate the advantages of the HSARBF-ELM network.

To ensure the superiority of the proposed network structure, an appropriate learning algorithm for the HSARBF-ELM network is presented. In this paper, we effectively combines the methods of density clustering with a potential function, the center-oriented unidirectional repulsive force and the existing ELM algorithm. A method of density clustering for the sample spatial distribution is utilized to incrementally generate RBF hidden nodes in the HSARBF-ELM network, which is implemented based on the potential function. By density clustering, different regions in the sample space can be adaptively covered by corresponding generated hidden nodes. Each covering corresponds to establishing a new hidden node until the whole sample space is completely covered. In this way, the global information of the sample spatial distribution is effectively utilized. In addition, a method of center-oriented unidirectional repulsive force is utilized to optimize node parameters, which exploits the neighborhood information of the region covered by each initial RBF hidden node in the HSARBF-ELM network. After that step, the width is adjusted to a proper range. This process continues until all RBF hidden node parameters of the HSARBF-ELM network are optimized. Therefore, each center and width and the number of RBF hidden nodes in the HSARBF-ELM network can be estimated effectively. Once all RBF kernel parameters of the HSARBF-ELM network are optimized, the input nodes of the subsequently connected ELM network are determined; thus, the overall HSARBF-ELM network is established, where the connected ELM network parameters can be optimized by the existing ELM algorithm.

An artificial binary classification problem and 14 benchmark data sets from the UCI machine learning repository [28] are utilized to evaluate the performance of HSARBF-ELM. The performances of several typical single-layer feed-forward network(SLFN) training algorithm, including minimal resource allocation network (MRAN) [29], ELM, SaE-ELM and SVM, are compared with the performance of HSARBF-ELM. The results show that the HSARBF-ELM network classifier clearly outperforms the ELM network and outperforms other classifiers on most data sets.

In this paper, our major contributions include the following:

- 1) A HSARBF-ELM network classifier is designed. HSARBF-ELM consists of a SARBF network and an ELM network of cascade, where the output of the SARBF network hidden layer is used as the input layer of the ELM network. In HSARBF-ELM, adaptive localization mapping by the SARBF network can effectively boost the separability of the

- original data set; then, the ELM network can be utilized to provide a superior classification surface. In this way, HSARBF-ELM combines the advantages of the SARB network and the ELM network, and can improve the ELM network performance effectively.
- 2) The quantitative conditions for the separability enhancement and the corresponding theoretical proof in the SARB network are given, and the superiority of the HSARBF-ELM network is explained theoretically. This demonstrates that when input vectors go through the SARB network, adaptively adjusting the RBF kernel parameters can boost the separability of the original sample space, and is beneficial to improving the classification performance of the resulting ELM network. Thus, the classification performance of the HSARBF-ELM network can be guaranteed theoretically.
 - 3) An appropriate learning algorithm for the HSARBF-ELM network is presented, which effectively combines the methods of density clustering with a potential function, center-oriented unidirectional repulsive force and the existing ELM algorithm. A method of density clustering for the sample spatial distribution that makes use of the global information of the sample spatial distribution is utilized to generate RBF hidden nodes in the HSARBF-ELM network incrementally. Next, a method of center-oriented unidirectional repulsive force is utilized to optimize node parameters, which makes full use of the neighborhood information of the region covered by each initial RBF hidden node. Once the input nodes of the subsequently connected ELM network are determined, the connected ELM network parameters can be optimized by the existing ELM algorithm.
 - 4) The quantitative conditions for the separability enhancement are validated on an artificial data set, and the performance of the presented HSARBF-ELM network classifier is compared with those of other typical classifiers on an artificial data set and 14 benchmark data sets. The results show that the HSARBF-ELM network classifier outperforms the ELM network obviously, and outperforms other classifiers on most data sets.

The remainder of this paper is organized as follows. Section II describes the implementation mechanism of ELM. Section III presents the HSARBF-ELM network structure, and provides the separability analysis and corresponding theoretical proof of the HSARBF-ELM network, followed by the learning algorithm of HSARBF-ELM. Section IV presents the performance comparisons between HSARBF-ELM and other classifiers. Section V presents the paper's conclusions.

II. EXTREME LEARNING MACHINE

Given N training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbf{R}^t$ and $y_i \in \mathbf{R}^h$, assume the number of hidden nodes is L . Next, the output of an SLFN is expressed as

$$o_i = \sum_{j=1}^L \beta_j g(a_j \cdot \mathbf{x}_i + b_j), \quad i = 1, 2, \dots, N \quad (1)$$

where $\beta_j \in \mathbf{R}^h$ is the weight vector between the j th hidden node and output nodes, $g(a_j \cdot \mathbf{x}_i + b_j)$ is the output of the j th ELM hidden node, $g(a_j \cdot \mathbf{x}_i + b_j) = \frac{1}{1+e^{-a_j \cdot \mathbf{x}_i + b_j}}$, and $a_j \in \mathbf{R}^t$ and $b_j \in \mathbf{R}$ are learning parameters of the j th hidden node.

Traditionally, to minimize the objective function $E = \sum_{i=1}^N ||o_i - y_i||$, the network weight parameters are updated iteratively based on the gradient descent method, which can be expressed as the following equation:

$$W_k = W_{k-1} - \eta \frac{\partial E(W_{k-1})}{\partial W_{k-1}} \quad (2)$$

where W is the parameter set of (a_j, b_j, β_j) , and η is the learning rate. In feedforward neural networks, a back-propagation algorithm based on gradient descent is usually used to optimize the weights of each layer; however, this algorithm often suffers from a slow convergence rate, becoming trapped in local minima, and overfitting.

Unlike traditional gradient descent algorithms, in ELM, L hidden nodes with an activation function $g(\mathbf{x})$ can be used to approximate the objective function with arbitrary precision, which can be denoted as $\sum_{i=1}^N ||o_i - t_i|| < \varepsilon$, where ε is an arbitrarily small value, i.e., there exist β_j , a_j and b_j such that the approximation can be expressed as

$$\mathbf{H}\beta = \mathbf{Y} \quad (3)$$

where β is the output weight matrix, \mathbf{H} is the output matrix of the hidden layer, and \mathbf{Y} is the target output, with $\mathbf{H} = \{h_{ij}\}$, $h_{ij} = g(a_j \cdot \mathbf{x}_i + b_j)$, $\beta = (\beta_1 \beta_2 \dots \beta_L)$ and $\mathbf{Y} = (y_1 y_2 \dots y_N)$.

On the basis of the proofs of ELM [6], when $g(\cdot)$ is infinitely differentiable in any interval, by generating hidden node parameters randomly and later adjusting the output weights properly, SLFNs can work as universal approximators. The output weights can be optimized as

$$\hat{\beta} = \mathbf{H}^\dagger \mathbf{Y} \quad (4)$$

where \mathbf{H}^\dagger is the Moore-Penrose generalized inverse of matrix.

III. HSARBF-ELM

We first present the network structure of HSARBF-ELM, and then give theoretical explanation of HSARBF-ELM. Next, we present the HSARBF-ELM algorithm.

A. NETWORK STRUCTURE OF HSARBF-ELM

HSARBF-ELM consists of a SARB network and an ELM network of cascade, where the output of the SARB network hidden layer is used as the input layer of the ELM network. In the HSARBF-ELM network classifier, the SARB network is utilized to achieve adaptively localizing kernel mapping of input vectors. After that step, the ELM network is utilized to implement global nonlinear classification of mapping samples. Adaptive localization mapping by SARB can effectively boost the separability of the original data set, while the resulting ELM network can provide a superior classification surface. Thus, HSARBF-ELM indicates the combination of localized kernel mapping learning and the global nonlinear classification. In this way,

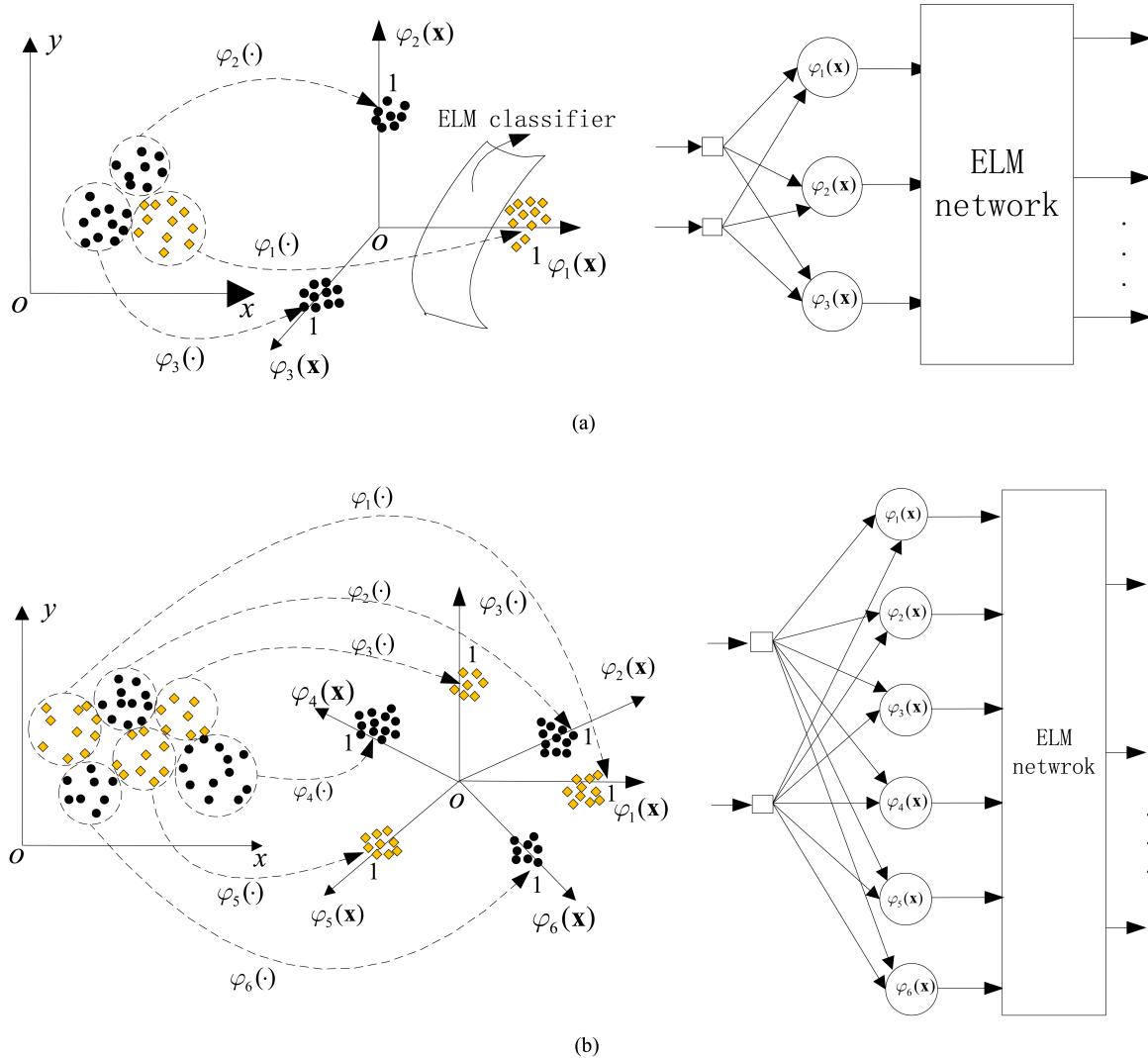


FIGURE 1. Illustrative diagram of different data sets going through the HSARBF-ELM network. (a) The mapping dimension is 3. (b) The mapping dimension is 6.

HSARBF-ELM combines the advantages of the SARBF network and the ELM network, and can improve the ELM network performance effectively. To illustrate the situation, Fig. 1 shows a diagram of different data sets going through the HSARBF-ELM network.

HSARBF-ELM consists of four layers, as illustrated in Fig. 2, and each layer's specific information is given below:

1. The input layer. This layer consists of t nodes, where $\mathbf{x} \in R^t$ is the input vector and t is the dimensionality of \mathbf{x} .
2. The SARBF hidden layer. This layer consists of K nodes, which are described by a set of Gaussian kernels:

$$\varphi_k(\mathbf{x}) = \exp\left(-\frac{1}{2\sigma_k^2} \|\mathbf{x} - \mu_k\|^2\right), \quad k = 1, 2, \dots, K \quad (5)$$

where μ_k is the center, σ_k is the width, and K determines the SARBF network size. For different classification problems, K is adaptively adjusted according to the spatial distributions of the data sets.

3. The ELM hidden layer. This layer consists of L nodes. Assume that $\varphi(\mathbf{x})$ is the input vector of the ELM hidden layer, here $\varphi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \dots, \varphi_K(\mathbf{x}))$. The ELM hidden layer output matrix is \mathbf{H} , where $\mathbf{H} = \{h_{ij}\}$ and $h_{ij} = g(a_j \cdot \varphi(\mathbf{x}_i) + b_j)$, and the ELM hidden node parameters $\{a_j, b_j, L\}_{j=1}^L$ can be generated randomly.

4. The Output layer. The output of the HSARBF-ELM network can be given as follows:

$$o_i = \sum_{j=1}^L \beta_j g(a_j \cdot \varphi(\mathbf{x}_i) + b_j), \quad i = 1, 2, \dots, N \quad (6)$$

where $\hat{\beta} = \mathbf{H}^\dagger \mathbf{Y}$, \mathbf{H}^\dagger is the Moore-Penrose generalized inverse of matrix \mathbf{H} and $\mathbf{Y} = (y_1 y_2 \dots y_N)$ is the target output.

B. THEORETICAL EXPLANATION OF HSARBF-ELM

To quantify and explain the classification performance improvement of the single ELM network due to the

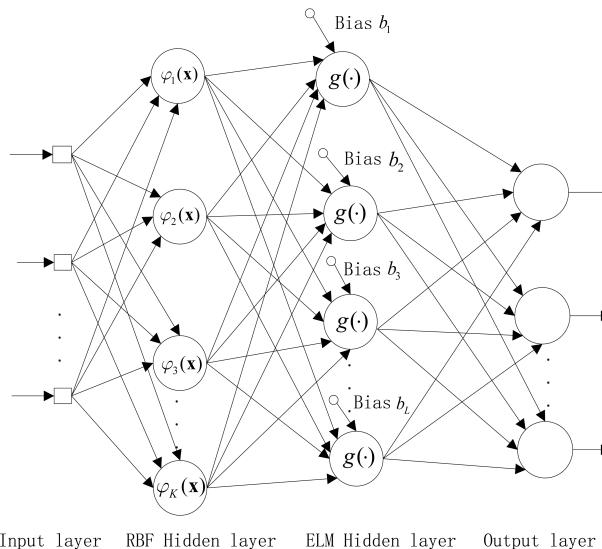


FIGURE 2. Network structure of HSARBF-ELM.

presented HSARBF-ELM network classifier, in this section, the quantitative conditions for the separability enhancement and the corresponding theoretical proof are given, and the superiority of the HSARBF-ELM network is explained theoretically. To clarify the situation, we consider two cases:

- 1) The samples do not go through the SARBF network, and each sample is directly taken as the center of the RBF hidden nodes to complete high-dimensional mapping;
- 2) The samples go through the SARBF network, where different hidden nodes of the SARBF network cover different regions of the sample space.

Let K be the number of RBF hidden nodes. The training set $X = \{x_i\}_{i=1}^N$, where $x_i \in R^n$, is divided into two categories: $X_1 = \{x_i\}_{i=1}^{N_1}$ and $X_2 = \{x_i\}_{i=N_1+1}^{N_1+N_2}$ are the first-and second-category sample sets, respectively. Here, $X = X_1 \cup X_2$ and $N = N_1 + N_2$. Let S_B and S_T denote the between-class scatter matrix and total scatter matrix of the training set, respectively, and let \hat{S}_B and \hat{S}_T denote the between-class scatter matrix and total scatter matrix after the samples go through the RBF network.

For case 1), we have the following lemmas

Lemma 1: If $K = N$, when meeting the condition

$$\frac{N}{N-1} \frac{\sum_{i=1}^N \|x_i - \mu_0\|^2}{\|\mu_1 - \mu_2\|^2} \geq \frac{\sum_{i=1}^N \varphi_i^2}{\frac{1}{N_1^2} \sum_{i=1}^{N_1} \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=N_1+1}^N \varphi_i^2},$$

we can obtain $\frac{\text{tr}(\hat{S}_B)}{\text{tr}(\hat{S}_T)} \geq \frac{\text{tr}(S_B)}{\text{tr}(S_T)}$, where μ_1 , μ_2 , and μ_0 are the means of the first-category, second-category and all training samples, respectively. φ_i is the output of x_i going through the i th RBF hidden node.

The proof process is detailed in the Appendix.

Lemma 1 shows states that if $K = N$, when the original samples go through the RBF network, once the ratio of the total scatter to the between-class scatter in the original samples is no less than the ratio of the sum of squares of all mapped values to the mean-square sum of the mapped values for each pattern category, the separability of the original input vectors can be improved. $K = N$ means that when the original samples go through the RBF network, the region covered by each hidden node may not contain many other samples; therefore, lemma 1 is especially suitable for the situation where the size of the training set is small, and the dimension of the sample space is relatively high. To boost the separability of the original sample set, we need to adjust the center of each hidden node to make φ_i reach a suitable value; however, if the size of the training set is relatively large or the sample space is relatively dense, the number of corresponding mapping parameters φ_i is too large, resulting in difficulty making the adjustment in practice. Thus, for this situation, constructing optimized RBF hidden nodes is necessary, which can be utilized to boost the separability of the original sample set.

For case 2), we have the following lemmas.

Lemma 2: If $K < N$, when meeting the condition

$$\frac{\sum_{i=1}^N \|x_i - \mu_0\|^2}{\|\mu_1 - \mu_2\|^2} \geq \frac{\frac{1}{N} \sum_{i=1}^K m_i(N-m_i)\varphi_i^2}{\frac{1}{N_1^2} \sum_{i=1}^l m_i^2 \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=l+1}^K m_i^2 \varphi_i^2},$$

we can obtain $\frac{\text{tr}(\hat{S}_B)}{\text{tr}(\hat{S}_T)} \geq \frac{\text{tr}(S_B)}{\text{tr}(S_T)}$, where μ_1 , μ_2 , and μ_0 are the means of the first-category, second-category and all training samples, respectively. The parameter l is the number of generated RBF hidden nodes corresponding to the first-category samples. The parameter m_i denotes the number of training samples covered by the i th RBF hidden node, and φ_i is the output of m_i training samples going through the i th RBF hidden node.

The proof process is detailed in the Appendix.

Lemma 2 states that if $K < N$, when the original samples go through the SARBF network, once the ratio of the total scatter to the between-class scatter in the original samples is no less than the ratio of the weighed mean-square sum of all mapped values to the weighed mean-square sum of the mapped values for each pattern category, the separability of the original samples can be boosted. Thus, we can adjust the parameters φ_i , m_i , l and K to reach a suitable value. On the one hand, adjusting the values of φ_i , m_i , l and K is equivalent to adjusting to the SARBF network hidden node parameters, including the centers, widths and the number of RBF hidden nodes. On the other hand, the method of density clustering with a potential function and center-oriented unidirectional repulsive force can generate optimized RBF hidden nodes, and each hidden node parameter can be adjusted adaptively, which can be regarded as an efficient approximation of the unknown sample space to obtain the optimal parameters of

the network. In other words, we can always construct suitable parameters to boost the separability of the original sample set. In this way, when the original samples go through the SARBF network, the separability can be boosted.

Note that lemma 2 is established on the premise that each RBF hidden node covers a certain number of samples; if the training sample space is relatively sparse or the width parameter is relatively small, the coverage effect of RBF hidden nodes will be greatly reduced, which may lead to the failure of the density clustering and center-oriented unidirectional repulsive force to some extent. Under these circumstances, the optimization of case 2) is degraded to that of case 1), where the mapping mechanism of the RBF network is still at work. Thus, case 1) can be considered as case 2) in a special situation.

As a comparison, to demonstrate the advantages of the HSARBF-ELM network, we perform a brief analysis of several typical RBF network hidden node training mechanisms: k-means clustering, which is a typical batch learning algorithm [30], and MRAN and growing and pruning RBF (GAP-RBF) [31], which are typical sequential learning algorithms. On the one hand, the method of k-means clustering trains RBF hidden nodes by effectively utilizing the global spatial distribution information of the sample set; however, the number of RBF hidden nodes needs to be given in advance, and once the training sample set changes, the adaptability of the given network is poor. In addition, the width parameter often does not well fit the local region of the sample space, which is determined by the characteristics of the k-means clustering itself. Because of the diversity of the clustering subsets, the established RBF hidden node often cannot optimize the local coverage of the subset, even if we select the number of optimal clusters. In other words, it is inevitable that the established hidden nodes will cover other categories of samples in general. Therefore, the classification performance will be affected to varying degrees.

On the other hand, as typical sequential learning RBF network methods, MRAN and GAP-RBF can generate RBF hidden nodes adaptively; however, the sequential learning algorithms separate the integrity of the sample space, and the global spatial distribution information of the sample space is lost to a certain degree, which may lead to a decrease in classification performance. In other words, the coverage effect of the established RBF hidden nodes is limited, which further increases the burden of the output weight adjustments.

From this point of view, as the SARBF network can automatically generate RBF hidden nodes and adaptively adjust the kernel parameters to cover the sample space, it effectively boosts the separability of the original sample space, which is beneficial to improving the classification performance of the resulting ELM network. Thus, the classification performance of the HSARBF-ELM network can be guaranteed.

C. LEARNING ALGORITHM OF HSARBF-ELM

To ensure the superiority of the proposed network structure, an appropriate learning algorithm for the HSARBF-ELM

network is presented. In this paper, we effectively combine the method of density clustering with a potential function, center-oriented unidirectional repulsive force and the existing ELM algorithm.

Potential functions were first proposed in [32], which contains several different types that can be used for density clustering in the image domain. In this study, we attempt to use a potential function as a basic mathematical tool to build RBF hidden nodes incrementally in the HSARBF-ELM network. The following potential function is chosen:

$$\gamma(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{1 + A \cdot d^2(\mathbf{x}_1, \mathbf{x}_2)} \quad (7)$$

where $\mathbf{x}_1, \mathbf{x}_2$ are input vectors, $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ is the potential between the two input vectors, $d(\mathbf{x}_1, \mathbf{x}_2)$ is the distance between the two input vectors, and A is a constant. Equation (7) indicates when the distance between the two vectors in the data space is small, the potential is relatively large; otherwise, the potential is relatively small. Once the interactions of all vectors are combined, the densities of different regions in the data space can be measured.

Given a dataset S that consists of N training samples $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^n$ and $y_i \in \mathbb{R}^h$, let S_i be the set of N_i feature vectors of the i th pattern category, $S_i = \{\mathbf{x}_1^i, \mathbf{x}_2^i, \dots, \mathbf{x}_{N_i}^i\}$. Thus, $S = \cup_{i=1}^h S_i$, $S_i \cap S_j = \emptyset, \forall i \neq j$. For two arbitrary input vectors $\mathbf{x}_u^i, \mathbf{x}_v^i$ in S_i , their potential is given as

$$\gamma(\mathbf{x}_u^i, \mathbf{x}_v^i) = \frac{1}{1 + A \cdot d^2(\mathbf{x}_u^i, \mathbf{x}_v^i)} \quad (8)$$

Take \mathbf{x}_v^i as the benchmark input vector. The potential of all input vectors except \mathbf{x}_v^i can be calculated as

$$\rho(\mathbf{x}_v^i) = \sum_{u=1, u \neq v}^{N_i} \gamma(\mathbf{x}_u^i, \mathbf{x}_v^i), \quad v = 1, 2, \dots, N_i \quad (9)$$

Once the potential of each input vector in S_i has been calculated, we can select the vector with the maximum corresponding potential to be the initial center. Denoting this vector as \mathbf{x}_p^i , we can get

$$\mu_k = \mathbf{x}_p^i \quad (10)$$

where k denotes the k th generated RBF hidden node in the HSARBF-ELM network.

When the initial width parameter is set, a corresponding RBF hidden node is generated automatically, and can be utilized to cover the corresponding region in the sample space. However, the coverage is based solely on the information of the current pattern category, which may contain input vectors of other pattern categories; thus, the effectiveness of the coverage is affected. Therefore, it is necessary to optimize the center and width of the hidden node; in this study, a method of center-oriented unidirectional repulsive force is presented, which utilizes neighborhood information of the region covered by each initial hidden node in the SARBF network. The main idea is to first determine whether

the region covered by the current SARBF network hidden node contains other categories of input vectors, if it does, there exists a unidirectional repulsive force from the vector to the center, and this force is used for optimization of the center.

According to the above description, for an initial RBF hidden node in the HSARBF-ELM network, set the initial center to μ_k , where $\mu_k = \mathbf{x}_p^i \in S_i$, and denote the initial width parameter as σ . For an arbitrary input vector \mathbf{x}_q^j , if $\mathbf{x}_q^j \notin S_i$ and $\|\mathbf{x}_q^j - \mu_m\| < \lambda \cdot \sigma$, λ can be regarded as the width covering factor; then, there exists a center-oriented unidirectional repulsive force from \mathbf{x}_q^j to μ_k , which is defined as

$$\mathbf{F}_{\mathbf{x}_q^j} \propto \frac{1}{d(\mathbf{x}_q^j, \mu_k)} \cdot \frac{\mathbf{x}_q^j - \mu_k}{\|\mathbf{x}_q^j - \mu_k\|} \quad (11)$$

To express the correspondence between the distance and the center-oriented unidirectional repulsive force, a negative exponential function is chosen, and the center-oriented unidirectional repulsive force can be expressed as

$$\mathbf{F}_{\mathbf{x}_q^j} = \exp(-\alpha \cdot d(\mathbf{x}_q^j, \mu_k)) \cdot \frac{\mathbf{x}_q^j - \mu_k}{\|\mathbf{x}_q^j - \mu_k\|} \quad (12)$$

where α is the repulsive force control factor, which can take a positive constant value. Assume M_j is the number of samples in other categories covered by the current region. The resultant forces can be used for optimization of the center, and can be adjusted as follows:

$$\mu'_k = \mu_k + \sum_{q=1}^{M_j} \mathbf{F}_{\mathbf{x}_q^j} \quad (13)$$

It is noteworthy that for an arbitrary input vector \mathbf{x} , if $\|\mathbf{x} - \mu_k\| < \lambda \cdot \sigma$ and $\mathbf{x} \notin S_i$, M_j is incremented by one. Similarly, assume M_i is the number of samples of the current category in the region; if $\|\mathbf{x} - \mu_k\| < \lambda \cdot \sigma$ and $\mathbf{x} \in S_i$, M_i is incremented by one.

In practice, the adjustment of the center often requires multiple iterations. In this study, we introduce the idea of simulated annealing. When the iteration step is increased, the magnitude of the center can be adjusted to be relatively small. Meanwhile, we set iterative conditions to ensure that the center will gradually converge to a proper location. Therefore, Equation (13) can be corrected as follows:

$$\mu'_k = \mu_k + \frac{1}{M} \sum_{q=1}^{M_j} \mathbf{F}_{\mathbf{x}_q^j} \quad s.t. M'_i \geq M_i \text{ and } M'_j \leq M_j \quad (14)$$

where M is the iteration step, M'_i is the number of samples of the current category in the updated region, and M'_j is the number of samples of other categories covered by the updated region.

Once the center is optimized to a proper location, it is necessary to optimize the width to further improve the coverage effect of the current SARBF network hidden nodes. To reduce the coverage of other categories of samples, decreasing the

width is an option; however, if the width is too small, the coverage of the current category of samples may also be reduced, leading to a reduction in generalization performance. If the current coverage region still contains samples of other categories, the updated width parameter is determined according to the distance between the center in the current region and the nearest sample of other categories. Otherwise, the width remains unchanged. Therefore, the width is adjusted as follows:

$$\sigma_k = \begin{cases} \max\{\min d(\mu'_k, \mathbf{x}_q^j)/\beta, \sigma_{\min}\}, & \text{if } M'_j > 0. \\ \sigma, & \text{if } M'_j = 0. \end{cases} \quad (15)$$

where β can be regarded as the width constraint factor, and σ_{\min} is the constrained minimum width parameter. This adjustment ensures the relative diversity of different SARBF network hidden nodes covering different regions, which can achieve a balance between the coverage effect and generalization performance.

When the current RBF hidden node in the HSARBF-ELM network is generated, all input vector potentials in the region covered by the hidden node need to be counteracted to find the new input vector that corresponds to the maximum potential, which is given by

$$\rho'(\mathbf{x}_n^i) = \rho(\mathbf{x}_n^i) - \rho(\mathbf{x}_p^i) \cdot \exp\left(-\frac{1}{2\sigma_k^2} \|\mathbf{x}_n^i - \mathbf{x}_p^i\|^2\right), \quad n = 1, 2, \dots, N_i \quad (16)$$

where \mathbf{x}_p^i is the initial center of the k th SARBF network hidden node. When the inequality

$$\max\{\rho(\mathbf{x}_1^i), \rho(\mathbf{x}_2^i), \dots, \rho(\mathbf{x}_{N_i}^i)\} > \delta \quad (17)$$

is satisfied, the learning process continues, and the next new initial SARBF network hidden node can be found and established, where δ is a potential learning threshold. Otherwise, the process switches to learning other pattern categories until the end.

It is worth noting that in the HSARBF-ELM network, the centers, widths and number of RBF hidden nodes interact with one another. When the center and width parameters are optimized properly, the coverage effect of each hidden node can be guaranteed. As a result, the network generalization performance is further improved and a suitable network size can be achieved. Fig. 3 shows an illustrative diagram of the optimization of RBF hidden node parameters in the HSARBF-ELM network.

In HSARBF-ELM, when all RBF kernel parameters are optimized, the subsequently connected ELM network parameters can be optimized by the existing ELM algorithm.

Combined with the network structure of HSARBF-ELM, the learning algorithm is summarized in Algorithm 1.

IV. PERFORMANCE EVALUATION OF HSARBF-ELM

The performance of HSARBF-ELM is evaluated with an artificial data set and 14 UCI benchmark data sets: Concrete Circle, Blood Transfusion (BT), Breast Cancer (BC),

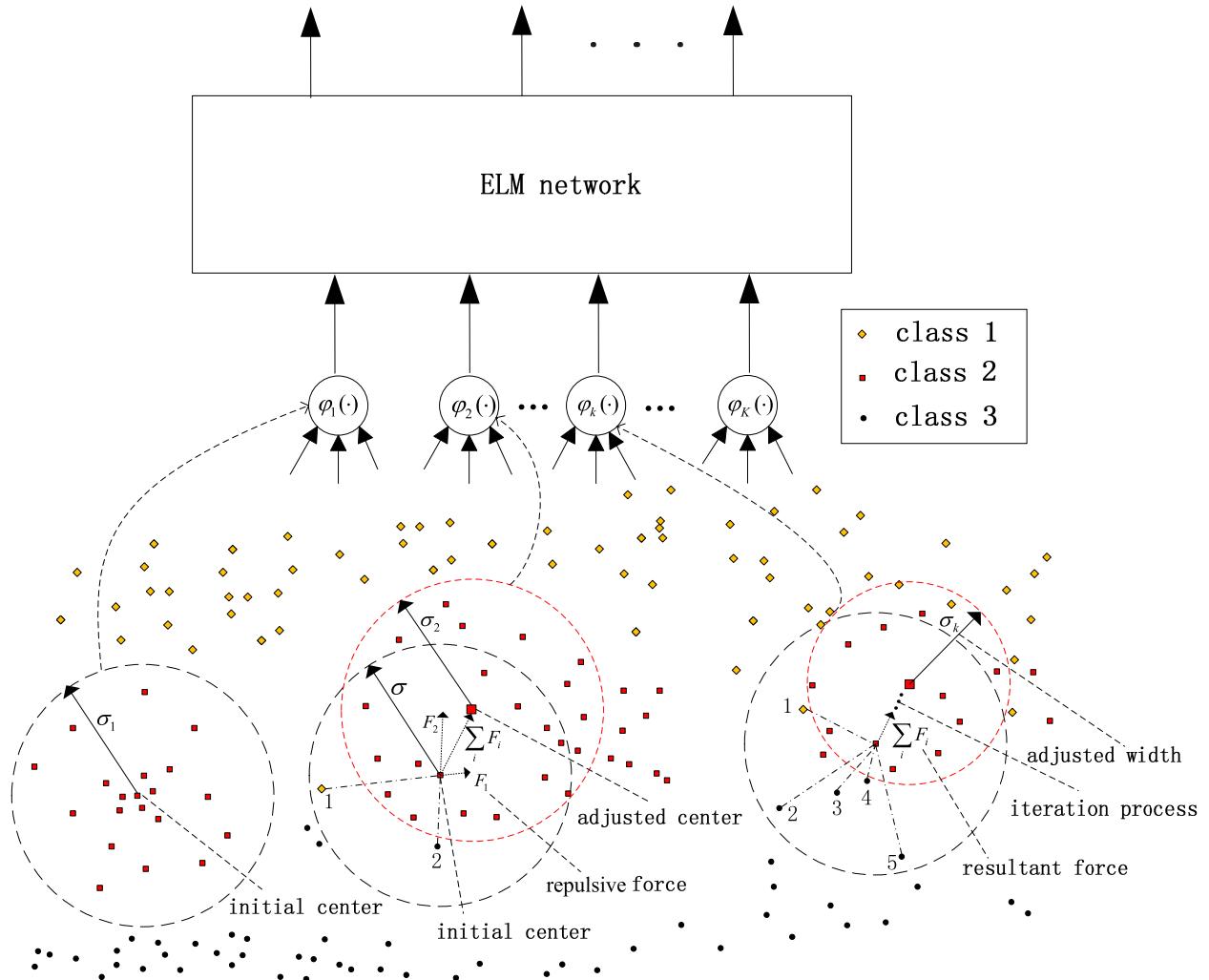


FIGURE 3. Illustrative diagram of optimizing RBF hidden node parameters in HSARBF-ELM.

Pima Diabetes (PD), Heart Disease (HD), Wholesale, Climate, German Credit (GC), Ionosphere, Sonar, Planning Relax (PR), Image Segmentation (IS), Knowledge, Steel Faults (SF) and Forest. The classification data sets are described in Table 1. Except for the IS data set, all benchmark datasets are imbalanced datasets. In all UCI benchmark data sets, the inputs to all classifiers are scaled appropriately in $[-1, 1]$.

The performances of several typical SLFN network training algorithms, namely MRAN, ELM, SaE-ELM and SVM, are compared with the performance of HSARBF-ELM. The main parameters of MRAN are chosen according to [29]. For SVM, ELM and SaE-ELM, the main parameters are chosen as suggested in [22]. The main parameters of HSARBF-ELM are chosen as $A = 1$, $\delta = 0.001$, $\lambda = 1.5$, $\beta = 1.3$, where the initial width parameter σ is chosen in $[0.1, 0.2, \dots, 1.5]$, σ_{\min} is chosen in the set $\{\sigma_{\min} | \sigma - 0.2 \leq \sigma_{\min} < \sigma\}$, and the repulsive force factor α is chosen in $[2^0, 2^1, 2^2, 2^3, 2^4]$. All these simulations are conducted in MATLAB 2013a running on an PC with 3.2GHZ CPU

TABLE 1. Descriptions of UCI classification data sets.

Data sets	No. of Classes	No. of Features	No. of samples		
			Training	Validation	Testing
BT	2	4	374	187	187
BC	2	9	138	70	69
PD	2	8	576	100	92
HD	2	13	151	76	76
Wholesale	2	7	220	110	110
Climate	2	18	270	135	135
GC	2	24	500	250	250
Ionosphere	2	34	175	88	88
Sonar	2	60	104	52	52
PR	2	12	91	46	46
IS	7	19	210	420	1680
Knowledge	4	5	130	128	145
SF	7	20	971	385	385
Forest	4	27	198	100	225

and 4GB RAM. For each algorithm, 20 trials are conducted. For SVM, the RBF kernel is used, and the simulations are implemented with the popular LIBSVM package [33].

Algorithm 1 Learning Algorithm of the HSARBF-ELM Network

```

Initialization;
for i=1 : h % here h is the number of pattern categories
    Use (8) and (9) to calculate the potential of all input vectors;
    while max{ $\rho(\mathbf{x}_1^i), \rho(\mathbf{x}_2^i), \dots, \rho(\mathbf{x}_{N_i}^i)$ } >  $\delta$ 
        Determine the maximum potential value over all samples;
        Use (10) to generate an initial center, the SARBF hidden nodes k is incremented by one;
        Count  $M_i, M_j$ .
        Use (12) and (13) to adjust the initial center, count  $M'_i, M'_j$ .
        while  $M'_j \neq 0 \& \& M \leq \text{num\_Epoch}$ 
            if  $M'_i \geq M_i \& \& M'_j \leq M_j$ 
                 $M_i \leftarrow M'_i, M_j \leftarrow M'_j$ .
            Use (14) to adjust the center, update  $M'_i, M'_j$ .
            Increment M by 1.
        else
            Use (15) to adjust  $\sigma_k$ .
            Break;
        end if
    end while
    Use (16) to counteract the potential of corresponding input vectors covered by the hidden node.
end while
end for
Use (5) to calculate  $\varphi_k(\mathbf{x})$ , set  $\varphi(\mathbf{x})$  as the input of ELM, where  $\varphi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \dots, \varphi_K(\mathbf{x}))$ .
Calculate the output of ELM hidden layer

```

$$h_{ij} = g(a_j \cdot \varphi(\mathbf{x}_i) + b_j)$$

where ELM hidden node parameters $\{a_j, b_j\}_{j=1}^L$ are randomly assigned.

Calculate the output weight matrix

$$\hat{\beta} = \mathbf{H}^\dagger \mathbf{Y}$$

where \mathbf{H}^\dagger is the Moore-Penrose generalized inverse of ELM hidden layer output matrix \mathbf{H} and $\mathbf{Y} = (y_1 y_2 \dots y_N)$ is the target output.

A. PERFORMANCE MEASURES

In this paper, the network classification performance is measured by the overall (η_o) and average (η_a) per-category classification accuracies. Let N denote the number of testing samples, N_i^T the number of testing samples in the i th pattern category, q_{ii} the number of correct testing samples in the i th pattern category from the classification, and h the number of pattern categories. In this instance, η_o and η_a can be defined as

$$\eta_o = 100 \times \frac{1}{N} \sum_{i=1}^h q_{ii} \quad (18)$$

$$\eta_a = 100 \times \frac{1}{h} \sum_{i=1}^h \frac{q_{ii}}{N_i^T} \quad (19)$$

B. PERFORMANCE COMPARISONS

1) ARTIFICIAL DATA SET

The Concentric Circle data set is plotted in Fig. 4, which is mainly used to verify the characteristics of our presented method. For HSARBF-ELM, the number of ELM hidden

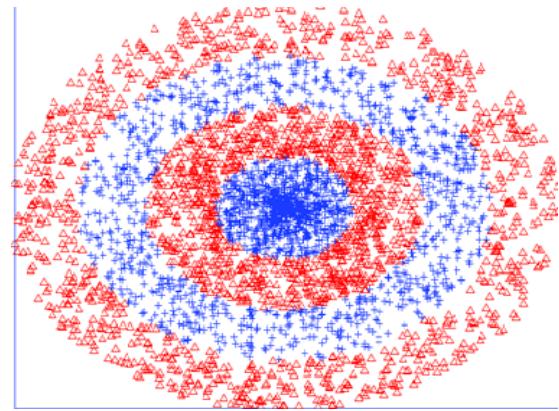


FIGURE 4. Concentric Circle classification problem.

nodes is set to $L = 100$. Fig. 5 shows the coverage effect of our presented method on the Concentric Circle data set. Each generated RBF hidden node effectively covers a local region in the training sample space, where the bold circle represents the region covered by the first generated hidden node and can be regarded as the densest region in its pattern

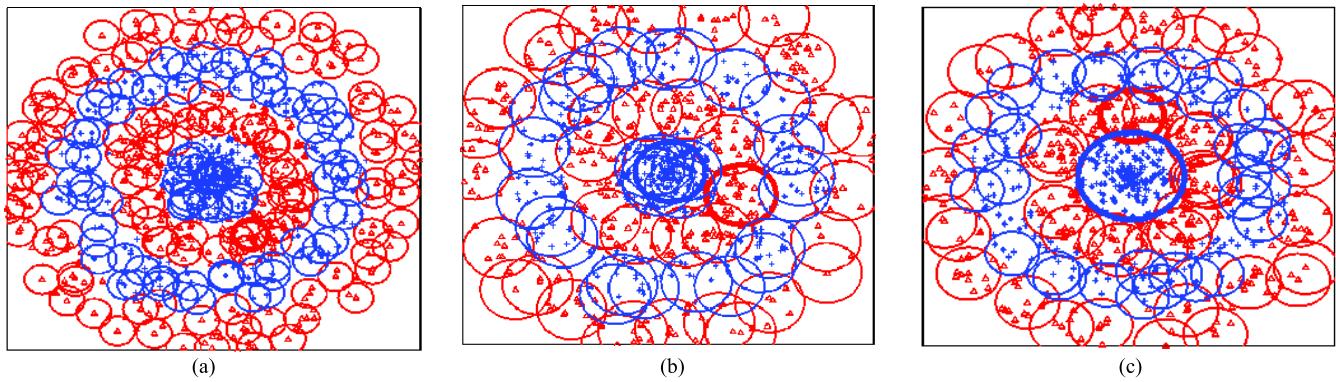


FIGURE 5. Using different width parameter to cover the training sample space. (a) $\sigma = 0.1$ (b) $\sigma = 0.2$ (c) $\sigma = 0.3$.

category. Even if the initial width parameter takes different values, the methods of density clustering with a potential function and center-oriented unidirectional repulsive force can ensure a relatively good learning effect on the training sample space. In this way, the optimized network parameters can be estimated effectively.

To confirm that the SARBF network can boost the separability of the original data space, we verify lemma 2 by calculating the parameters generated in the learning process. Note that the derivation of theoretical conditions in lemma 2 is based on the ideal covering map; however, there is a certain overlap between different regions covered by different hidden nodes in practice. Thus, it is not possible to achieve the ideal mapping completely. Nevertheless, we still calculate the corresponding parameters with the actual coverage effect. The purpose is to show that the separability of the original data space can be boosted once effective coverage is achieved. Meanwhile, to simplify the calculation, the mapping values of the samples covered by each RBF hidden node are set to 1.

TABLE 2. Main parameters of the SARBF network.

$N = 800, N_1 = 413, N_2 = 387, K = 65, l = 27.$
$\mu_1 = (0.0026, 0.0180)^T, \mu_2 = (0.0191, 0.0044)^T.$
$\mu_0 = (0.0104, 0.0115)^T.$
$m_1 \sim m_{27} : 38, 33, 35, 29, 18, 20, 19, 16, 17, 15, 16, 15,$
$14, 14, 13, 13, 12, 11, 12, 11, 9, 9, 8, 7, 6, 5, 5.$
$m_{28} \sim m_{65} : 93, 84, 81, 72, 57, 28, 25, 26, 22, 20, 21, 18,$
$19, 18, 16, 16, 17, 15, 14, 14, 15, 14, 13, 12, 13,$
$14, 11, 10, 11, 12, 11, 11, 10, 11, 9, 10, 9, 9.$

According to the above setting, the obtained process parameters of the SARBF network are shown in Table 2.

On the one hand, in Lemma 2,

$$\frac{\sum_{i=1}^N \|x_i - \mu_0\|^2}{\|\mu_1 - \mu_2\|^2} = \frac{377.6665}{4.5721 \times 10^{-4}} \approx 8.26 \times 10^5$$

On the other hand,

$$\frac{\frac{1}{N} \sum_{i=1}^K m_i(N - m_i)\varphi_i^2}{\frac{1}{N_1^2} \sum_{i=1}^l m_i^2 \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=l+1}^K m_i^2 \varphi_i^2} \approx \frac{\frac{943914}{800}}{\frac{8554}{413^2} + \frac{38927}{387^2}} \approx 3806$$

By comparing the above calculation results, the theoretical conditions in lemma 2 can be verified.

Fig. 6 (a) demonstrates that with the change in sample size, HSARBF-ELM generates fewer RBF hidden nodes than the other algorithms, which further illustrates the validity of the overall coverage by the generated hidden nodes; thus, a relatively compact network size can be obtained. In Fig. 6 (b), HSARBF-ELM has better overall classification accuracy than other algorithms in most cases. However, when the number of training samples is 200, the overall classification accuracy of HSARBF-ELM is slightly lower than that of SVM. To explain this case, in Fig. 7, we can see that an inadequate number of training samples leads to a reduction in the learning effect. When the training set is limited in size, the randomness of the spatial distribution is enhanced, resulting in failure of the density clustering with the potential function and center-oriented unidirectional repulsive force to some extent. In this case, other learning algorithms will also fail to varying degrees.

2) BENCHMARK BINARY CATEGORY DATA SETS

In this section, we use ten UCI binary category data sets to compare HSARBF-ELM with other classifiers. The results are shown in Table 3. HSARBF-ELM outperforms the other classifiers in terms of overall and average per-category classification accuracies on 8 data sets: BT, BC, PD, HD, Wholesale, Climate, GC and Ionosphere. However, for the Sonar data set, the overall efficiency of HSARBF-ELM is lower than that of SVM by approximately 4.3%, and the average per-category efficiency of HSARBF-ELM is lower than that of SVM by approximately 9.5%. For the PR data set, the overall efficiency of HSARBF-ELM is lower than that of SVM by approximately 2.2%, and the average per-category efficiency of HSARBF-ELM is lower than that of SVM by

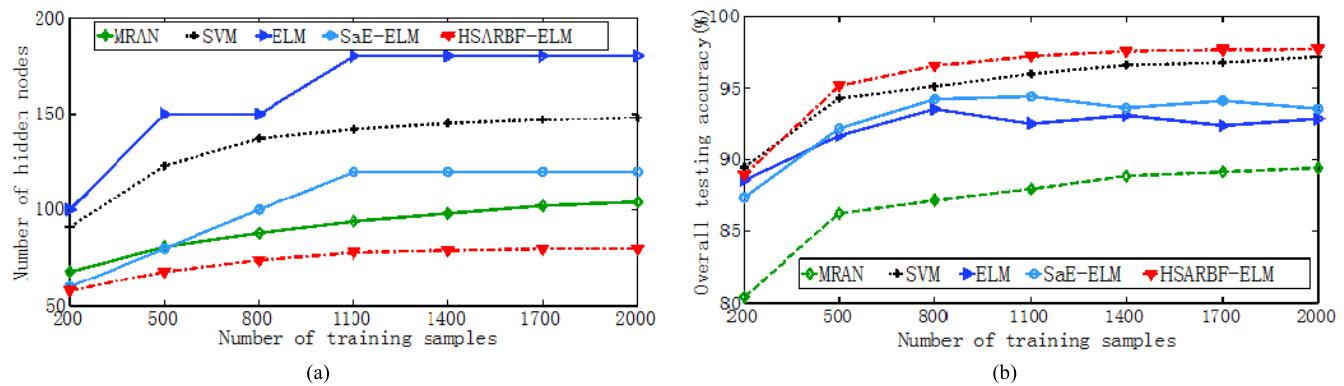


FIGURE 6. Performance comparisons between HSARBF-ELM and other classifiers. (a) Number of training samples-Number of hidden neurons. (b) Number of training samples-Overall testing accuracy (%).

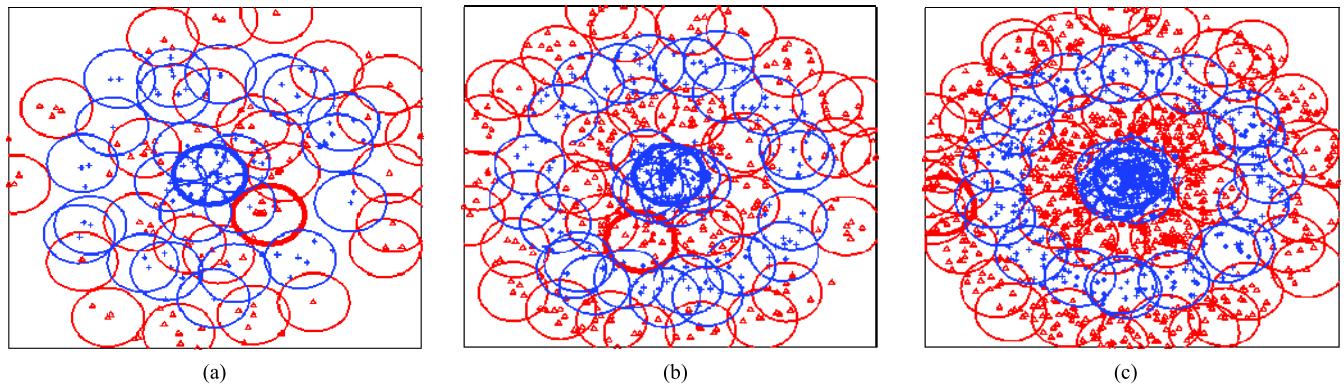


FIGURE 7. The coverage effect of training sample space when N takes different values. (a) $N = 200$ (b) $N = 600$ (c) $N = 1500$.

approximately 1.9%. For a small number of training data sets, such as the Sonar and PR data sets, the actual spatial distribution of whole data set is not effectively reflected in general; in particular, for high-dimensional data sets, the spatial distribution in the sample space is often relatively sparse, resulting in enhancement of the randomness of the spatial distribution. Therefore, the learning effect in the training sample space may degrade to some extent, and the classification accuracy on the testing set may decrease.

Among these ten benchmark binary data sets, the RBF network size in HSARBF-ELM is lower than that of MRAN on most data sets. Because the presented classifier is based on optimizing the coverage of the sample space, it can use fewer RBF hidden nodes and get better classification performance. In addition, the number of ELM hidden nodes in HSARBF-ELM is lower than that in the ELM network, which demonstrates that HSARBF-ELM reduces the dependence on the ELM hidden node parameters selection, and achieves better classification performance.

3) BENCHMARK MULTI- CATEGORY DATA SETS

In this section, we use four UCI multi-category data sets to compare HSARBF-ELM with other classifiers. The results are shown in Table 4. The overall and average per-category classification accuracies of HSARBF-ELM outperform those

of other classifiers on the IS, Knowledge and SF data sets. For the Forest data set, the overall efficiency of HSARBF-ELM is lower than that of SVM by approximately 2.6%, and the average per-category efficiency of HSARBF-ELM is lower than that of SVM by approximately 5.7%; however, the overall and average testing accuracies of HSARBF-ELM are higher than those of MRAN, ELM, and SaE-ELM by approximately 0.5%-4.1%.

It is worth noting that for all data sets, the classification performance of HSARBF-ELM is higher than that of ELM, which further shows that SARBF can boost the classification performance of the original sample space, and our presented HSARBF-ELM classifier is effective.

C. DISCUSSION

1) INFLUENCE OF THE INITIAL WIDTH ON HSARBF-ELM

When the initial width varies, the different coverage regions of hidden nodes cause the number of samples to vary, leading to differences in the adjusted center and width parameters, and the generation of different numbers of RBF hidden nodes. To measure the influence of the initial width on HSARBF-ELM, three benchmark date sets, namely, PD, Ionosphere and IS, are used in this section. In Fig. 8, we can see that when the initial width value is too small, the generated RBF network size in HSARBF-ELM is relatively large, and the

TABLE 3. Performance comparisons on benchmark binary category data sets.

Data sets	Methods	Training time (s)	Testing		Hidden nodes
			η_o	η_a	
BT	MRAN	13.64	75.83	73.21	177
	SVM	5.1	77.27	75.53	265 ^a
	ELM	0	76.48	75.32	80
	SaE-ELM	0.14	77.64	76.28	40
	HSARBF-ELM	11.54	79.93	78.81	109&40 ^b
BC	MRAN	8.64	63.25	64.42	87
	SVM	0.2	79.86	80.64	94 ^a
	ELM	0	62.82	63.28	60
	SaE-ELM	0.06	65.71	65.24	50
	HSARBF-ELM	7.83	84.27	84.89	95&60 ^b
PD	MRAN	6.48	75.39	74.73	42
	SVM	4.92	80.63	77.41	301 ^a
	ELM	0	78.64	78.36	50
	SaE-ELM	0.21	80.13	79.72	22
	HSARBF-ELM	2.82	82.62	83.14	19&30 ^b
HD	MRAN	0.61	78.16	77.94	29
	SVM	0.08	81.70	85.85	42 ^a
	ELM	0	79.56	78.62	30
	SaE-ELM	0.15	82.52	82.42	20
	HSARBF-ELM	0.74	84.37	84.76	12&20 ^b
Wholesale	MRAN	2.53	87.55	85.21	41
	SVM	1.07	90.84	90.42	55 ^a
	ELM	0	88.92	87.31	60
	SaE-ELM	0.25	90.41	89.45	30
	HSARBF-ELM	6.67	92.89	92.13	36&30 ^b
Climate	MRAN	2.86	88.26	86.68	27
	SVM	1.93	92.32	92.64	49 ^a
	ELM	0	91.85	91.53	50
	SaE-ELM	0.18	92.51	91.94	30
	HSARBF-ELM	4.42	94.28	93.36	13&20 ^b
GC	MRAN	14.64	68.45	66.29	137
	SVM	3.74	77.38	76.39	234 ^a
	ELM	0	75.54	70.50	120
	SaE-ELM	0.31	77.51	72.63	80
	HSARBF-ELM	18.42	83.40	82.78	258&50 ^b
Ionosphere	MRAN	1.47	84.34	82.29	86
	SVM	0.61	91.26	91.82	50 ^a
	ELM	0	89.77	85.93	100
	SaE-ELM	0.35	90.75	88.14	60
	HSARBF-ELM	2.87	94.89	94.42	54&40 ^b
Sonar	MRAN	3.37	67.26	65.29	74
	SVM	0.12	80.85	84.97	46 ^a
	ELM	0	70.37	70.06	50
	SaE-ELM	0.07	70.24	69.62	30
	HSARBF-ELM	1.32	76.52	75.42	49&20 ^b
PR	MRAN	0.87	67.24	65.86	26
	SVM	0.16	71.43	70.52	69 ^a
	ELM	0	65.47	66.23	30
	SaE-ELM	0.07	66.52	66.58	20
	HSARBF-ELM	1.13	69.25	68.67	23&20 ^b

^a Support vectors.^b RBF&ELM hidden nodes.

overall classification accuracy is relatively poor, e.g., for the Ionosphere dataset, when $\sigma = 0.2$, the number of generated RBF hidden nodes in HSARBF-ELM is 92, and the overall testing accuracy is 60.64%. The main reason is that if the initial width parameter takes too small, the established hidden nodes may not effectively cover the sample space. In this situ-

TABLE 4. Performance comparisons on benchmark multi-category data sets.

Data sets	Methods	Training time (s)	Testing		Hidden nodes
			η_o	η_a	
IS	MRAN	11.42	85.49	--	81
	SVM	11.93	90.56	--	96 ^a
	ELM	0	90.31	--	100
	SaE-ELM	0.14	91.17	--	50
	HSARBF-ELM	3.82	92.86	--	32&40 ^b
Knowledge	MRAN	2.62	78.25	76.13	65
	SVM	1.75	80.65	81.23	84 ^a
	ELM	0	79.07	78.61	50
	SaE-ELM	0.05	80.24	79.25	30
	HSARBF-ELM	2.81	83.61	82.76	56&40 ^b
SF	MRAN	26.83	64.56	62.82	313
	SVM	86.24	71.51	70.87	263 ^a
	ELM	0.07	72.38	71.53	200
	SaE-ELM	0.62	72.85	71.61	130
	HSARBF-ELM	37.67	73.84	72.75	274&150 ^b
Forest	MRAN	12.60	66.25	64.71	73
	SVM	14.84	72.19	74.53	125 ^a
	ELM	0	68.56	67.63	60
	SaE-ELM	0.23	68.84	68.38	40
	HSARBF-ELM	5.71	69.52	68.86	51&30 ^b

^a Support vectors.^b RBF&ELM hidden nodes.

ation, the methods of density clustering with a potential function and center-oriented unidirectional repulsive force will fail to some extent, and a relatively large network size leads to lower generalization performance; however, if the initial width parameter takes a relatively large value, the network size decreases and the overall classification performance is improved. In particular, when the initial width σ is between 0.9–1.5, the overall testing accuracy is between 92.97% and 94.89%, thus a relatively high classification performance can be obtained. This shows that the HSARBF-ELM network has better adaptability to the initial width. In this respect, HSARBF-ELM can well adapt to the sample space and establish an RBF network of optimum size if the initial width is not too small.

2) LIMITATIONS FOR HSARBF-ELM

In HSARBF-ELM, to obtain good network performance, the training set should be of a certain size; in particular, for high-dimensional data sets, the spatial distribution in the sample space is often relatively sparse, resulting in the enhancement of the randomness of the spatial distribution. As a result, the spatial distribution of the whole data set cannot be effectively represented. Therefore, the learning effect in the training sample space may decrease to some extent, and the classification accuracy on the testing set may decrease.

To ensure the adequacy of the sample space coverage by different RBF hidden nodes, for HSARBF-ELM, the initial width parameter should not be too small. Otherwise, the network size in the RBF hidden layer will be greatly increased,

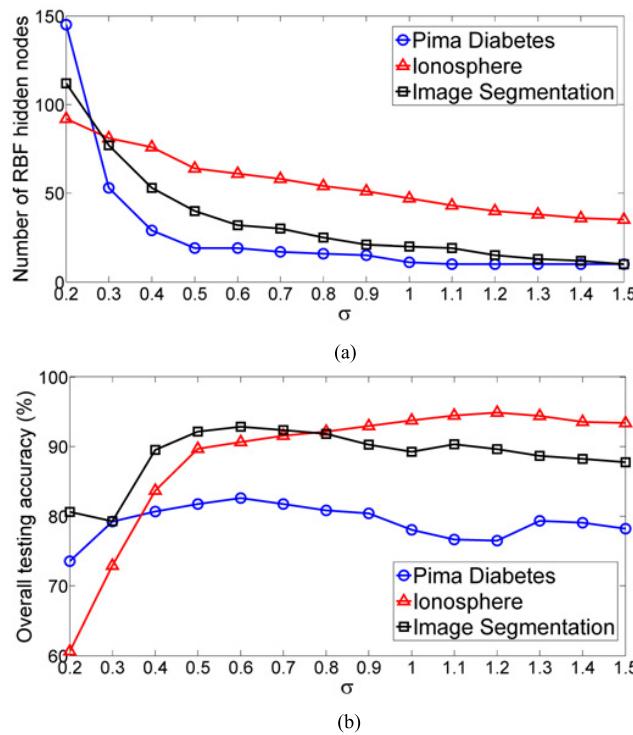


FIGURE 8. The influence of initial width on HSARBF-ELM. (a) σ - Number of RBF hidden nodes. (b) σ - Overall testing accuracy (%).

which leads to a reduction in generalization ability. In addition, in this situation, the number of samples covered by each generated RBF hidden node is too small, which leads to the failure of the learning algorithm.

V. CONCLUSION

In this paper, a HSARBF-ELM network classifier is presented. The presented network structure is not restricted by the single ELM model framework, and is independent of the adjustment of the ELM network parameters. The HSARBF-ELM network consists of a SARBF network and an ELM network of cascade, where the output of the SARBF network hidden layer is used as the input layer of the ELM network, and an optimized complementary network structure model can be constructed. In HSARBF-ELM, adaptive localization mapping by the SARBF network can boost the separability of the original data set; then, the ELM network can be utilized to complete the global classification of the mapping samples in the kernel space and provide a superior classification surface. HSARBF-ELM indicates the combination kernel mapping learning and the global nonlinear classification, which combines the advantages of the SARBF network and the ELM network.

The quantitative conditions for the separability enhancement and the corresponding theoretical proof in the SARBF network are given, which demonstrate that when input vectors go through the SARBF network, adaptively adjusting the RBF kernel parameters can boost the separability of the original sample space and is beneficial to improving the classification performance of the resulting

ELM network. In this way, HSARBF-ELM can boost the classification performance of the single ELM network theoretically, and the classification performance of the presented HSARBF-ELM network can be further guaranteed.

To ensure the superiority of the proposed network structure, an appropriate learning algorithm for the HSARBF-ELM network is then presented, which effectively combines the method of density clustering with a potential function, the center-oriented unidirectional repulsive force and the existing ELM algorithm. The method of density clustering with a potential function makes use of the global information of the sample spatial distribution to generate initial RBF hidden nodes, and the method of center-oriented unidirectional repulsive force is further utilized to optimize node parameters, which makes full use of the neighborhood information of the region covered by each initial RBF hidden node. In this way, the mapping of samples in kernel space can guarantee better classification performance for the resulting ELM network.

The performance of the presented HSARBF-ELM classifier is compared with those of other typical classifiers on an artificial data set and 14 benchmark data sets. The results show that HSARBF-ELM outperforms the other classifiers on most data sets, and HSARBF-ELM outperforms ELM obviously, which accords with the conclusion of the theoretical analysis. In addition, among these different classification problems, the number of ELM hidden nodes in HSARBF-ELM is lower than that in the ELM network, which demonstrates that HSARBF-ELM reduces the dependence on the ELM hidden node parameter selection and achieves better classification performance.

However, the method of density clustering with a potential function and the iterations of the center-oriented unidirectional repulsive force increase the computational burden. In this paper, HSARBF-ELM is primarily used for supervised classification. In many cases, obtaining enough labels for supervised tasks is expensive; thus, in the future, we will examine how to overcome the weaknesses of supervised learning and utilize unlabeled samples.

APPENDIX

Lemma 1 Proof: According to the definition of the between class scatter matrix and total scatter matrix,

$$S_B = \sum_{i=1}^2 N_i(\mu_i - \mu_0)(\mu_i - \mu_0)^T,$$

$$S_T = \sum_{i=1}^N (x_i - \mu_0)(x_i - \mu_0)^T$$

Set $\rho_i = N_i/N$. Then

$$\mu_0 = \frac{1}{N} \sum_{k=1}^N x_k = \sum_{i=1}^c \frac{N_i}{N} \cdot \frac{1}{N_i} \sum_{k=1}^{N_i} x_k^i = \sum_{i=1}^c \frac{N_i}{N} \mu_i = \sum_{i=1}^c \rho_i \mu_i,$$

thus

$$S_B = N_1[\mu_1 - (\rho_1 \mu_1 + \rho_2 \mu_2)][\mu_1 - (\rho_1 \mu_1 + \rho_2 \mu_2)]^T$$

$$\begin{aligned}
& + N_2[\mu_2 - (\rho_1\mu_1 + \rho_2\mu_2)][\mu_2 - (\rho_1\mu_1 + \rho_2\mu_2)]^T \\
& = N_1[\rho_2\mu_1 - \rho_2\mu_2][\rho_2\mu_1 - \rho_2\mu_2]^T \\
& \quad + N_2[\rho_1\mu_1 - \rho_1\mu_2][\rho_1\mu_1 - \rho_1\mu_2]^T \\
& = N_1\rho_2^2(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T + N_2\rho_1^2 \\
& \quad \times (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \\
& = \left(\frac{N_1N_2^2}{N^2} + \frac{N_2N_1^2}{N^2} \right) (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \\
& = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T N_1N_2/N
\end{aligned}$$

Let $\hat{\mu}_1$, $\hat{\mu}_2$, and $\hat{\mu}_0$ be the output means of the first-category, second-category and all training samples going through the RBF network, respectively. Similarly, we can obtain

$$\begin{aligned}
\hat{S}_B &= (\hat{\mu}_1 - \hat{\mu}_2)(\hat{\mu}_1 - \hat{\mu}_2)^T N_1N_2/N \\
\hat{S}_T &= \sum_{i=1}^N (\hat{x}_i - \hat{\mu}_0)(\hat{x}_i - \hat{\mu}_0)^T
\end{aligned}$$

To simplify the problem and facilitate theoretical analysis, for an arbitrary sample x_i , let \hat{x}_i denote the output of x_i from the RBF network, where $\hat{x}_i = (0, 0, \dots, \varphi_i, 0, \dots, 0)$.

On the one hand,

$$\begin{aligned}
tr(\hat{S}_B)tr(S_T) &= tr[((\hat{\mu}_1 - \hat{\mu}_2)(\hat{\mu}_1 - \hat{\mu}_2)^T N_1N_2/N)] \\
&\quad \cdot tr[\sum_{i=1}^N (x_i - \mu_0)(x_i - \mu_0)^T] \\
&= ||\hat{\mu}_1 - \hat{\mu}_2||^2 N_1N_2/N \cdot \sum_{i=1}^N ||x_i - \mu_0||^2
\end{aligned}$$

As

$$\begin{aligned}
\hat{\mu}_1 &= \frac{1}{N_1} \sum_{i=1}^{N_1} \hat{x}_i = \frac{1}{N_1} (\varphi_1, \dots, \varphi_{N_1}, 0, \dots, 0)^T, \\
\hat{\mu}_2 &= \frac{1}{N_2} \sum_{i=1}^{N_2} \hat{x}_i = \frac{1}{N_2} (0, \dots, 0, \varphi_{N_1+1}, \varphi_{N_1+2}, \dots, \varphi_N)^T,
\end{aligned}$$

we can get

$$\begin{aligned}
\hat{\mu}_0 &= \frac{N_1}{N} \hat{\mu}_1 + \frac{N_2}{N} \hat{\mu}_2 = \frac{1}{N} (\varphi_1, \varphi_2, \dots, \varphi_N), \\
||\hat{\mu}_1 - \hat{\mu}_2||^2 &= \frac{\varphi_1^2}{N_1^2} + \frac{\varphi_2^2}{N_1^2} + \dots + \frac{\varphi_{N_1}^2}{N_1^2} + \frac{\varphi_{N_1+1}^2}{N_2^2} + \dots + \frac{\varphi_N^2}{N_2^2} \\
&= \frac{1}{N_1^2} \sum_{i=1}^{N_1} \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=N_1+1}^N \varphi_i^2.
\end{aligned}$$

Thus,

$$\begin{aligned}
tr(\hat{S}_B)tr(S_T) &= tr[((\hat{\mu}_1 - \hat{\mu}_2)(\hat{\mu}_1 - \hat{\mu}_2)^T N_1N_2/N)] \\
&= tr[((\hat{\mu}_1 - \hat{\mu}_2)(\hat{\mu}_1 - \hat{\mu}_2)^T N_1N_2/N)]
\end{aligned}$$

$$\begin{aligned}
&\cdot tr[\sum_{i=1}^N (x_i - \mu_0)(x_i - \mu_0)^T] \\
&= \frac{N_1N_2}{N} \left(\frac{1}{N_1^2} \sum_{i=1}^{N_1} \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=N_1+1}^N \varphi_i^2 \right) \cdot \sum_{i=1}^N ||x_i - \mu_0||^2
\end{aligned} \tag{20}$$

On the other hand,

$$\begin{aligned}
tr(S_B)tr(\hat{S}_T) &= tr[((\mu_1 - \mu_2)(\mu_1 - \mu_2)^T N_1N_2/N)] \\
&\quad \cdot tr[\sum_{i=1}^N (\hat{x}_i - \hat{\mu}_0)(\hat{x}_i - \hat{\mu}_0)^T] \\
&= ||\mu_1 - \mu_2||^2 N_1N_2/N \cdot \sum_{i=1}^N ||\hat{x}_i - \hat{\mu}_0||^2 \\
&= ||\mu_1 - \mu_2||^2 N_1N_2/N \\
&\quad \cdot \left\{ \left[\frac{(N-1)^2}{N^2} \cdot \varphi_1^2 + \frac{1}{N^2} \varphi_2^2 + \dots + \frac{1}{N^2} \varphi_N^2 \right] \right. \\
&\quad \left. + \dots + \left[\frac{1}{N^2} \cdot \varphi_1^2 + \frac{1}{N^2} \varphi_2^2 + \dots + \frac{(N-1)^2}{N^2} \varphi_N^2 \right] \right\} \\
&= ||\mu_1 - \mu_2||^2 N_1N_2/N \cdot \left\{ \frac{(N-1)}{N} \sum_{i=1}^N \varphi_i^2 \right\}
\end{aligned} \tag{21}$$

Thus, combining (20) and (21), when meeting the condition

$$\frac{N}{N-1} \frac{\sum_{i=1}^N ||x_i - \mu_0||^2}{||\mu_1 - \mu_2||^2} \geq \frac{\sum_{i=1}^N \varphi_i^2}{\frac{1}{N_1^2} \sum_{i=1}^{N_1} \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=N_1+1}^N \varphi_i^2},$$

we can obtain $\frac{tr(\hat{S}_B)}{tr(\hat{S}_T)} \geq \frac{tr(S_B)}{tr(S_T)}$.

Lemma 2 Proof: Let $X_1 = W_1 \cup W_2 \dots \cup W_l$ and $X_2 = W_{l+1} \cup W_{l+2} \cup \dots \cup W_K$, where $W_1 = \{x_1, x_2, \dots, x_{m_1}\}$, $W_2 = \{x_{m_1+1}, x_{m_1+2}, \dots, x_{m_1+m_2}\}$, \dots , $W_l = \{x_{m_1+\dots+m_{l-1}+1}, x_{m_1+\dots+m_{l-1}+2}, \dots, x_{m_1+\dots+m_{l-1}+m_l}\}$, \dots , $W_K = \{x_{m_1+\dots+m_{K-1}+1}, x_{m_1+\dots+m_{K-1}+2}, \dots, x_{m_1+\dots+m_{K-1}+m_K}\}$. We can get $m_1 + \dots + m_{l-1} + m_l = N_1$, $m_{l+1} + m_{l+2} + \dots + m_K = N_2$, $X = X_1 \cup X_2 = (W_1 \cup W_2 \dots \cup W_l) \cup (W_{l+1} \cup W_{l+2} \cup \dots \cup W_K)$, where $W_i \cap W_j = \emptyset$ ($i, j = 1, 2, \dots, K$).

The schematic diagrams of subset partitioning and mapping are shown in Fig. 9.

To simplify the problem and facilitate theoretical analysis, for an arbitrary sample $x_k \in W_i$ ($i = 1, 2, \dots, K$), let \hat{x}_k be the output of x_k going through the RBF network, where $\hat{x}_k = (0, \dots, \varphi_i, \dots, 0)$.

Let $\hat{\mu}_1$, $\hat{\mu}_2$, $\hat{\mu}_0$ be the output means of the first-category, second-category and all training samples going through the RBF network, respectively. Thus, we can obtain

$$\begin{aligned}
\hat{\mu}_1 &= \frac{1}{N_1} (m_1\varphi_1, m_2\varphi_2, \dots, m_l\varphi_l, 0, \dots, 0)^T, \\
\hat{\mu}_2 &= \frac{1}{N_2} (0, \dots, 0, m_{l+1}\varphi_{l+1}, m_{l+2}\varphi_{l+2}, \dots, m_K\varphi_K)^T,
\end{aligned}$$

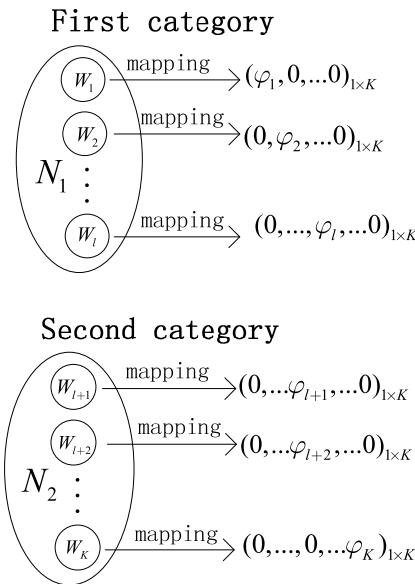


FIGURE 9. The schematic diagrams of subset partitioning and mapping.

$$\hat{\mu}_0 = \frac{1}{N}(m_1\varphi_1, m_2\varphi_2, \dots, m_K\varphi_K),$$

$$\begin{aligned} & \|\hat{\mu}_1 - \hat{\mu}_2\|^2 \\ &= \frac{m_1^2\varphi_1^2}{N_1^2} + \dots + \frac{m_l^2\varphi_l^2}{N_1^2} + \frac{m_{l+1}^2\varphi_{l+1}^2}{N_2^2} + \dots + \frac{m_K^2\varphi_K^2}{N_2^2} \\ &= \frac{1}{N_1^2} \sum_{i=1}^l m_i^2\varphi_i^2 + \frac{1}{N_2^2} \sum_{i=l+1}^K m_i^2\varphi_i^2. \end{aligned}$$

$$\begin{aligned} & \sum_{i=1}^N \|\hat{x}_i - \hat{\mu}_0\|^2 \\ &= \sum_{i=1}^{m_1} \|\hat{x}_i - \hat{\mu}_0\|^2 + \sum_{i=m_1+1}^{m_1+m_2} \|\hat{x}_i - \hat{\mu}_0\|^2 \\ &\quad + \dots + \sum_{i=m_1+\dots+m_{s-1}+1}^K \|\hat{x}_i - \hat{\mu}_0\|^2 \\ &= m_1 \left[\left(\varphi_1 - \frac{m_1\varphi_1}{N} \right)^2 + \frac{m_2^2\varphi_2^2}{N^2} + \dots + \frac{m_K^2\varphi_K^2}{N^2} \right] \\ &\quad + m_2 \left[\frac{m_1^2\varphi_1^2}{N^2} + \left(\varphi_2 - \frac{m_2\varphi_2}{N} \right)^2 + \dots + \frac{m_K^2\varphi_K^2}{N^2} \right] \\ &\quad + \dots + m_K \left[\frac{m_1^2\varphi_1^2}{N^2} + \frac{m_2^2\varphi_2^2}{N^2} + \dots + \left(\varphi_1 - \frac{m_K\varphi_K}{N} \right)^2 \right] \\ &= \frac{1}{N} \sum_{i=1}^K m_i(N - m_i)\varphi_i^2 \end{aligned}$$

On the one hand,

$$\begin{aligned} & \text{tr}(\hat{S}_B)\text{tr}(S_T) = \text{tr}[(\hat{\mu}_1 - \hat{\mu}_2)(\hat{\mu}_1 - \hat{\mu}_2)^T N_1 N_2 / N] \\ &\quad \cdot \text{tr}[\sum_{i=1}^N (x_i - \mu_0)(x_i - \mu_0)^T] \end{aligned}$$

$$\begin{aligned} &= \|\hat{\mu}_1 - \hat{\mu}_2\|^2 N_1 N_2 / N \cdot \sum_{i=1}^N \|x_i - \mu_0\|^2 \\ &= \left(\frac{1}{N_1^2} \sum_{i=1}^l m_i^2 \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=l+1}^K m_i^2 \varphi_i^2 \right) \\ &\quad \times \sum_{i=1}^N \|x_i - \mu_0\|^2 \cdot N_1 N_2 / N \end{aligned} \quad (22)$$

On the other hand,

$$\begin{aligned} & \text{tr}(S_B)\text{tr}(\hat{S}_T) \\ &= \text{tr}[(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T N_1 N_2 / N] \\ &\quad \cdot \text{tr}[\sum_{i=1}^N (\hat{x}_i - \hat{\mu}_0)(\hat{x}_i - \hat{\mu}_0)^T] \\ &= \|\mu_1 - \mu_2\|^2 N_1 N_2 / N \cdot \sum_{i=1}^N \|\hat{x}_i - \hat{\mu}_0\|^2 \\ &= \frac{N_1 N_2}{N^2} \|\mu_1 - \mu_2\|^2 \sum_{i=1}^K m_i(N - m_i)\varphi_i^2 \end{aligned} \quad (23)$$

Thus, combining (22) and (23), when meeting the condition

$$\frac{\sum_{i=1}^N \|x_i - \mu_0\|^2}{\|\mu_1 - \mu_2\|^2} \geq \frac{\frac{1}{N} \sum_{i=1}^K m_i(N - m_i)\varphi_i^2}{\frac{1}{N_1^2} \sum_{i=1}^l m_i^2 \varphi_i^2 + \frac{1}{N_2^2} \sum_{i=l+1}^K m_i^2 \varphi_i^2},$$

we can obtain $\frac{\text{tr}(\hat{S}_B)}{\text{tr}(S_T)} \geq \frac{\text{tr}(S_B)}{\text{tr}(S_T)}$.

REFERENCES

- [1] F. Meng, X. Li, and J. Pei, "A feature point matching based on spatial order constraints bilateral-neighbor vote," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4160–4171, Nov. 2015.
- [2] J. Pei, H. Fan, and L. Pu, "Range space super spherical cap discriminant analysis," *Neurocomputing*, vol. 244, pp. 112–122, Jun. 2017.
- [3] Y. Song, J. Liang, J. Lu, and X.-W. Zhao, "An efficient instance selection algorithm for k nearest neighbor regression," *Neurocomputing*, vol. 251, pp. 26–34, Aug. 2017.
- [4] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy K -nearest neighbor algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. SMC-15, no. 4, pp. 580–585, Jul./Aug. 1985.
- [5] M. Javadian, S. B. Shouraki, and S. S. Kourabbasou, "A novel density-based fuzzy clustering algorithm for low dimensional feature space," *Fuzzy Sets Syst.*, vol. 318, pp. 34–55, Jul. 2017.
- [6] L. Liang-Qun and X. Wei-Xin, "Intuitionistic fuzzy joint probabilistic data association filter and its application to multitarget tracking," *Signal Process.*, vol. 96, pp. 433–444, Mar. 2014.
- [7] G. Lixin, X. Weixin, and P. Jihong, "Segmented minimum noise fraction transformation for efficient feature extraction of hyperspectral images," *Pattern Recognit.*, vol. 48, no. 10, pp. 3216–3226, Oct. 2015.
- [8] X. Liang, L. Zhu, and D.-S. Huang, "Multi-task ranking SVM for image cosegmentation," *Neurocomputing*, vol. 247, pp. 126–136, Jul. 2017.
- [9] V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Netw.*, vol. 17, no. 1, pp. 113–126, Jan. 2004.
- [10] A. Rakotomamonjy, "Variable selection using SVM-based criteria," *J. Mach. Learn. Res.*, vol. 3, nos. 7–8, pp. 1357–1370, Nov. 2003.
- [11] H. Wang, P. Shi, H. Li, and Q. Zhou, "Adaptive neural tracking control for a class of nonlinear systems with dynamic uncertainties," *IEEE Trans. Cybern.*, vol. 99, pp. 1–13, Sep. 2016.

- [12] H. Wang, W. Sun, and P. X. Liu, "Adaptive intelligent control of nonaffine nonlinear time-delay systems with dynamic uncertainties," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1474–1485, Jul. 2017.
- [13] X. Zhao, P. Shi, X. Zheng, and J. Zhang, "Intelligent tracking control for a class of uncertain high-order nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 9, pp. 1976–1982, Sep. 2016.
- [14] X. Zhao, X. Wang, G. Zong, and X. Zheng, "Adaptive neural tracking control for switched high-order stochastic nonlinear systems," *IEEE Trans. Cybern.*, vol. 99, pp. 1–12, Mar. 2017.
- [15] H. Wang, X. Liu, and K. Liu, "Robust adaptive neural tracking control for a class of stochastic nonlinear interconnected systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 510–523, Mar. 2016.
- [16] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2004, pp. 985–999.
- [17] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, May 2006.
- [18] G.-B. Huang and L. Chen, "Convex incremental extreme learning machine," *Neurocomputing*, vol. 70, nos. 16–18, pp. 3056–3062, 2007.
- [19] G.-B. Huang and L. Chen, "Enhanced random search based incremental extreme learning machine," *Neurocomputing*, vol. 71, nos. 16–18, pp. 3460–3468, Oct. 2008.
- [20] G. Feng, G.-B. Huang, Q. Lin, and R. Gay, "Error minimized extreme learning machine with growth of hidden nodes and incremental learning," *IEEE Trans. Neural Netw.*, vol. 20, no. 8, pp. 1352–1357, Aug. 2009.
- [21] Q.-Y. Zhu, A. K. Qin, P. N. Suganthan, and G.-B. Huang, "Evolutionary extreme learning machine," *Pattern Recognit.*, vol. 38, no. 10, pp. 1759–1763, Oct. 2005.
- [22] J. Cao, Z. Lin, and G.-B. Huang, "Self-adaptive evolutionary extreme learning machine," *Neural Process. Lett.*, vol. 36, no. 3, pp. 285–305, Dec. 2012.
- [23] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1411–1423, Nov. 2006.
- [24] W. Zong, G.-B. Huang, and L. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing*, vol. 101, pp. 229–242, Feb. 2013.
- [25] K. Chen, Q. Lv, Y. Lu, and Y. Dou, "Robust regularized extreme learning machine for regression using iteratively reweighted least squares," *Neurocomputing*, vol. 230, pp. 345–358, Mar. 2017.
- [26] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2405–2417, Dec. 2014.
- [27] H. Wen, W. Xie, and J. Pei, "A structure-adaptive hybrid RBF-BP classifier with an optimized learning strategy," *PLoS ONE*, vol. 11, no. 10, p. e0164719, Oct. 2016.
- [28] C. Blake and C. Merz. (1998). "UCI repository of machine learning databases," Dept. Inf. Comput. Sci., Univ. California, Irvine, CA, USA, Tech. Rep. [Online]. Available: <http://archive.ics.uci.edu/ml/>
- [29] L. Yingwei, N. Sundararajan, and P. Saratchandran, "A sequential learning scheme for function approximation using minimal radial basis function neural networks," *Neural Comput.*, vol. 9, no. 2, pp. 461–478, Feb. 1997.
- [30] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Comput.*, vol. 1, no. 2, pp. 281–294, Jun. 1989.
- [31] G.-B. Huang, P. Saratchandran, and N. Sundararajan, "An efficient sequential learning algorithm for growing and pruning RBF (GAP-RBF) networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 6, pp. 2284–2292, Dec. 2004.
- [32] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Trans. Electron. Comput.*, vol. EC-14, no. 3, pp. 326–334, Jun. 1965.
- [33] C.-C. Chang and C.-J. Lin. (2016). "LIBSVM: A library for support vector machines," Dept. Comput. Sci. Inf. Eng., National Taiwan Univ., Taipei, Taiwan, Tech. Rep. [Online]. Available: <http://www.csie.ntu.edu.tw/cjlin/libsvm/S>



HUI WEN was born in 1981. He is currently pursuing the Ph.D. degree with the College of Information Engineering, Shenzhen University, Shenzhen, China. His current research interests include machine learning and neural networks.



HONGGUANG FAN was born in 1983. He is currently pursuing the Ph.D. degree with the College of Information Engineering, Shenzhen University, Shenzhen, China. His current research interests include signal analysis, pattern recognition, and control theory.



WEIXIN XIE was born in 1941. He received the degree from Xidian University, Xi'an. He was a Faculty Member with Xidian University in 1965. From 1981 to 1983, he was a Visiting Scholar with the University of Pennsylvania, USA. In 1989, he was invited to the University of Pennsylvania as a Visiting Professor. He is currently a Professor with Shenzhen University, where he is also the Director of the National Natural Science Foundation of China with Electronic discipline and the Academic Committee. His current research interests include intelligent information processing, fuzzy information processing, image processing, and pattern recognition.



JIHONG PEI was born in 1966. He received the B.S. degree in electronics engineering from Beihang University in 1989, and the M.S. degree in physical electronics and optoelectronics and the Ph.D. degree in signal and information processing from Xidian University, in 1994 and 1998, respectively. He is currently a Professor with Shenzhen University, where he is also the Director of the Electronics Engineering Department, CIE. His research interests include intelligent information processing and video and image analysis.

• • •