Optimal SVM learning method based on adaptive sparse sampling and granularity shift factor

Hui Wen[1]*, Dongshun Jia[2], Zhiqiang Liu[1], Hang Xu[1] and Guangtao Hao[1]
[1] New engineering industry college, Putian University, Putian, China
[e-mail: wen_hui81@163.com, 77700332@qq.com, hangxu520@hotmail.com, sduhgt@126.com]
[2] Department of Liaohe Geophysical Prospecting, Bureau of Geophysical Prospecting INC, Panjin, China
[e-mail: dongshunjia@126.com]
*Corresponding author: Hui Wen

Received August 10, 2021; revised October 22, 2021; revised March 4, 2022; accepted April 7, 2022; published April 30, 2022

Abstract

To improve the training efficiency and generalization performance of a support vector machine (SVM) in a large-scale set, an optimal SVM learning method based on adaptive sparse sampling and the granularity shift factor is presented. The proposed method combines sampling optimization with learner optimization. First, an adaptive sparse sampling method based on the potential function density clustering is designed to adaptively obtain sparse sampling samples, which can achieve a reduction in the training sample set and effectively approximate the spatial structure distribution of the original sample set. A granularity shift factor method is then constructed to optimize the SVM decision hyperplane, which fully considers the neighborhood information of each granularity region in the sparse sampling set. Experiments on an artificial dataset and three benchmark datasets show that the proposed method can achieve a relatively higher training efficiency, as well as ensure a good generalization performance of the learner. Finally, the effectiveness of the proposed method is verified.

Keywords: Support vector machine, sparse sampling, granularity, granularity shift factor, large scale set

This research was supported by the Natural Science Foundation of Fujian Province (Nos. 2019J01815, 2019J01816, 2020J05213, 2020J01917 and 2020H0047), the Industry-University Cooperation and Education Projects of the Ministry of Education under Grant (202101127002, 202102015028), Putian science and technology project (2021G3001-1) and the Department of Education Planning of Fujian Province (FJJKCG20-101).

http://doi.org/10.3837/tiis.2022.04.002 **ISSN : 1976-7277**

1. Introduction

With the progress of science and technology and the improvement of human management level and knowledge level, a large number of data in the real world need to be processed. How to deal with massive data mining has become a research hotspot in the field of machine learning in recent years. As a typical kernel machine learning method, support vector machine (SVM) [1] has been widely applied to many fields such as industrial diagnosis [2], medical detection [3], image recognition [4], vehicle communication [5], time series prediction [6], et al. By using kernel technique and edge maximization criterion, SVM can build optimal decision surface, which has unique advantages in solving small sample, nonlinear and high-dimensional pattern recognition. However, the essence of SVM training process is a convex quadratic optimization (QP) problem. Given the number of training samples is N, the training time complexity is $O(N^3)$ and the space complexity is at least $O(N^2)$, which results in a time-consuming training process under a large-scale data mining problems.

Many methods have been explored to improve SVM training and classification in large scale data sets. The typical methods include chunking algorithm [1], decomposition algorithm [7], sequential minimum optimization (SMO) algorithm [8], parallel SVM method [9,10], however, these methods are highly dependent on the selected working sets, and the partition of different working sets has a great impact on the generalization performance of SVM. Random down sampling based SVM method [11], clustering based SVM method [12,13] can reduce the size of the training sample set, however, the selected samples often can not reflect the spatial distribution characteristics of the original sample set, and the generalization performance of SVM may be reduced.

Granular SVM (GSVM) [14-16] is another typical method to optimize the training efficiency of SVM. In the existing GSVM methods, the number of initial granules needs to be determined in advance, which often leads to the selected granules can not fully reflect the distribution structure of the sample space. From a model perspective, how to mine the spatial structure features of sample set to construct an appropriate granular structure is worthy of further study [17].

Among the above research methods, the essence of improving SVM training efficiency can be regarded as by reducing the size of training set to reduce the complexity of solving problems. However, these optimization methods often take the cost of generalization performance of SVM, and different training set reduction methods lead to different effects on the performance of SVM. Additionally, the optimization of the selected training set and hyperplane in most methods needs many iterations, which increases the training burden. For large scale data classification problems, improving the training efficiency and ensuring the generalization performance of SVM are often a pair of contradictions.

To effectively improve the performance of SVM under large scale set, based on the above research, a SVM learning method based on adaptive sparse sampling and granularity shift factor is presented. The main motivation of our work is to improve the training efficiency of SVM and effectively ensure the generalization performance of SVM. To achieve this goal, the proposed method effectively combines the sampling optimization with SVM hyperplane optimization. The goal of sampling optimization is to greatly reduce the size of the training set and effectively approximate the spatial distribution structure of the original training set. Then, the hyperplane optimization stage aims to overcome the influence of optimal hyperplane offset

caused by sampling sparsity. In this way, the training efficiency of SVM can be greatly improved, and the generalization performance of SVM can be guaranteed.

To realize the effective reduction of the original training set, in the process of sampling optimization, the potential values of samples in different areas of training sample space are measured by potential function density clustering, which can effectively utilize the global distribution information of the sample space, and establish Gaussian kernel with different parameters to complete the effective coverage of different areas of the training sample space. Each coverage generates a sampling sample incrementally, and these sampling samples come from different local areas of the original training sample space. In this way, sampling samples can be adaptively determined according to the distribution of the original training sample space, which overcomes the distortion of the spatial structure caused by insufficient samples in the method of random sampling SVM. Compared with the clustering based SVM method, the proposed method can overcome the problem that the number of clusters needs to be adjusted manually and the scale of subspace coverage is inconsistent. Compared with existing GSVM methods, the proposed method can approximate the specific structure of training sample set to construct a granular structure. Additionally, the extraction of sampling samples only needs one scan of the original training sample set, and the training time of SVM can be further reduced.

Although the proposed adaptive sampling optimization method has the above advantages, compared with the SVM decision surface directly trained by the original sample set, the sparsity of sampling samples may cause the SVM classification boundary to deviate to a certain extent. To solve this problem, in the stage of hyperplane optimization, the local area information of the sampling samples is further considered. By measuring the density of each sampling sample region and the mixing degree with heterogeneous samples, a granularity shift factor is defined and different granularity shift factors are calculated for different granules. In this way, a new convex quadratic optimization is constructed.

To verify the characteristics of the proposed method, the proposed method is compared with other methods on an artificial data set and several benchmark data sets. Experimental results show that the proposed method can effectively improve the training efficiency of SVM and ensure good generalization performance in the classification of large scale data sets.

The contribution of this paper can be summarized as follows:

1) A new sparse sampling method based on potential function density clustering is presented, which can generate sparse sampling set adaptively and approximate the spatial distribution structure of the original training sample space.

2) A new granularity shift factor method is constructed and measured by the density of each sampling sample region and the mixing degree with heterogeneous samples. It fully considers the neighborhood information of the sampling samples, and effectively overcomes the influence of decision hyperplane offset caused by sparse sampling.

3) The adaptive sparse sampling set and granularity shift factor are combined to optimize the SVM decision hyperplane, which can achieve relatively higher training efficiency, as well as ensure the good generalization performance of the learner.

2. Related work

To improve the training efficiency of SVM, Vapnik et al. [1] proposed a chunking algorithm. By decomposing the large-scale QP problem to eliminate non-support vectors one by one, the storage requirement in the training process is reduced. However, when the number of support vectors is large, the amount of chunking data will also increase, which affects the training efficiency of the algorithm. Osuna et al. [7] proposed a decomposition based SVM algorithm,

which decomposes the QP problem into several smaller scale QP problems by iteratively selecting the working set. The working set selected directly affects the convergence performance of the algorithm. Sequential minimum optimization algorithm (SMO) [8] selects only two samples at a time, and uses heuristic method with a two nested loop to find the samples to be optimized, however, the computational cost is too high in judging the optimal conditions. In [9,10], the original sample space is divided into different subsets and then combined with the parallel SVM algorithm. However, the partition of different subsets will still have different effects on the generalization performance of SVM. Different from the above methods, the down sampling based SVM method reduces the size of training samples by extracting or clustering representative samples from the original sample set to improve the training efficiency of SVM. Random down sampling based SVM method [11] and clustering based SVM method [12,13] are two typical down sampling SVM methods. The disadvantage of the random down sampling based SVM method is the obtained sampling samples often can not reflect the spatial distribution characteristics of the original sample set. The clustering based SVM method takes the clustering center of the training samples as the new training set of SVM, where the number of clustering needs to be determined in advance. Although it can greatly reduce the size of the training sample set, these clustering centers often change the spatial structure distribution of the original data set, and the generalization performance of SVM will be affected.

To improve the performance of traditional SVM, Tang [14] proposed a granular SVM (GSVM) method, which integrates granular computing theory and SVM optimization methods. GSVM first establishes a series of information granules in the original sample space, then extracts some important information samples from the divided granules for learning, and finally fuses the different important information obtained from different granules to get the final classifier. GSVM has a significant improvement in the training efficiency of SVM, however, the divided granules may lead to the difference of data distribution and reduce the generalization performance of SVM. To improve the generalization performance of GSVM, the GSVM based on mixed measure [15] maps original samples into the high-dimensional space by mercer kernel, and then divides these samples into different granules, and those mixed granules are extracted and trained by SVM. The GSVM based on hierarchy tree [16] divides the data into some granules by means of hierarchical and dynamic granulation. According to the density and radius of the granules, the granules closest to the hyperplane are dynamically extracted and re-granulated at the subtle level. Other methods [17-27] of optimizing SVM are designed to make SVM more suitable for specific problems.

3. Method

3.1 Principle of SVM algorithm

For the classification problem, the essence of SVM is to find an optimal hyperplane as the decision surface under the given training sample set, where the isolated edge between positive and negative examples is maximized. Given the training set $\{x_i, d_i\}_{i=1}^N$, d_i is the pattern category label of x_i, the hyperplane is optimized by the following optimization problem

$$\begin{aligned} &\max Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j x_i^T x_j \\ &s.t. \sum_{i=1}^{N} \alpha_i d_j = 0, 0 \le \alpha_i \le C, i = 1, 2, ..., N \end{aligned} \tag{1}$$

where α_i is the Lagrange multiplier and C is the selected positive parameter. When $\alpha_i \neq 0$, the corresponding training samples are support vectors. The above optimization problem can be extended by nonlinear mapping of input samples to high-dimensional feature space, and the following kernel trick is given as

$$K(x_i, x) = \phi(x_i)^T \cdot \phi(x) \tag{2}$$

The corresponding decision boundary of SVM can be expressed as

$$f(x) = sign\left[\sum_{i=1}^{N_s} \alpha_i d_i K(x, x_i) + b\right] \tag{3}$$

where N_s is the number of support vectors.

3.2 Adaptive sparse sampling method based on potential function density clustering

The potential function [28] reflects the influence degree of two vectors in space changing with distance. Let x and y denote two vectors of the pattern space respectively, and $p(x, y)$ denote the potential function established by the two vectors. According to the description of reference [28], a common potential function model is given as follows:

$$p(x, y) = \frac{1}{1 + Td^2(x, y)} \tag{4}$$

where T is a constant; $d(x, y)$ represents the distance between x and y.

According to the definition of potential function, the mathematical model of potential function is introduced into the training sample space, and the learning mechanism of potential function density clustering is designed to automatically extract the sampling samples. Given the training set $B = \{(x_i, d_i)\}_{i=1}^{N}$, where d_i is the pattern category label related to the sample $\mathbf{x}$ in B, $d_i \in R^h$, h is the number of pattern categories. Let B_i be the set of eigenvectors labeled d_i, $B_i = \{\mathbf{x}_1^i, \mathbf{x}_2^i, ..., \mathbf{x}_{N_i}^i\}$, where N_i is the number of training samples in the i-th pattern categories, $\sum_{i=1}^{h} N_i = N$. Here $B = \cup_{i=1}^{h} B_i$, $B_i \cap B_j = \varnothing, \forall i \neq j$. For a pair of sample $(\mathbf{x}_u^i, \mathbf{x}_v^i)$ in B_i, the interaction potential between $\mathbf{x}_u^i$ and $\mathbf{x}_v^i$ is

$$p(\mathbf{x}_u^i, \mathbf{x}_v^i) = \frac{1}{1 + T \cdot d^2(\mathbf{x}_u^i, \mathbf{x}_v^i)}, \quad u, v = 1, 2, ...N_i, \; u \neq v \tag{5}$$

Taking $\mathbf{x}_v^i$ as the benchmark sample, the interaction potential of all other samples to $\mathbf{x}_v^i$ in the i-th pattern categories is accumulated, which can be expressed as

$$\kappa(\mathbf{x}_v^i) = \sum_{u=1, u \neq v}^{N_i} p(\mathbf{x}_u^i, \mathbf{x}_v^i) \tag{6}$$

Thus, the potential value set κ^i in B_i can be obtained, here $\kappa^i = \{\kappa(\mathbf{x}_1^i), \kappa(\mathbf{x}_2^i), ..., \kappa(\mathbf{x}_{N_i}^i)\}$. To cover the sample space effectively, the largest potential in κ^i can be chosen and the corresponding sample $\mathbf{x}_p^i$ can be determined, where $\kappa(\mathbf{x}_p^i) = \max\{\kappa(\mathbf{x}_1^i), \kappa(\mathbf{x}_2^i), ..., \kappa(\mathbf{x}_{N_i}^i)\}$. The sample $\mathbf{x}_p^i$ can be incorporated into the sparse sampling set. When the initial width is given, a

corresponding Gaussian kernel can be established with the center $\mathbf{x}_p^i$ and the given kernel width to cover a local area in the original sample space. Additionally, it is necessary to eliminate the sample potential value of the current coverage area as far as possible, so as to find a new round of maximum potential value and the corresponding sampling sample. The potential updating process is given as follows:

$$\kappa'(\mathbf{x}_v^i) = \kappa(\mathbf{x}_v^i) - \kappa(\mathbf{x}_p^i) \cdot \exp(-\frac{1}{2\sigma_k^2} \| \mathbf{x}_v^i - \mathbf{x}_p^i \|^2), \quad v = 1, 2, ...N_i, v \neq p \tag{7}$$

where $\kappa'(\mathbf{x}_v^i)$ is the updated sample potential value in B_i, and σ_k is the selected kernel width. Here σ_k represents the coverage scale of different local areas in the original training sample space, which can be taken a fixed value in the experimental stage. Equation (7) shows that when an arbitrary sample $\mathbf{x}_v^i$ in B_i is closer to the center, its potential value is more offset, when it is farther from the center sample, its potential value is less offset due to the attenuation of Gaussian kernel function. When the following inequality is satisfied:

$$\max\{\kappa'(\mathbf{x}_1^i), \kappa'(\mathbf{x}_2^i), ..., \kappa'(\mathbf{x}_{N_i}^i)\} > \delta \tag{8}$$

the proposed method turns to find the next representative sampling sample, where δ is the threshold. In this way, the effective coverage of the sample space is completed step by step. Otherwise, the method of adaptive sparse sampling turns to other pattern categories until all pattern categories are learned, and finally the sparse sampling set can be constructed.

Combined with the above description, the adaptive sparse sampling method based on potential function clustering is given in **Table 1**.

Table 1. Adaptive sparse sampling method based on potential function clustering

Initialize the number of sampling samples $k = 0$, the sparse sampling set is $B' = \{\}$. Set the initial Gauss kernel width σ and the parameter *T*. Given the training set $B = \bigcup_{i=1}^{h} B_i$, $B_i = \{\mathbf{x}_1^i, \mathbf{x}_2^i, ..., \mathbf{x}_{N_i}^i\}$.

For each pattern category B_i, do
1. Use (5) and (6) to calculate the potential value of each sample.
2. Find the sample $\mathbf{x}_p^i$ with the maximum potential value.
3. The sample $\mathbf{x}_p^i$ is incorporated into the sparse sampling set, where $B' \leftarrow B' \cup \{\mathbf{x}_p^i\}$.
4. Use (7) to update each sample potential value in B_i.
5. Set iteration termination conditions

If $\max\{\kappa'(\mathbf{x}_1^i), \kappa'(\mathbf{x}_2^i), ..., \kappa'(\mathbf{x}_{N_i}^i)\} > \delta$

Go to step 2

else

The learning process of B_i is over. Turn to learn other pattern categories until all pattern categories are learned.

EndIf

3.3 Method of optimizing hyperplane by using granularity shift factor

In the above adaptive sparse sampling method, the sampling samples are directly extracted from different local areas of the original sample space, which can effectively approximate the structure distribution of the original sample space, thus ensuring the generalization

performance of the learning model. However, compared with the SVM decision surface directly trained from the original sample set, the sparsity of the sampling set may cause the SVM classification boundary to deviate to a certain extent. To overcome this deficiency, in this section, a method of granularity shift factor is designed to correct the hyperplane of SVM.

In the process of establishing the granularity shift factor, it is necessary to adjust the areas covered by different Gaussian kernels in the potential function density clustering to establish an appropriate granularity size set. Set that the sampling set is $B^{'}$, where sampling samples are extracted from each pattern category by the method of potential function density clustering. After merging these sampling samples, the total number of sampling samples obtained is *M*. Then, the sampling space is divided into *M* granules, which can be expressed as $X=\{X_1, X_2, ..., X_M\}$. Here $X_i=\left\{x_{i_p}\right\}_{p=1}^{n_i}$, n_i is the number of samples contained in the *i*-th granularity, $\sum_{i=1}^{M} n_i = N$. Each granularity can be regarded as a super ball, then the optimized grain center μ_i and radius r_i are $\mu_i=\frac{1}{n_i}\sum_{p=1}^{n_i} x_{i_p}$, $r_i=\max_{x\in X_i}(x_i-\mu_i)$, respectively.

To effectively mine the information contained in each granularity, the information entropy [29] of granularity is introduced in this work. Let the number of positive and negative samples contained X_i is n_i^+ and n_i^-, respectively, $n_i^+ + n_i^- = n_i$. The information entropy of X_i is

$$E(X_i)=-\frac{n_i^+}{n_i}\log_2\frac{n_i^+}{n_i}-\frac{n_i^-}{n_i}\log_2\frac{n_i^-}{n_i} \tag{9}$$

here $0\le E(X_i)\le 1$.

On the basis of granularity information entropy, the purity of X_i can be measured, which is inversely proportional to the entropy of X_i. Here a negative exponential function is introduced to express the granularity purity, which is expressed as

$$P(X_i)=e^{-E(X_i)} \tag{10}$$

Generally, when the purity of X_i is relatively high, X_i is relatively easy to be divided, then the decision hyperplane can be appropriately far away from X_i. On the contrary, when the purity of X_i is smaller, the proportion of mixed heterogeneous samples in X_i is relatively large, which indicates that the area of X_i located is closer to the real decision hyperplane, and the decision hyperplane should be appropriately close to X_i.

The density of X_i can be expressed as

$$\rho(X_i)=\frac{n_i}{N}\cdot\frac{1}{n_i}\sum_{p=1}^{n_i}\exp\left(-\frac{\|x_{i_p}-\mu_i\|^2}{2r_i^2}\right)=\frac{1}{N}\sum_{p=1}^{n_i}\exp\left(-\frac{\|x_{i_p}-\mu_i\|^2}{2r_i^2}\right) \tag{11}$$

If the density of X_i is higher, there will be more samples distributed in the vicinity of this region in the testing set. To improve the generalization performance, the decision hyperplane should be appropriately far away from X_i.

In combination with granularity purity and granularity density, the granularity shift factor ϑ_i of X_i was defined as

$$\vartheta_i = P(X_i)\cdot\rho(X_i) \tag{12}$$

here $0 \le \vartheta_i < 1$.

The improved soft-interval SVM quadratic optimization problem can be expressed as

$$\begin{aligned} &\min_{\omega,b} \ \frac{1}{2}\| \omega \|^2 + C\sum_{i=1}^{M} \vartheta_i \xi_i \\ &s.t. \ \ d_i(\omega^T \phi(x_i) + b) \ge 1 - \vartheta_i - \xi_i, \ i = 1,2,...,M, \xi_i \ge 0. \end{aligned} \tag{13}$$

where ξ_i is the relaxation variable and C is the penalty factor. The duality problem is expressed as

$$\begin{aligned} &\max Q(\alpha) = \sum_{i=1}^{M} (\alpha_i - \alpha_i \vartheta_i) - -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i \alpha_j d_i d_j K(x_i, x_j) \\ &s.t. \sum_{i=1}^{M} \alpha_i d_j = 0, 0 \le \alpha_i \le \tau_i C, i = 1,2,...,M. \end{aligned} \tag{14}$$

To illustrate the characteristics of this method, **Fig. 1** shows the process of generating sparse samples by potential function clustering. This process can greatly reduce the size of training samples. On this basis, the neighborhood information of each sampled sample is further considered to obtain the optimized hyperplane. The SVM optimized method combining adaptive sparse sampling with granularity shift factor is given in **Table 2**.

3.4 Computational complexity analysis

In this study, the potential density clustering method is used to construct the sparse sampling set, then, a method of granularity shift factor is used to correct the hyperplane of SVM. The computational complexity is analyzed as follows:

(1) Set the number of initial training set be N, and the number of B' obtained by down sampling be M. In the process of incremental construction of sparse sampling set, the label information of each category of samples is considered, and the calculation of sample potential value needs to traverse all other samples in the current pattern category. Here, the initial training sample set is set to contain two pattern categories, and the number of samples in each pattern category is N_1 and N_2, respectively, $N_1+N_2=N$. The complexity of calculating sample potential value $O\left((N_1-1)^2+(N_2-1)^2\right)$. On this basis, Gaussian kernels with different parameters are established to cover the training sample space, and the computational complexity of the sample potential updating process is $O(M)$. Combined with the calculation of sample space potential and the process of potential updating, the computational complexity is $O\left(N^2 - 2N_1N_2 - 2N + M\right)$.

(2) In the process of calculating the granularity shift factor, the number of samples covered by each granularity needs to be counted, which needs to traverse all training samples, and the computational complexity is $O(MN)$. Then, the sparse sampling set B' is used for SVM training, the computational complexity is $O\left(M^3\right)$.

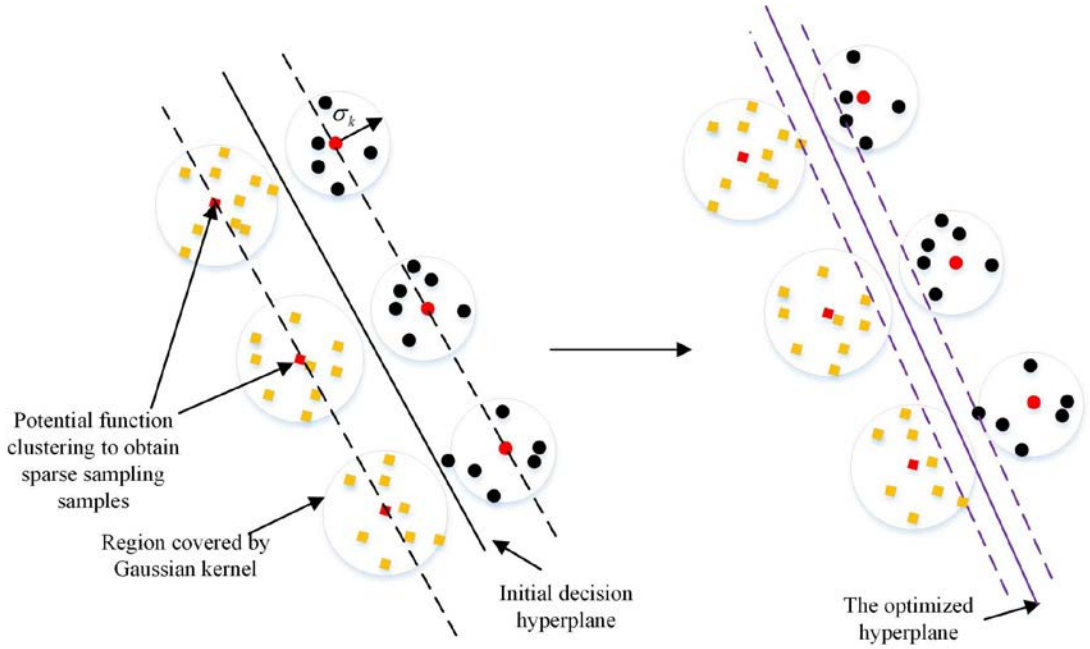

Fig. 1. Schematic diagram of sparse sampling and optimization of SVM hyperplane

Table 2. SVM based on adaptive sparse sampling with granularity shift factor

Initialize: Given the training set $B=\{(x_i,d_i)\}_{i=1}^{N}$, set the kernel function of SVM and initial parameters.

1. Use the adaptive sparse sampling algorithm in **Table 1** to extract samples from the original training set, and the number of sampling samples obtained is *M*.
2. Optimize the areas covered by different Gaussian kernels in **Table 1** to obtain a series of information granules $\{X_1, X_2, ..., X_M\}$, where $X_i=\left\{x_{i_p}\right\}_{p=1}^{n_i}$, n_i is the number of samples contained in X_i. The optimized center μ_i and radius r_i in X_i are $\mu_i=\frac{1}{n_i}\sum_{p=1}^{n_i}x_{i_p}$, $r_i=\max_{x\in X_i}(x_i-\mu_i)$, respectively.
3. Calculate the purity $P(X_i)$ and the density $\rho(X_i)$ of each information granularity according to (9) - (11), then use (12) to calculate the granularity shift factor.
4. Construct and solve the SVM optimization problem according to (14), and the optimal solution $\boldsymbol{\alpha}^*$ is obtained as $\boldsymbol{\alpha}^*=(\alpha_1^*,\alpha_2^*,\ldots,\alpha_M^*)$. Find the non-zero value of each component in $\boldsymbol{\alpha}^*$ and record it as $\alpha_{0,i}$, here $i=1,2,...,N_s$, N_s is the number of support vectors. Calculate the optimal weight vector $\omega^*=\sum_{i=1}^{N_s}\alpha_{0,i}d_i\phi(x_i)$, and select a component $\alpha_{0,m}$ from $\alpha_{0,i}$ to calculate $b^*=d_m-\frac{\tau_m}{d_m}\sum_{i=1}^{N_s}d_i\alpha_{0,m}d_iK(x_i,x_m)$.
5. For an arbitrary sample x, the decision hyperplane is constructed as $f:\omega^{*T}\cdot\phi(x)+b^*=0$, optimal decision function is obtained $f(x)=\mathrm{sgn}(\sum_{i=1}^{N_s}\alpha_{0,i}d_iK(x_i,x)+b^*)$ can be obtained.

Combined with the above analysis, the computational complexity of the proposed method is $O\left(N^2-2N_1N_2-2N+M+MN+M^3\right)$. For large scale set, the computational complexity of using the original sample set to train SVM directly is $O\left(N^3\right)$. Considering $M<<N$, compared with the method of training SVM directly using the original sample set, the training efficiency of the proposed method can be greatly improved.

4. Experimental comparison and analysis

In this section, the performance of the presented method is evaluated using an artificial dataset, namely the Double moon [30] and three benchmark datasets [31] from University of California, Irvine (UCI): Credit, Occupancy and Record. The performance of the proposed method is compared with a LIBSVM algorithm [32], a random sampling SVM (RM_SVM) [11], a clustering SVM (C_SVM) [12], GSVM [14], Granular support vector machine based on mixed measure (M_GSVM) [15] and SVM based on hierarchical and dynamical granulation (HD_GSVM) [16], respectively. All data samples in each dataset are scaled to [−1, 1], the parameter of distance weighting factor is set as T=1, the kernel width parameter σ of potential function clustering is set between 0.1 and 0.7, the learning threshold of potential function is set as δ=0.001. In each SVM optimization method, the selected kernel function is the radial basis function, and the kernel width γ of SVM is taken from the set [0.5, 1, 2], the penalty parameter is set to C=1000. For RM_SVM, C_SVM and GSVM, the number of down sampling samples is given in advance; For M_GSVM and HD_GSVM, the number of initial granules is given first, and then other granules are extracted adaptively and iteratively. For the presented method, the number of sampling samples is generated adaptively. The operating environment of the experiment was an Intel (R) core i7-9700, 3.00 GHz CPU, 8 G RAM, and MATLAB 2013. Each experiment was repeated 10 times. **Table 3** provides a description of the classification datasets.

Table 3. Information description of different classification datasets

Datasets	Number of classes	Number of features	Number of training samples	Number of testing samples
Double Moon	2	2	500-2000	4000
Credit	2	23	15000	20000
Occupancy	2	5	10808	9752
Record	2	7	20000	104913

4.1 Double Moon classification problem

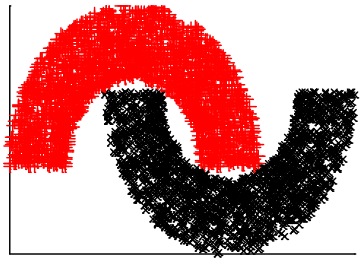

Fig. 2. Double moon classification dataset

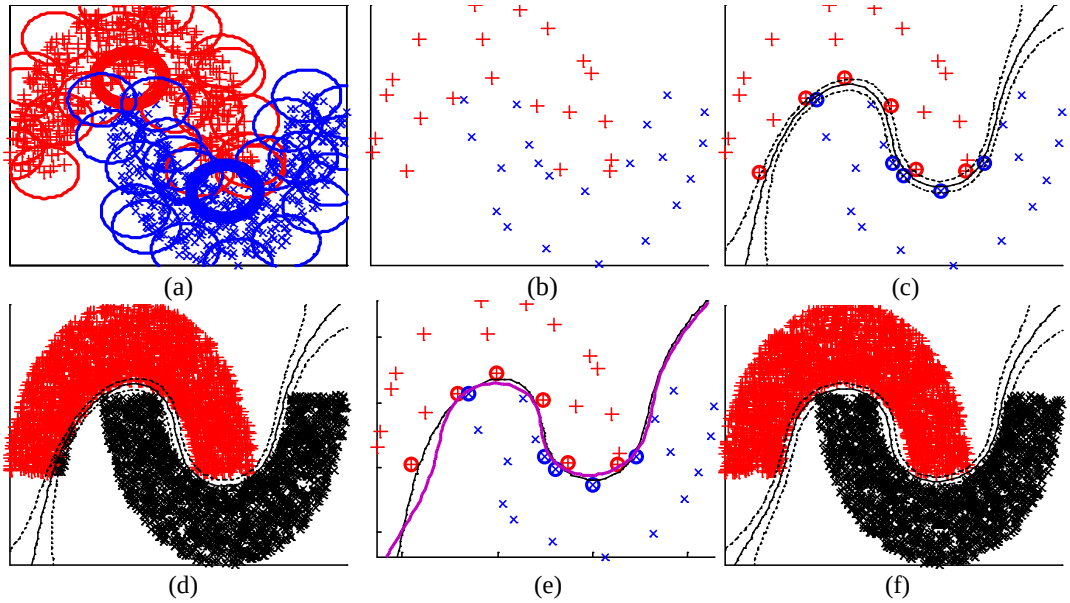

Fig. 3. The learning and classification effects of the proposed method. (a) Covering training sample space by the method of potential function clustering (b) All the center samples are extracted as the adaptive sparse sampling set (c) The generated sparse sample set is used as the training of SVM to optimize the hyperplane. (d) The hyperplane is used to classify testing samples (e) Comparison of the effect on training set before and after using granularity shift factor (f) Classification effect of the corrected hyperplane on the testing set

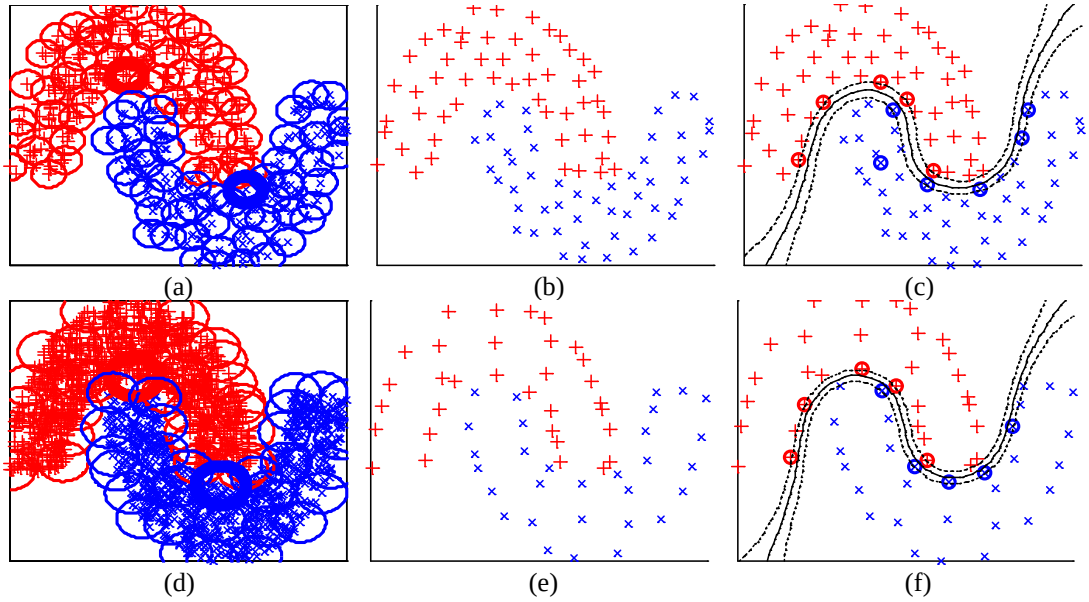

Fig. 4. Effect of adaptive sparse sampling when the number of training set and σ change（a）Potential function clustering is used to cover the training sample space (N=500, $\sigma = 0.1$) (b) Clustering centers are taken as the sparse sampling set (M=102)（c）SVM hyperplane is optimized by the sparse sampling set (M=102)（d）Potential function clustering is used to cover the training sample space (N=2000, $\sigma = 0.15$) (e) Clustering centers are taken as the sparse sampling set (M=65)（f）SVM hyperplane is optimized by the sparse sampling set (M=65)

Table 4. Performance comparisons of different methods on Double moon dataset (N=2000)

Methods	No. of times to scan the training set	(C, γ)	No. of sampling samples	Training time/s	Testing accuracy/%
LIBSVM	1	(1000,0.5)	--	329.89	100
RM_SVM	1	(1000,0.5)	500	20.75	98.57
C_SVM	1	(1000,0.5)	100	16.83	99.86
GSVM	1	(1000,0.5)	100	16.34	99.72
M_GSVM	2	(1000,0.5)	82	13.63	99.83
HD_SVM	2	(1000,0.5)	76	13.42	99.91
Proposed sampling($\sigma = 0.1$)	1	(1000,0.5)	102	17.28	99.83
Proposed[a]					
$\sigma = 0.1$	1	(1000,0.5)	102	18.21	100
$\sigma = 0.15$	1	(1000,0.5)	65	10.65	100
$\sigma = 0.2$	1	(1000,0.5)	41	8.52	100

[a] The combination of adaptive sparse sampling with hyperplane optimization.
[b] The number of initial granules is given manually, then other granules are extracted adaptively.

In this section, an artificial dataset, namely Double moon is used to evaluate the performance of the proposed method graphically. **Fig. 2** shows a graphical representation of the Double moon classification dataset. **Fig. 3** shows the learning and classification effects of the proposed method on the Double moon dataset. The proposed adaptive sparse sampling method achieves the extraction of different sampling samples by optimizing the coverage of different regions of the sample space, so as to generate an optimized sparse sampling set, which can greatly reduce the number of training samples and achieve effective approximation of the spatial structure features of sample set. However, the sparsity of the sampling set leads to the deviation of the hyperplane to a certain extent, this deficiency can be overcome by constructing the granularity shift factor, where the information of different local areas in the original sample set is taken into account, the decision hyperplane of SVM can be furtheroptimized. From the classification effect on the testing set, the generalization performance of SVM is improved.

Fig. 4 shows the effect of adaptive sparse sampling when the number of training set and kernel width σ change. Although the adaptive sparse sampling set generated also changes accordingly, it can still effectively approximate the structural distribution of sample space, so as to obtain the optimized hyperplane of SVM. This process shows that the proposed adaptive sparse sampling method has good adaptability to the sample space. Thus, the effectiveness of the proposed method is verified.

Table 4 shows the performance comparisons of different methods on Double moon dataset. When the kernel width σ change, the proposed sparse sampling method can generate different number of samples sampling samples adaptively. When σ is set too small, the number of sparse sampling samples generated is relatively large, and the training time of SVM is slightly higher than other optimized SVM methods. However, when σ is in a certain range, the training efficiency of the proposed method outperforms other optimized SVM methods. Additionally, the method combining adaptive sparse sampling with granularity shift factor improves the generalization performance of the learner.

4.2 UCI benchmark classification problems

Under the UCI benchmark datasets, the performance comparisons of the proposed method and other methods are shown in **Tables 5-7**. Compared with LIBSVM, the training time of the proposed method is greatly reduced, and the classification accuracy of the proposed method is equivalent to that of LIBSVM. Compared with RM_SVM, GSVM, M_GSVM and HD_SVM, the training time and classification accuracy of the proposed method are significantly improved. Under the Credit data set, the training efficiency of the proposed method is significantly better than those of other methods, and the testing accuracy of the proposed method is about 0.8%-4.8% higher than those of other methods. Under the Occupancy data set, the training efficiency of the proposed method is better than RM_SVM, GSVM, M_GSVM and HD_SVM, and comparable to that of C_SVM, however, the testing accuracy of the

Table 5. Performance comparisons of different methods on Credit dataset

Methods	No. of times to scan the training set	(C, γ)	No. of sampling samples	Training time/s	Testing accuracy/%
LIBSVM	1	(1000,1)	--	2695.61	82.52
RM_SVM	1	(1000,1)	1500	184.57	77.64
C_SVM	1	(1000,1)	600	54.52	79.72
GSVM	1	(1000,1)	700	59.93	80.57
M_GSVM	4	(1000,1)	469	62.17	81.13
HD_SVM	4	(1000,1)	426	52.42	81.68
Proposed	1	(1000,1)	382	43.63	82.43

Table 6. Performance comparisons of different methods on Occupancy dataset

Methods	No. of times to scan the training set	(C, γ)	No. of sampling samples	Training time/s	Testing accuracy/%
LIBSVM	1	(1000,1)	--	1776.64	80.74
RM_SVM	1	(1000,1)	2162	359.38	76.92
C_SVM	1	(1000,0.5)	500	35.54	78.52
GSVM	1	(1000,0.5)	600	43.48	78.68
M_GSVM	5	(1000, 0.5)	521	46.26	79.29
HD_SVM	5	(1000, 0.5)	576	49.53	79.41
Proposed	1	(1000, 0.5)	359	35.87	80.58

Table 7. Performance comparisons of different methods on Record dataset

Methods	No. of times to scan the training set	(C, γ)	No. of sampling samples	Training time/s	Testing accuracy/%
LIBSVM	1	(1000,2)	--	3842.65	97.64
RM_SVM	1	(1000,2)	2000	348.53	92.91
C_SVM	1	(1000,1)	400	42.67	95.24
GSVM	1	(1000,1)	600	53.31	95.09
M_GSVM	4	(1000,1)	428	53.89	97.32
HD_SVM	4	(1000,1)	416	64.25	97.28
Proposed	1	(1000,1)	258	42.37	97.41

proposed method is about 2% higher than that of C_SVM, and about 1.1%-3.6% higher than those of other methods to varying degrees. Under the Record data set, the training efficiency of the proposed method is better than those of other methods, and the generalization performance of the proposed method is improved to varying degrees.

Different from other methods, the sparse sampling samples of the proposed method are generated automatically according to the spatial distribution of the original data set. Compared with the original training set, the sample size of the sparse sampling set is greatly reduced, thus, the training efficiency of the proposed method can be greatly improved. Additionally, the granularity shift factor is constructed for the optimization of the SVM decision hyperplane, where the neighborhood information of each granularity in the sparse sampling set is further considered. The generalization performance of SVM can be guaranteed effectively.

4.3 Discussion

4.3.1 Influence of the width σ on the presented method

In this work, the kernel width parameter σ of the potential function cluster determines the coverage scale of the training sample space, and then affects the size of the generated sparse sampling set. Therefore, the kernel width parameter σ is discussed here. **Tables 8-10** show that by adjusting the value of σ, the generated adaptive sparse sampling set also changes, but the overall classification performance of the proposed method is relatively stable, which shows that the proposed method has good adaptability to the sample space. However, when σ is too small, the coverage scale of the proposed method to the sample space is too low, which leads to the large scale of the sampling set. For example, in **Table 7**, when σ=0.2, the training efficiency of the proposed method decreases sharply due to too large sparse sampling set. Especially when σ=0.1, the scale of the sampling sample set is the same as that of the original training sample set, so the proposed method is invalid and directly converted to LIBSVM

Table 8. Performance comparison of the proposed method under different σ on Credit dataset

σ	No. of down sampling set	Training time/s	Testing accuracy/%
0.1	15000	--	82.52
0.2	8573	945.41	82.58
0.3	1652	205.84	82.41
0.4	753	76.52	82.47
0.5	382	43.63	82.43
0.6	275	36.58	82.26
0.7	224	31.74	81.84

Table 9. Performance comparison of the proposed method under different σ on Occupancy dataset

σ	No. of down sampling set	Training time/s	Testing accuracy/%
0.1	10800	--	80.74
0.2	1467	157.85	80.61
0.3	715	71.73	80.49
0.4	463	58.52	80.52
0.5	359	35.87	80.58
0.6	226	28.69	80.24
0.7	184	26.36	79.81

Table 10. Performance comparison of the proposed method under different σ on Record dataset

σ	No. of down sampling set	Training time/s	Testing accuracy/%
0.1	20000	--	97.64
0.2	2072	437.64	97.58
0.3	855	109.53	97.49
0.4	534	57.49	97.52
0.5	381	46.18	97.36
0.6	258	42.37	97.41
0.7	215	38.76	97.28

directly. Therefore, in practical application, to ensure the effectiveness, σ should not be too low. This is also the main limiting factor of the proposed method.

4.3.2 The sampling influence of δ value on the presented method

In this paper, the parameter δ is the learning threshold of potential function density clustering, and its value affects the number of sampling samples in each pattern category to a certain extent. To effectively extract sampling samples from different local areas of the sample space, the parameter δ should be set to an appropriate value to make the potential function density clustering algorithm converge, so that the number of sampling samples will remain in a stable range. Here, select the occupancy benchmark data set. **Fig. 5** shows the number of sampling samples generated when δ takes different values. For the convenience of drawing, the value of δ is converted logarithmically. When δ is relatively large, e. g., δ=10, the number of samples generated is relatively small, indicating that the potential function density clustering algorithm can not effectively extract samples from different local areas of the sample space. When the δ value is relatively small, e. g., δ = 0.01, the number of sampling samples remains in a stable range, which means that the potential function density clustering algorithm can achieve convergence. For convenience, the value of δ in each data set experiment is uniformly set as δ = 0.001.

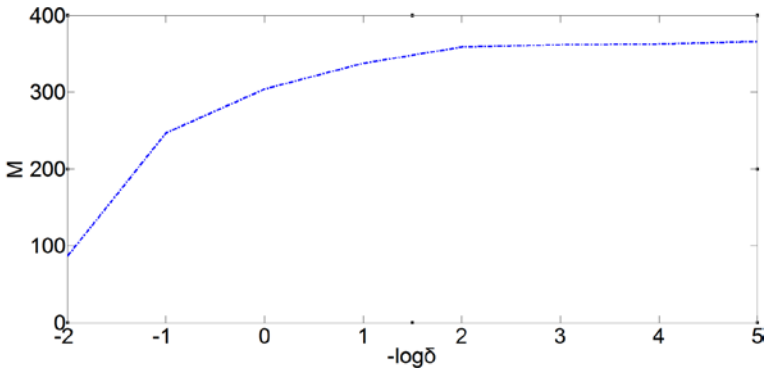

Fig. 5. Influence of δ value on the number of sampling samples

5. Conclusion

An optimal SVM learning method combining adaptive sparse sampling and the granularity shift factor for large-scale sample sets was presented. The method generalizes the SVM to include nonlinear problems in a novel manner. By applying the potential function density clustering method, sparse sampling samples can be obtained adaptively in each local area of the sample space. By applying the granularity shift factor method, the decision hyperplane of the SVM can be corrected in an optimal manner. The presented method only requires one scan of the original training sample set, which makes it attractive for large-scale classification problems. Experiments on an artificial dataset and three benchmark datasets show that the proposed method can obtain good classification results with efficient training efficiency.

In this study, the granularity shift factor method was used to mine the local area information of the samples. For the problem of an unbalanced data classification, the combination of a global imbalance and local imbalance of different categories of samples can be used as a future research objective to solve the problem of an unbalanced classification. In addition, the presented method focuses only on binary classification problems. In view of the complexity and diversity of classification problems in practice, multi-class classification, sequence sample learning, and semi-supervised learning problems are areas of future research.

References

[1] V. N. VAPNIK, “An overview of statistical learning theory,” *IEEE Transactions on neural networks*, vol. 10, no. 5, pp. 988-999, Sept. 1999. Article (CrossRef Link)

[2] X Lv, H Wang, X Zhang, et al., “An evolutional SVM method based on incremental algorithm and simulated indicator diagrams for fault diagnosis in sucker rod pumping systems,” *Journal of Petroleum Science and Engineering*, vol. 203, 2021. Article (CrossRef Link)

[3] Q. H. Nguyen, B. P. Nguyen, T. B. Nguyen, et al., “Stacking segment-based CNN with SVM for recognition of atrial fibrillation from single-lead ECG recordings,” *Biomedical Signal Processing and Control*, vol. 68, no.10, July. 2021. Article (CrossRef Link)

[4] J. X. Bian, B. J. Ma, A. Paul, et al., “Research on electrochemical discharge machining based on image features and SVM algorithm,” *Journal of Intelligent & Fuzzy Systems*, vol. 40, no.4, pp. 7247-7258, 2021. Article (CrossRef Link)

[5] D Li, J Zhu, H Zhao, et al., “SVM-based online learning for interference-aware multi-cell mmWave vehicular communications,” *IET Communications*, vol. 15, no. 8, pp. 1015-1027, March. 2021. Article (CrossRef Link).

[6] J Ruan, X Wang, Y Shi, “Developing fast predictors for large-scale time series using fuzzy granular support vector machines,” *Applied Soft Computing Journal*, vol. 13, no. 9, pp. 3981–4000, 2013. Article (CrossRef Link)

[7] E. Osuna, R. Freund, F. Girosi, “Training support vector machines: an application to face detection,” in *Proc. of the IEEE conference on computer vision and pattern recognition*, Los Alamitos, Puerto Rico: IEEE Computer Society, pp.130−136, 1997. Article (CrossRef Link)

[8] J. Platt, *Advances in Kernel Methods-Support Vector Learning*, Cambridge, Mass., USA: MIT Press, 1998.

[9] Z You, J Yu, L Zhu, et al., “A MapReduce based parallel SVM for large-scale predicting protein interactions,” *Neurocomputing*, vol. 145, pp. 37-43, Dec. 2014. Article (CrossRef Link)

[10] W Guo, N. K. Alham, Y Liu, et al., “A Resource Aware MapReduce Based Parallel SVM for Large Scale Image Classifications,” *Neural Processing Letters*, vol. 44, no.1, pp.161-184, 2016. Article (CrossRef Link)

[11] J. Balczar, Y. Dai, O. Watanabe, “A random sampling technique for training support vector machines,” in *Proc. of the 12th International Conference on Algorithmic Learning Theory*, Berlin, Germany: Springer-Verlag, pp. 119–134, 2001.

[12] Y Wang, X Zhang, S Wang, et al., "Nonlinear clustering-based support vector," *Optimization Methods and Software*, vol. 23, no. 4, pp. 533-549, 2008. Article (CrossRef Link)
[13] A. Mozafari, M. Jamzad, "Cluster-based adaptive SVM: A latent subdomains discovery method for domain adaptation problems," *Computer Vision and Image Understanding*, vol. 162, no. 1, pp. 116-134, SEP. 2017. Article (CrossRef Link)
[14] Y Tang. B Jin, Y Zhang, "Granular support vector machines with association rules mining for protein homology prediction," *Artificial Intelligence in Medicine*, vol. 35, no.1-2, pp. 121-134, 2005. Article (CrossRef Link)
[15] H Guo, W Wang, "Support vector machine based on hierarchical and dynamical granulation," *Neurocomputing*, vol. 211, pp. 22–33, 2016. Article (CrossRef Link).
[16] W Wang, H Guo, Y Jia, "Granular support vector machine based on mixed measure," *Neurocomputing*, vol. 101, pp. 116-128, 2013. Article (CrossRef Link).
[17] H Guo, W Wang, "Granular support vector machine: a review," *Artifitial Intelligence Review*, vol. 51, no. 1, pp. 19-32, 2019. Article (CrossRef Link)
[18] Zhou S, "Sparse SVM for Sufficient Data Reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. Article (CrossRef Link)
[19] S Ding, H Huang, J Yu, et al., "Research on the hybrid models of granular computing and support vector machine," *Artifitial Intelligence Review*, vol. 43, no. 6, pp. 565–577, 2015. Article (CrossRef Link)
[20] S Lu, Y Chen, X Zhu, et al., "Exploring Support Vector Machines for Big Data Analyses," in *Proc. of 2021 4th International Conference on Computer Science and Software Engineering (CSSE 2021)*, Chengdu, China, pp.42-48, 2021.
[21] Z Liu, D. Elashoff, S. Piantadosi, "Sparse support vector machines with l_0 approximation for ultra-high dimensional omics data," *Artificial intelligence in medicine*, vol. 96, pp. 134-141, 2019. Article (CrossRef Link)
[22] J Yin, Q Li, "A semismooth Newton method for support vector classification and regression," *Computational Optimization and Applications*, vol. 73, no. 2, pp. 477-508, 2019. Article (CrossRef Link)
[23] Y Yan, Q Li, "An efficient augmented Lagrangian method for support vector machine," *Optimization Methods and Software*, vol. 35, no. 4, pp. 855-883, 2020. Article (CrossRef Link)
[24] P. Konstantinos, T. Ioannis, D. George, et al., "A Support Vector Machine model for classification of efficiency: An application to M&A," *Research in International Business and Finance*, vol. 61, 2022. Article (CrossRef Link)
[25] P. D'Urso, J. M. Leski, "Fuzzy c-ordered medoids clustering for interval-valued data," *Pattern Recognition*, vol. 58, pp. 49–67, 2016. Article (CrossRef Link).
[26] G Guo, J Zhang, "Reducing examples to accelerate support vector regression," *Pattern Recognition Letters*, vol. 28, no. 16, pp. 2173–2183, 2007. Article (CrossRef Link)
[27] J Li, Y Tan, A Zhang, "The Application of Internet Big Data and Support Vector Machine in Risk Warning," in *Proc. of Journal of Physics: Conference Series*, vol. 1952, no. 4, pp. 1-11, 2021. Article (CrossRef Link)
[28] O. A. Bashkerov, E. M. Braverman, I.E. Muchnik, "Potential function algorithms for pattern recognition learning machines," *Automatic Remote Control*, vol. 25, no. 5, pp. 692-695, Jan. 1964.
[29] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379- 423, July. 1948. Article (CrossRef Link)
[30] S. HAYIN, *Neural networks and learning machines*, 3rd ed. Beijing: China Machine Press, 2009.
[31] C. BLAKE, C. MERZ, "UCI repository of machine learning databases," 2018. [Online]. Available: http://archive.ics.uci.edu/ml/datasets.php
[32] C. CHANG, C. LIN, "LIBSVM: a library for support vector machines," 2016. [Online]. Available: https://www.csie.ntu.edu.tw/~cjlin/libsvm/

Hui Wen was born in 1981, he received the Ph.D. degree in intelligent information processing from Shenzhen University, China, in 2018. He is currently an associate professor in New Engineering Industry College, Putian University. His current research interests include machine learning and neural networks.

Dongshun Jia was born in 1981, he received the M.S. degree in signal and information processing from Yangtze University, China, in 2007. His research interests include seismic exploration, seismic signal processing and pattern recognition.

Zhiqiang Liu was born in 1983, he received the Ph.D. degree in intelligent information processing from Communication University of China, China, in 2019. He is currently a lecturer in Institute of Electromechanical and Information Engineering, Putian University. His current research interests include machine learning, remote sensing image processing.

Hang Xu was born in 1990, he received the Ph.D. degree in computer science and technology in Xiamen University, China, in 2018. He is currently an associate professor in Institute of Electromechanical and Information Engineering, Putian University. His research interests include pattern recognition, multi objective optimization and machine learning.

Guangtao Hao was born in 1980, he received the Ph.D. degree in power system and its automation in Shandong University, China, in 2018. He is currently an associate professor in Institute of Electromechanical and Information Engineering, Putian University. His research interests include power system analysis and control, pattern recognition.