Neurocomputing 211 (2016) 22–33

Contents lists available at ScienceDirect

Neurocomputing

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

Support vector machine based on hierarchical and dynamical granulation

Husheng Guo [a,b], Wenjian Wang [a,b,*]

[a] School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China
[b] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan 030006, Shanxi, China

ARTICLE INFO

Article history:
Received 16 July 2015
Received in revised form
25 September 2015
Accepted 16 October 2015
Available online 7 June 2016

Keywords:
Granular support vector machine
Hierarchical and dynamical granulation
HD_GSVM model
Density
Radius

ABSTRACT

Support vector machine (SVM) has been a promising method for classification and regression areas due to its solid statistical foundations, such as margin maximization and kernel methods. However, SVM is not typically used for large-scale data mining problems because its training complexity is highly dependent on the dataset size. This paper presents an improved granular support vector machine learning model based on hierarchical and dynamical granulation, namely, HD_GSVM, to solve the low learning efficiency and generalization performance problem of traditional granular support vector machines (GSVM). For HD_GSVM, the original data will be mapped into a high-dimensional space by a Mercer kernel. Then, the data are divided into several granules, and those granules near the approximate hyperplane are extracted and re-granulated on a subtle level by their density and radius degree. Finally, the decision hyperplane will be obtained through all of the granules at different hierarchical and dynamical granulation levels effectively. During the granulation process, the granulation level of all granules can be dynamically changed continuously. With this method, different classification information can be obtained from different levels of granules; to meet a variety of needs for various practical problems from different perspectives. The experimental results on the UCI benchmark datasets demonstrate that the proposed HD_GSVM model can improve the generalization performance greatly with high efficiency synchronously.

1. Introduction

Support vector machine (SVM) introduced by Vapnik [1] is an effective method for solving pattern recognition and regression problems such as handwritten recognition, face image recognition, and time series prediction. At present, SVM has become a research hotspot of machine learning. In the applications of SVM, researchers pay considerable attention to its learning efficiency and generalization performance, and some scholars have already proposed novel approaches to improve them of SVM [2–9]. Although some achievements have been made, unlike traditional pattern recognition and machine learning, real-world data mining applications often involve large numbers of data records. Thus, it is too expensive to perform multiple scans on the entire dataset, and it is also infeasible to place the dataset in memory. Therefore, the studies on how to improve the learning efficiency and generalization performance of SVM by combining it with other artificial intelligence methods still have important theoretical and practical value.

Granular computing is a new concept and computing paradigm in the domain of information processing [10,11]. It covers all of the studies regarding theories, methods, techniques and tools of granulation. In addition, it can be used to process uncertain, fuzzy, incomplete, and large-scale information. The essence of granular computing is to find an approximate solution, which is simple and low-cost, to replace the exact solution by using inaccurate and large-scale information to achieve the tractability, robustness, and low cost to better describe the real world of intelligent systems or intelligent control. In a word, the combination of granular computing with intelligence computing approaches is becoming a hotspot to stablish efficient algorithms for complex problems.

To improve the performance of traditional SVM, granular support vector machine (GSVM), which combines statistical learning theory and granular computing theory, is first proposed by Tang [12]. In general, a GSVM first creates a sequence of information granules in the original data space and then learns on some of

* Corresponding author at: School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China.
E-mail address: wjwang@sxu.edu.cn (W. Wang).

http://dx.doi.org/10.1016/j.neucom.2015.10.136
0925-2312/

Fig. 1. The basic structure diagram of GSVM model.

these granules when necessary. Finally, it aggregates information in these granules at a suitable and abstract level. The basic structure diagram of GSVM is shown in Fig. 1. This method can obtain better generalization for a linear separable classification problem; and also increase "linear separability" for a linear non-separable problem (or even transfer a linear non-separable problem to a totally linear separable problem). Compared with traditional SVM, the training speed of GSVM can be greatly improved, and a satisfactory generalization performance can be obtained as well.

In fact, long before Tang, several other scholars had already proposed a few effective SVM models, which could be regarded as the prototype of GSVM, such as the classical "Chunking Algorithm" [1], "Decomposed Algorithm" [13], "SMO Algorithm" [14], and "LIBSVM Algorithm Library" [15].

Additionally, some scholars have already designed a number of specific GSVM algorithms, such as GSVM models based on clustering. A clustering based GSVM approach divides the original data into a number of granules by combining commonly used clustering methods with certain evaluation rules, and it takes them into classification or regression after choosing granules with more information (such as granules that include more support vectors) [16].

Some GSVM models based on a geometric technique are designed, such as a method based on the distance between samples, and the approximate best hyperplane is proposed by Cheng et al. [17]. It considers two geometric aspects simultaneously: the first is the distance between samples and the approximate best hyperplane, and the second is the distance between the approximate best hyperplane and the obtained hyperplane.

Considering the difference in granulation on the kernel space and the original space, a GSVM model based on a kernel space is proposed by Guo et al. [18], and the rules of granulation on kernel space were given through geometric analysis. However, these approaches may not be effective for some datasets, where the distance between data can not be measured by European distance.

In addition, Tang et al. [19] presented a GSVM model based on particle swarm optimization and it is an intuitive and easy-to-implement algorithm from the swarm intelligence community. This approach is applicable to fault classification and outperforms several previous methods. Pai et al. [20] presented a GSVM model based on fraud warning, which integrates sequential forward selection, SVM and a classification and regression tree, and it can be used to overcome information overload problems. Deb et al. [21] combine artificial neural networks with SVM. By changing the parameters of neural networks, the model can effectively reduce the dataset size and maintain compressed data agreement with the original data in the distribution, but the interpretability of this model is absent. Moreover, other models, such as granular support vector machine based on Association Rule [22], Rough Sets [23] and Decision Trees [24], are also discussed by many scholars.

Presently, some scholars also have summarized the GSVM model study, such as Ding et al. [25] summarized the GSVM model with two classes: the original space GSVM and the feature space GSVM. In this paper, the scholars presented the GSVM models based on original granulation space systematically, and summarized the GSVM model based on kernel space. In another paper, Ding et al. [26] summarized three special GSVM model that Fuzzy SVM, Rough SVM and Quotient Space SVM. In literature [27], some traditional GSVM methods and the kernel GSVM are compared and presented in detailed.

Although these models are based on different granulation methods, such as clustering, neural networks and matrix decomposition, the nature of it is only granulated on the same abstract level, and the classification information may be largely lost. Although they can improve the learning efficiency, they have some losses in generalization performance mainly for the following two reasons: first, after granulation, the data distribution may be different in the original space than in the kernel space. Second, traditional GSVM often take granulation only once before training and take some informational samples (such as the center of granules) in the same abstract granulation level to replace the whole granule when training. Therefore, data distribution errors are inevitable. These two aspects may reduce the generalization ability of GSVM [19].

This paper presents a support vector machine model based on hierarchical and dynamical granulation, which first maps the original data into a high-dimensional space to reveal the features that are implicit in the original sample space. Then, the data are divided into some granules, and those granules near the approximate hyperplane are extracted and re-granulated at the subtle level by their density degree and radius degree. Finally, the decision hyperplane will be obtained through all of the granules on different abstract levels effectively. Compared with traditional GSVM models, the proposed HD_GSVM can largely improve the generalization performance with high learning efficiency simultaneously.

The remainder of this paper is organized as follows. We begin by presenting the traditional SVM model and SVM based on granulation, and we introduce the shortage of traditional SVMs based on the granulation model (Section 2). In Section 3, we describe the HD_GSVM method based on initial granulation, re-granulation and SVM training in three parts. In Section 4, we simulate experiments and discuss the advantages of the HD_GSVM model with high efficiency and high testing accuracy. In Section 5, we conclude this paper and present the shortage of this model.

2. SVM and GSVM overview

Support vector machine (SVM) is a learning system that uses a hypothesis space of linear functions in a high dimensional feature space, trained with a learning algorithm from an optimization theory. This learning strategy, introduced by Vapnik and his co-workers, is a very powerful method that in the few years since its introduction has already outperformed most systems in a wide variety of applications. SVM is based on two ideas: margin maximization and nonlinear classification using kernels. Let us assume a binary classification problem and a training set D comprising l labeled training samples $\{(x_i, y_i)\}_{i=1}^{l}$ with $x_i \in R^n$ and $y_i \in \{-1, 1\}$. The purpose of SVM classification is to obtain a classifier $f: f(x_i) = y_i$ from data to their labels. In a machine learning problem, the complexity of the classification function f greatly influences the performance achieved. Thus, in general, a highly complex function fits training data perfectly, but gives a poor generalization performance on unlabeled data. If the training data are linearly separable in the feature space, we can obtain the decision function:

$$f(x) = w^T\phi(x) + b \tag{1}$$

where w is a weight vector, b is a bias, and $y_i f(x_i) \geq 1$ for l labeled samples. For unlabeled data x, if $f(x) > 0$, the data are classified into Class 1, and if $f(x) < 0$, the data are classified into Class -1.

If the classification problem is not linearly separable in the feature space, the optimal separating hyperplane can be obtained by solving the following soft margin optimization problem:

$$Minimize\ \ Q(w, \xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i \tag{2}$$

$$Subject\ \ to\ \ y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, \cdots, l \tag{3}$$

where C is the regularization parameter that determines the tradeoff of model complexity and classification error, and ξ_i is the slack variable for x_i. The dual optimization problem is:

$$Maximize\ \ Q(a) = \sum_{i=1}^{l} a_i - \frac{1}{2}\sum_{i,j=1}^{l} a_i a_j y_i y_j K(x_i, x_j) \tag{4}$$

$$Subject\ \ to\ \ \sum_{i=1}^{l} y_i a_i = 0, \quad 0 \leq a_i \leq C \tag{5}$$

where $K(x_i, x_j)$ is a kernel function that is given by $K(x_i, x_j) = \phi(x_i)\phi(x_j)$. Any function that satisfies Mercer's theorem [1] can be used as a kernel function such as the Gaussian kernel $K(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2/\left(2\sigma^2\right)\right)$ and the polynomial kernel $K(x_i, x_j) = (x_i{}^T x_j + 1)^p$.

Granular support vector machine (GSVM), which combines statistical learning theory and granular computing theory, is first proposed by Tang. To better explain the HD_GSVM model, we give the generalization performance analysis of the traditional GSVM model first. Suppose k granules are produced after granulation, and a new training set $X^* \subseteq X$ with some samples belonging to k granules is constructed (Suppose there are l^* samples in X^*, $l^* < l$).

Suppose the classifier f^* is obtained on new training set X^*. Clearly, there is a performance difference between f and f^*. For the original dataset X, after granulation, replacement and other operations, the actual training set X^* may no longer follow the distribution $P(x, y)$ but a new distribution $P^*(x, y)$ ($P(x, y)$ is the distribution of X and $P^*(x, y)$ is the distribution of X^*. In X^*, generally, $P^*(x, y)$ is different from $P(x, y)$). The actual training dataset X^* and original dataset X may not meet the conditions of independent and identically distribution, and thus, some training and testing data would not be classified correctly (see Fig. 2).

Therefore, for a GSVM model, the optimal classification hyperplane f^* is only suitable for X^* but not for X. Therefore, the generalization performance of GSVM may be greatly reduced. To reduce the model error of a traditional GSVM model, this paper will focus on improving the generalization performance of GSVM from three aspects: (1) perform the granulation and SVM training in the same space to eliminate inconsistencies. (2) Different granules are granulated on different abstract granulation levels by the density and other factors. If the granule has more classification information, it is granulated at a small granulation level; otherwise, it is granulated at a large granulation level. Using this method, more classification information is obtained with a minor size training set. (3) Classification informational samples obtained from different abstract granulation levels are trained by SVM at the same time. In this way, the distribution difference by data operations may be reduced and the generalization performance will be improved greatly while maintaining a high learning efficiency.

3. Support vector machine based on hierarchical and dynamical granulation

At present, most studies of GSVM concentrate on statistic granulation, but studies on the hierarchical and dynamical granulation method are still absent. This paper improves the SVM based on the granulation algorithm presented by Tang et al. [12]; and a new SVM method based on hierarchical and dynamical granulation to solve the low generalization performance of traditional GSVM models. This method maps the original data into a high-dimensional space to reveal the features that are implicit in the original sample space and divide the data into several granules first. Then, it extracts those granules near the approximate hyperplane as informational granules and re-granulates them dynamically at a subtle level by their density degree and radius degree. Finally, the decision hyperplane will be obtained through all of the granules at different abstract levels effectively.

3.1. Initial granulation based on kernel

Generally, traditional GSVM training is in high-dimensional space and it divides granules and replaces data with granules in low-dimensional space. So the inherent distribution feature after data replacement with granules may not be reflected well in high dimension feature space, and then the prediction function may not be appropriate. This paper adopts a kernel granular support vector machine algorithm, which firstly maps the original data into a high-dimensional space, and then divides granules by some strategies so as to obtain high-dimensional granules. At last, SVM learning is accomplished by them. It can solve the problems with different distributions in low-dimensional space (Original training dataset) and high-dimensional space (Compressed training dataset). Therefore, the granulation of this paper is processed in the high-dimensional kernel feature space by the kernel granulation methods. To improve the generalization performance, the kernel k-means clustering is selected as the granulation method of HD_GSVM, because it can obtain the same radius granules, and the errors generated from the granulation and replacement of granules (that is the radius of superballs) will counteract each other [18].

For a given original training set $X_{0-1} = \{(x_i, y_i)\}_{i=1}^{l_{0-1}}, x_i \in R^n$ and $y_i \in \{-1, 1\}$ are classification labels. After nonlinear mapping Φ, the samples in high-dimensional space R^N are denoted as $X_0 = \{\Phi(x_i), y_i\}_{i=1}^{l}$. Samples are divided into k_1 granules, that is, $X_{1-i} = \{\Phi(x_j)\}_{j=1}^{l_{1-i}}$ (l_{1-i} is the number of data in the ith granule on the first granulation level of the ith granule and X_{1-i} on the first granulation level; and the number 1 of X_{1-i} presents these granules at the first granulation level). Each granule can be regarded as a super ball, and the center and radius are defined as follows.

Definition 1. (Center and Radius of a Granule Super Ball X_i) For a general granule super ball X_i, each N dimensional granule is called a granule super ball after granulation (For simplicity, the ith granule super ball corresponding to the ith granular is still denoted as X_i). The center μ_i and radius r_i of the ith granule super ball are defined as follows.

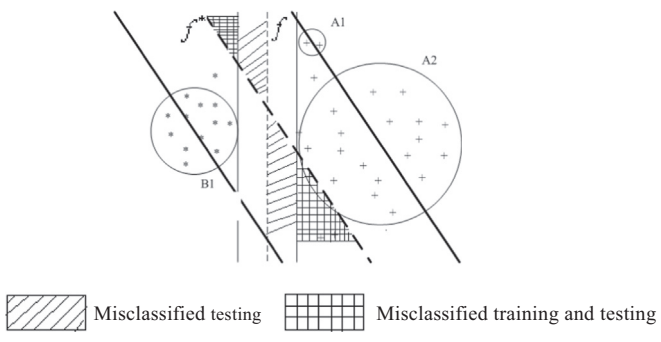

Fig. 2. Misclassified data in several regions.

$$\mu_i = \frac{1}{l_i}\sum_{p=1}^{l_i}\Phi(x_p) = \sqrt{\frac{1}{l_i^2}\left(\sum_{p=1}^{l_i}\Phi(x_p)\right)^2}$$
$$= \frac{1}{l_i}\sqrt{\sum_{p=1}^{l_i}\sum_{q=1}^{l_i}\Phi(x_p)\cdot\Phi(x_q)} = \frac{1}{l_i}\sqrt{\sum_{p=1}^{l_i}\sum_{q=1}^{l_i}K(x_p, x_q)} \quad (6)$$

$$r_i = \max_{x_s\in X_i}(\|\Phi(x_s) - \mu_i\|) = \max_{x_s\in X_i}\left(\sqrt{(\Phi(x_s))^2 - 2\Phi(x_s)\cdot\mu_i + {\mu_i}^2}\right)$$
$$= \max_{x_s\in X_i}\left(\sqrt{K(x_s, x_s) - \frac{2}{l_i}\sum_{p=1}^{l_i}K(x_s, x_p) + \frac{1}{l_i^2}\sum_{p=1}^{l_i}\sum_{q=1}^{l_i}K(x_p, x_q)}\right) \quad (7)$$

According to Definition 1, the distance from any sample x_j to the ith granule super ball X_i in N dimensional space is

$$d(x_j, X_i) = \sqrt{K(x_j, x_j) - \frac{2}{l_i}\sum_{p=1}^{l_i}K(x_j\cdot x_p) + \frac{1}{l_i^2}\sum_{p=1}^{l_i}\sum_{q=1}^{l_i}K(x_p, x_q)} \quad (8)$$

where the granulation is accomplished iteratively by using granule super balls and related measurements. The main steps of the granule dividing algorithm are summarized as follows.

Algorithm 1. Initial Granule dividing algorithm.

Step1: *Select k_1 samples randomly as the center of k_1 granules.*

Step2: *Classify samples according to formula (8) by nearest neighbor approach in kernel space.*

Step3: *Adjust the centers of k_1 granules by formula (6), and observe whether there are changes in these centers. If so, go back to step 2. Else, go to step 4.*

Step4:*End the algorithm and obtain the divided granules* $\{X_{1-1}, X_{1-2}, \ldots, X_{1-k_1}\}$.

3.2. Hierarchical and dynamical granulation

For granules $\{X_{1-i}\}_{i=1}^{k_1}$, $\{\mu_{1-i}\}_{i=1}^{k_1}$ and $\{r_{1-i}\}_{i=1}^{k_1}$ are the corresponding centers and radiuses of them. The centers $\{\mu_{1-i}\}_{i=1}^{k_1}$ are used as the actual training set of GSVM. The approximate hyperplane f_1 can be obtained by SVM training on this actual training set. However, for the difference of $\{\mu_{1-i}\}_{i=1}^{k_1}$ and X_0, the generalization performance loss of the classifier is large and the results of classification are poor. To solve this problem, the obtained hyperplane will be corrected by granulation for some granules by the distance of the samples and the approximate hyperplane f_1 and the density of these informational granules. To better present the HD_GSVM model, we give the Definition of the distance from a granule to the hyperplane.

Definition 2. (***Distance from a Granule to the Hyperplane***) For a general granule X_i, in N dimensional space, the distance from a granule X_i to the general hyperplane f: $y = w\cdot\Phi(x) + b$ is defined as.

$$d(X_i, f) = \frac{(w, -1)\cdot{\mu_i}^T + b}{\sqrt{w^2 + 1}} - r_i$$
$$= \frac{\frac{1}{l_i}\sum_{k=1}^{l_i}\sum_{j\in SVs}\alpha_j y_j K(x_j, x_k) + b}{\sqrt{\sum_{k\in SVs}^{j\in SVs}\alpha_j\cdot\alpha_k\cdot y_j\cdot y_k\cdot K(x_j, x_k)}} - r_i \quad (9)$$

where *SVs* is the set of support vectors. For the SVM, the distance between the support vector and classification hyperplane is equal to one. Then, the Definition of the information granule and the granule density is given based on Definition 2.

Definition 3. (***Informational granule extraction***) For the general granule X_i and general hyperplane f, if the distance $d(X_i, f)$ between a granule X_i and approximate hyperplane f is smaller than one, the granule X_i is called the informational granule.

Because the informational granules are samples for the GSVM model and because the distance between the *SVs* and the approximate hyperplane is equal to one, the informational granules overlap with the margin area of the approximate classifier f.

Definition 4. (***Granule density***) For a general granule X_i, suppose $X_i = \{x_i^j\}_{j=1}^{l_i}$ is the ith granule and μ_i is the center of X_i. The density ρ_i of granule X_i is defined as follows.

$$\rho_i = \frac{l_i}{\sum_{j=1}^{l_i} d(\mu_i, x_i^j)}$$
$$= \frac{l_i}{\sum_{j=1}^{l_i}\sqrt{\frac{1}{l_i}\sum_{p=1}^{l_i}\sum_{q=1}^{l_i}K(x_p, x_q) - \frac{2}{l_i}\sum_{p=1}^{l_i}K(x_p, x_i^j) + K(x_i^j, x_i^j)}} \quad (10)$$

The HD_GSVM method takes the initial granulation method on training dataset X and obtains the first level granules $\{X_{1-i}\}_{i=1}^{k_1}$ by using Algorithm 1. The centers and radiuses of them are $\{\mu_{1-i}\}_{i=1}^{k_1}$ and $\{r_{1-i}\}_{i=1}^{k_1}$, and the initial approximate hyperplane f^1 on the first granulation level is obtained on the approximate training set $\{\mu_{1-i}\}_{i=1}^{k_1}$ on the first granulation level. Second, we extract the informational granules $\{X_{1-j'}\}_{j=1}^{k_1'}\left(\{X_{1-j'}\}_{j=1}^{k_1'} \subseteq \{X_{1-i}\}_{i=1}^{k_1}\right)$, and the centers and radiuses corresponding to these informational granules are $\{\mu_{1-j'}\}_{j=1}^{k_1'}$ and $\{r_{1-j'}\}_{j=1}^{k_1'}$. Then, we compute the density $\{density_{1-j'}\}_{j=1}^{k_1'}$ of these informational granules $\{X_{1-j'}\}_{j=1}^{k_1'}$. For the jth informational granule $X_{1-j'}$ on the first granulation level, we divide it with the following formula.

$$k_{1-j} = \left\lceil \frac{r_{1-j'}\times density_{1-j'}}{d\_para} \right\rceil \quad (11)$$

d_para is the parameter granulation. When the *density* of a granule is large, we can think this granule has more classification information. So, we need re-granulation to this granule, and we mine the detailed classification information of it. Similarly, if the radius of a granule is large and if the center of it is used to replace the whole granule training set, more generalization performance loss may be produced, and we need re-granulation to this granule and we mine the detailed classification information of it. Suppose the jth informational granule $X_{1-j'}$ is the ith granule of all original granule set $\{X_{1-i}\}_{i=1}^{k_1}$ granules. Similarly, for the other informational granules on the first granulation level, we divide them by Algorithm 1 and we can obtain the granules set $\{X_{2-i}\}_{i=1}^{k_2}$ on the second granulation level (Certainly, $\{X_{2-i}\}_{i=1}^{k_2}$ contains granules from the first level that are not informational granules. At the same time, several old informational granules are replaced by new subgranules by dividing). If the ith granule of the first granulation level is not an informational granule, the corresponding $k_{2-i} = 1$ does not need dividing. Obviously, $k_2 = k_1 - k_1' + \sum_{j=1}^{k_1'} k_{1-j}$. Then, the centers and radiuses of the second granulation level granules are $\{\mu_{1-i}\}_{i=1}^{k_1}$ and $\{r_{1-i}\}_{i=1}^{k_1}$ can be computed by formulas (6) and (7), and the initial approximate hyperplane f_2 on the second granulation level is obtained on the approximate training set $\{\mu_{2-i}\}_{i=1}^{k_2}$ on the second granulation level. This procedure is processed until

there are no informational samples that need re-granulation, and the last classifier f_{last} is obtained by our HD_GSVM method. The feature of the HD_GSVM method is that the information extraction and granulation process is hierarchical and dynamical processing. Thus, a granule far from the present approximate hyperplane f_i is not an information granule on this granulation level, but this granule may transfer into an informational granule on the next granulation level because the approximate classifiers of different granulation levels are different.

3.3. Hierarchical and dynamical granulation SVM algorithm

The HD_GSVM method maps the original data into a high-dimensional space to reveal the features that are implicit in the original sample space and that divide the data into several granules first. Then, it extracts those granules near the approximate hyperplane as informational granules and re-granulates them dynamically at a subtle level by their density and radius degree. Finally, the decision hyperplane will be obtained through all of the granules on different abstract levels effectively. The HD_GSVM method can improve the generalization performance greatly with high learning efficiency synchronously.

SVM based on hierarchical and dynamical granulation contains three steps (see Fig. 3): the first step is initial granulation based on kernel space and initial SVM training on the first granulation level. The second step is hierarchical and dynamical re-granulation iteratively and is the most important step of this process. In this step, the informational granules are extracted by the distance of every granule to present an approximate hyperplane. Then, we compute the density of these informational granules to evaluate the classification information contained in them. By the density and radius of these granules, the granule number in the re-granulation process can be obtained, and we can obtain the smaller granules on the next granulation level. In this process, the information granule is dynamically changed, if a granule is not informational on this granulation; however, it changed into an informational granule on the next granulation level because the hyperplane is improved in this iterative process. By this method, for the high density training samples, the most important classification information can be obtained in this process by the distance of them and the approximate hyperplane, density and radius. At the same time, for the low density training samples, granules that are not important for classification can be deleted in this process and can improve the learning efficiency of the algorithm. At the last step, we train SVM on different granulation levels and obtain the last classifier. We only extract abstract (Large) granules to unimportant samples, but for important samples, we extract special (Small) granules. Therefore, the HD_GSVM model can solve the complex data mining problems when the densities of actual datasets are non-uniform and improve the generalization performance greatly with high learning efficiency synchronously.

Because this paper focuses on designing the GSVM model with high efficiency and good generalization performance in the given kernel space, selecting suitable kernel functions and parameters will be not discussed (they are described in Refs. [28–35]). In fact, the proposed HD_GSVM method can be combined with the existing kernel selection approaches. The Gaussian kernel is used in this paper, and the SVM based on hierarchical and dynamical granulation is as follows.

Algorithm 2. Improved SVM model based on hierarchical and dynamical granulation.

Initialize: Suppose the given initial training set is $X_{0\_1} = \{(x_i, y_i)\}_{i=1}^{l_{0\_1}}$, $x_i \in R^n$, *and* $y_i \in \{-1, 1\}$, *initial granulation number parameter* k_1, *Kernel function: RBF Kernel, the granulation level* $lev = 0$, *the granulation parameter is* d_para.

Step1: Initial granulation and SVM training.

Granulate initial training set X_{0-1} *and obtain* k_1 *granules* $\{X_{1-1}, X_{1-2}, \ldots, X_{1-k_1}\}$*based on kernel granulation method of Algorithm 1,* $X_{1-i} = \{\Phi(x_j)\}_{j=1}^{l_{1-i}}$. *Then, the granulation level parameter is* $lev = 1$. *In initial granulation process, we can obtain the centers and radiuses of these granules by formulas (6) and (7), respectively.*

Step2: Hierarchical and Dynamical re-granulation.

Step2.1: Training SVM on the compressing samples (Such as centers of granules$\{X_{lev\_1}, X_{lev\_2}, \ldots, X_{lev\_k_{lev}}\}$*) of this lev granulation level, and obtain the approximate hyperplane* f_{lev}.

Step2.2: Compute the distance between every granule of this granulation level and the approximate hyperplane f_{lev} *by formula (9), and select the informational granules set* $\{X_{lev\_j'}\}_{j=1}^{k_{lev'}}$ *by Definition 3.*

Step2.3: Compute the density and radius of these informational granules and compute the granulation number parameterk_{lev_j} *of the re-granulation process by the radius and density of this granulation level.* k_{lev_j} *is presented as the* j*th granule on the lev granulation level and needs to re-granulate into the number of sub-granules.*

$$k_{lev\_j} = \left\lceil \frac{r_{lev\_j'} \times density_{lev\_j'}}{d\_para} \right\rceil$$

Step2.4: For every granule of the lev granulation level, if it is an informational granule, it is divided into k_{lev_j} *sub-granules similar to Algorithm 1; else, if it is not need re-granulated, then* $lev \leftarrow lev + 1$.

Step2.5: If there are new granules produced in Step2.4, we compute the centers and radiuses of these new granules by using formulas (6) and (7), respectively. Then, we go to Step 2.1; else, we go to Step 3.

Step3: Last training.

Train SVM on the last compressing samples (Such as centers of granules$\{X_{last\_1}, X_{last\_2}, \ldots, X_{last\_k_{last}}\}$*) of this last granulation step, and obtain the last hyperplane* f_{last}.

Step4: Algorithm end.

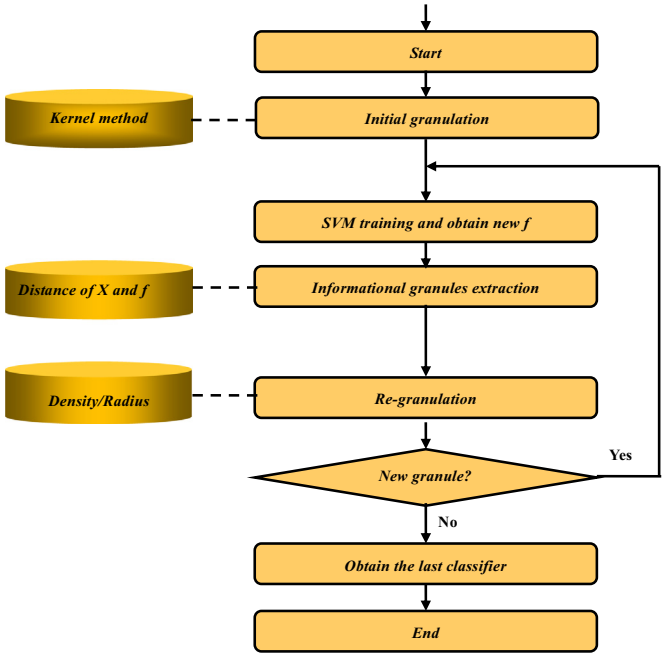

Fig. 3. SVM based on hierarchical and dynamical granulation.

Table 1
Datasets used in experiments.

Datasets	Size of training data	Size of testing data	Features
Thyroid	2800	1500	5
Diabetic	4680	3000	8
Breast_cancer	2000	770	9
Flare_solar	33300	20000	9
Titanic	3000	10255	3

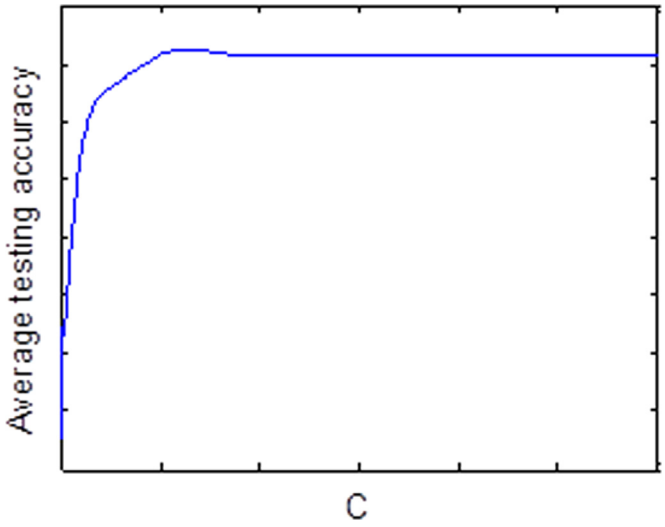

Fig. 4. The testing accuracy variation tendency versus penalty parameter.

4. Simulation experiments and discussions

The granulation process plays an important role in the GSVM model. It determines the granule number obtained from the original training set and the learning efficiency of the algorithm. At the same time, it determines the classification information obtained in this process and the generalization performance of the algorithm. If the granule number obtained in this process is too large, the factual training set may still be large and the learning efficiency of the algorithm may be too low. However, if it is too small, very important classification information may be lost in this process, so the generalization performance may be affected. So, designing a suitable granulation method is important. This paper tests the granule number by using the hierarchical and dynamical granulation of the HD_GSVM model and compares it with the total number granules of the traditional GSVM model.

The comparisons between the hierarchical and dynamical granulation of the HD_GSVM model and the statistic granulation of the traditional GSVM model for generalization performance and learning efficiency are accomplished by simulation experiments, and the influence of model parameters on generalization performance is also studied. To test the former indexes of these two methods, this paper selects five labeled binary class datasets with different sizes and dimensions from the UCI benchmark database [36]. Every dataset is randomly divided into two parts: training set and testing set. To make the experiment results more convincing, all of the datasets are dividing two times randomly and training two times, respectively. The training and testing data are shown in Table 1. The Gaussian kernel function is selected as the kernel function of the GSVM and HD_GSVM methods, and the parameters for them are 1.0. In experiment, we find that when kernel parameter takes a fixed value, the testing accuracy tendency on most datasets are as following Fig. 4. So to make the learning process stable, a large value should be set up for penalty parameter, and we set 200 for it in this paper. The experiment tests are run on a PC (2.66 Ghz CPU, 1G RAM), and the experiment platform is Matlab2008.

4.1. Comparison of granulation

To test the hierarchical and dynamical granulation process of the HD_GSVM model, this paper compares the statistic granulation of the traditional GSVM method based on clustering with the dynamical granulation of the HD_GSVM model. We also observe the granules by using the hierarchical and dynamical granulation method on every granulation level. For the statistic granulation method, the granulation number parameters directly determine the granulation circumstance, so we set the statistic granulation parameter with 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 for datasets *Thyroid, Diabetic, Breast_cancer* and *Flare_solar*, respectively, and with 5, 10, 15, 20, 25, 30 for dataset *Titanic*. For the HD_GSVM method, the initial granulation number parameter and the granulation parameter itself determine the granulation circumstance, and we set the initial granulation number parameter similar to the traditional statistical GSVM method. Moreover, to simplify the experiments, the granulation parameter *d_para* takes the statistical one in these experiments. In practice, we can set a suitable granulation parameter *d_para* by the special circumstance of the problems. When *d_para* is small, the iteration of granulation procedures is quick but the performance of model may be unstable. Conversely, the optimal value is obtained stably with more times iteration training procedures.

Fig. 5 represents the granules' number changing on a different granulation level based on a different initial granulation number parameter. In these figures, the horizontal axis represents the granulation level, and the vertical axis represents the granule number obtained on the corresponding granulation level.

We can observe from Fig. 5 that the granule numbers increased by the hierarchical and dynamical granulation level in all circumstances. However, the increment tendency is quick on the initial granulation levels but slow on the last granulation levels. These results may be caused by the initial granulation level because the density of many granules is large and the radius of many

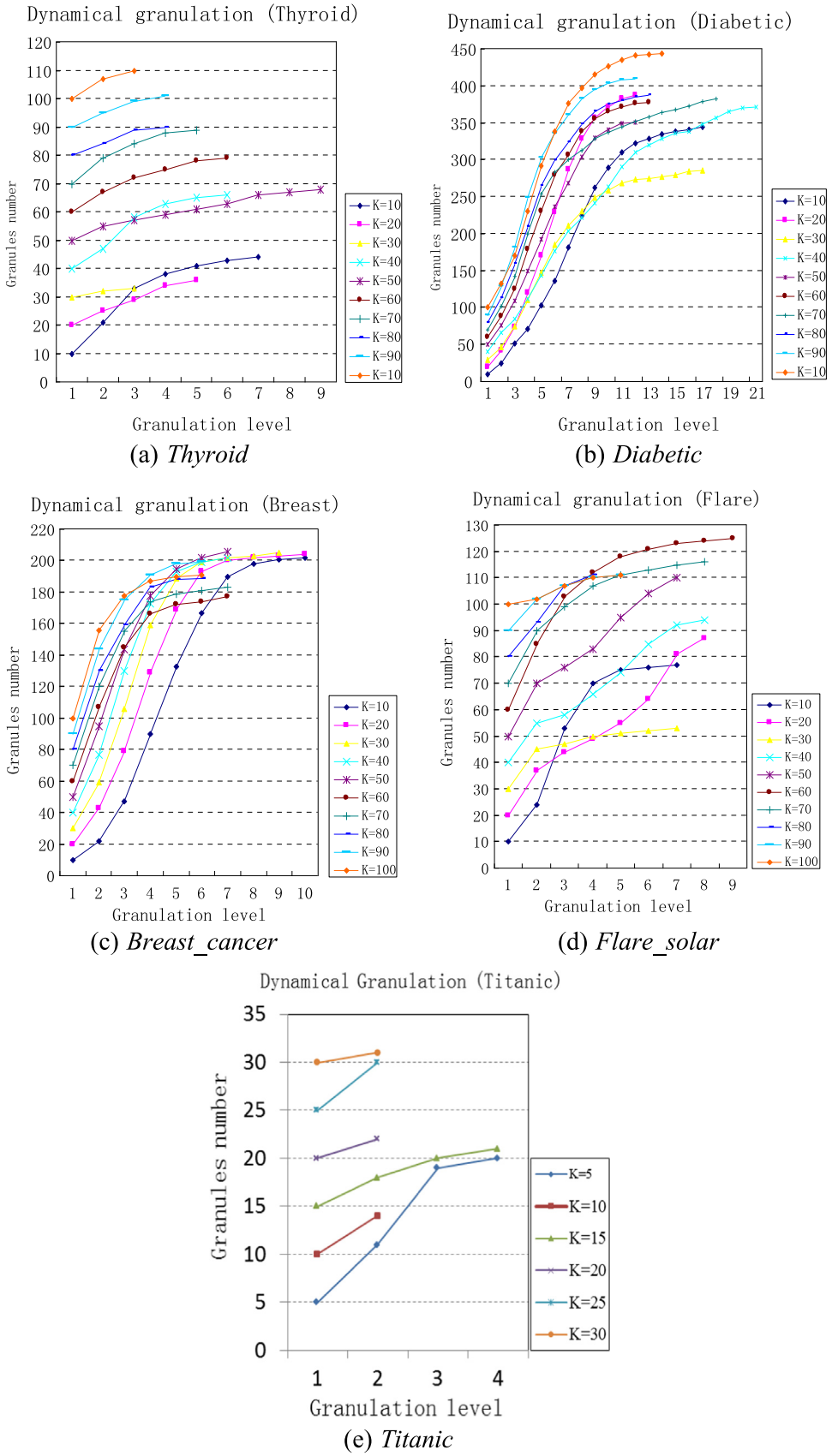

Fig. 5. Hierarchical and dynamical granulation result on every dataset. (a) *Thyroid* (b) *Diabetic* (c) *Breast_cancer* (d) *Flare_solar* (e) *Titanic*.

Table 2
Granule number of two methods.

Datasets	Dynamic granules	Statistic granules
Thyroid	[44/36/33/66/68/79/89/90/101/110]	[10/20/30/40/50/60/70/80/90/100]
Diabetic	[344/388/286/371/351/378/382/387/409/444]	[10/20/30/40/50/60/70/80/90/100]
Breast_cancer	[202/204/205/202/206/177/183/189/199/191]	[10/20/30/40/50/60/70/80/90/100]
Flare_solar	[77/87/53/94/110/125/116/111/111/111]	[10/20/30/40/50/60/70/80/90/100]
Titanic	[20/14/21/22/30/31]	[5/10/15/20/25/30]

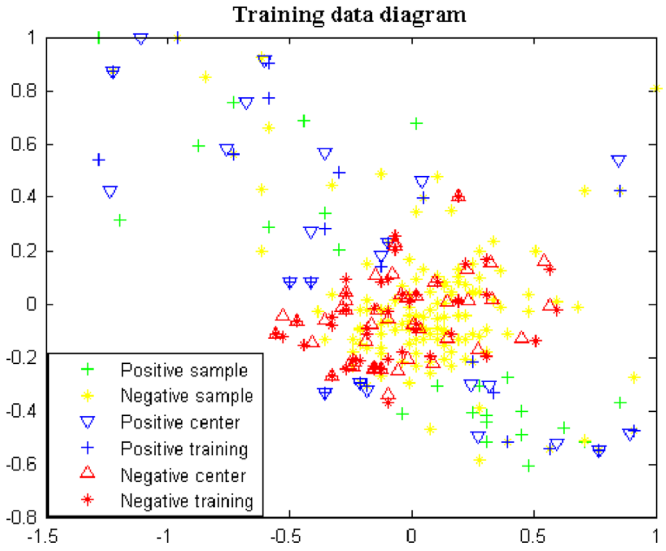

Fig. 6. Relevant dataset distribution of HD_GSVM ($K=50$).

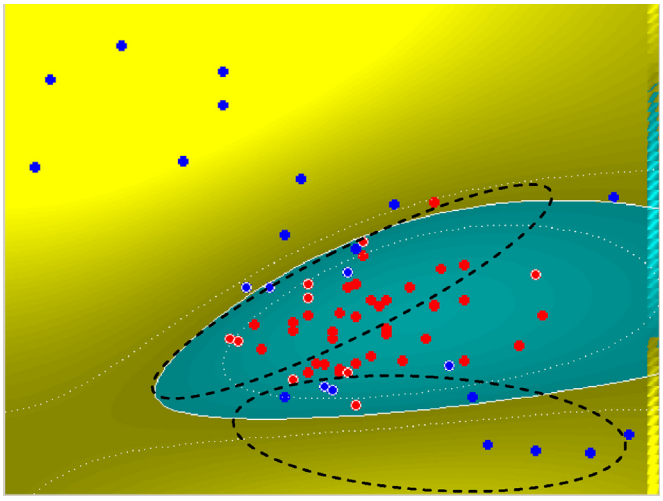

Fig. 7. The hyperplane of HD_GSVM on *Thyroid* ($K=50$).

granules is large, so the granules' number needed for re-granulation is very large. However, on the last granulation level, with the decrement of informational granules and the radius of them, the number of granules needing re-granulation is reduced. Therefore, the increment of the granulation number is slow on the last granulation level, and it illustrates that we can obtain important classification information by the smaller price of granule adding. We can also find that the granulation level number is smaller than 10 on all datasets except *Diabetic*. This illustrates that we can quickly obtain the optimal granulation circumstance by using the

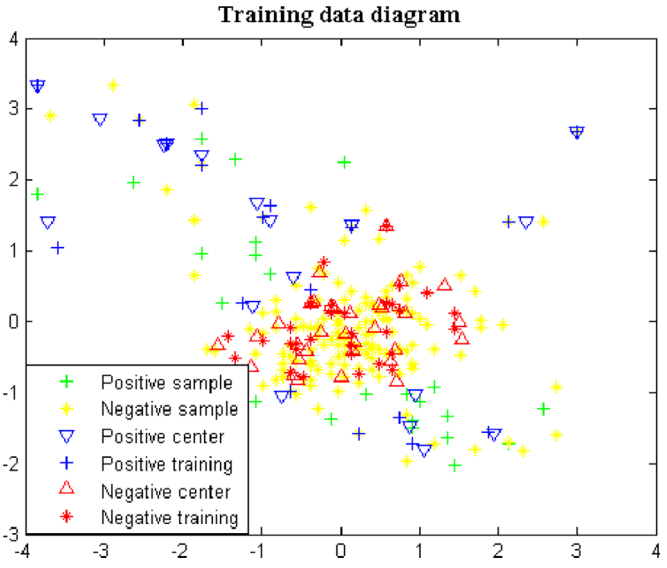

Fig. 8. Relevant dataset distribution of GSVM ($K=50$).

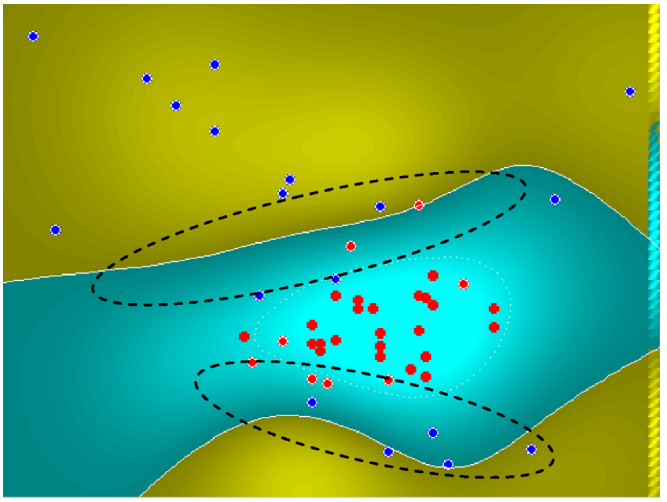

Fig. 9. The hyperplane of GSVM on *Thyroid* ($K=50$).

HD_GSVM method on most datasets at a smaller granulation level. Moreover, different from our estimation, the final granule number does not increase with an increase in the initial granulation number parameter. We can select a suitable initial granulation number parameter on practical problems. Table 2 is the final granule number of the HD_GSVM method and the traditional statistical GSVM method. For the traditional GSVM method, these values are equal to the granulation number parameters. However, for the HD_GSVM method, these values are larger than the initial granulation number parameters, and they are determined by the initial granulation number parameters.

We also give the original datasets distribution, granule centers distribution and last factual training data distribution on the *Thyroid* dataset when the initial granulation parameter of the HD_GSVM method takes 50 and the statistical granulation parameter of the traditional GSVM takes 50 (see Fig. 6–Fig. 9). In Figs. 6 and 8, the original positive and negative samples are represented by green "+" and yellow "*", and the blue "▽" and red "△" present the positive and negative granule center because these centers did not existed in the original datasets. So, we use the sample nearest to the centers of the same class to replace the centers of SVM training. In Figs. 6 and 8, the blue "+" and red "*" present the

Table 3
Testing accuracy of two methods with the same initial granulation number parameters (%).

Datasets	Methods	Testing accuracy										Range (▵(Acc))	Average (▵(Acc))
Thyroid	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100	[+0.9, +6.9]	+2.57
	HD_GSVM	96.5	97.0	96.3	96.6	98.1	98.2	98.7	98.7	97.9	97.7		
	GSVM	89.6	94.4	95.2	95.5	97.1	96.1	94.5	94.3	96.5	96.8		
Diabetic	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100	[+9.2, +19.5]	+13.6
	HD_GSVM	82.6	83.9	79.8	83.4	83.0	83.7	83.7	83.9	84.7	84.2		
	GSVM	66.6	64.5	70.6	67.5	70.7	69.3	72.3	74.3	70.2	70.9		
Breast_cancer	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100	[+10.4, +26.4]	+18.42
	HD_GSVM	92.1	91.4	92.2	93.8	92.7	88.7	87.7	88.3	89.9	88.3		
	GSVM	65.7	65.7	70.8	71.6	74.0	72.7	72.5	75.2	74.8	77.9		
Flare_solar	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100	[−10.2, +5.6]	+0.75
	HD_GSVM	64.6	65.3	64.1	65.3	65.3	67.7	67.7	56.6	55.7	55.7		
	GSVM	62.7	59.6	63.5	63.7	61.4	64.3	64.9	66.8	56.8	56.8		
Titanic	Initial granulation number parameter	GI=5	GI=10	GI=15	GI=20	GI=25	GI=30	–	–	–	–	[+0.1, +1.9]	+1.13
	HD_GSVM	79.2	78.5	79.2	78.5	79.2	78.5	–	–	–	–		
	GSVM	77.3	77.3	77.3	77.6	78.4	78.4	–	–	–	–		

Table 4
Training time of two methods with the same initial granulation number parameters (ms).

Datasets	Methods	Training time									
Thyroid	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100
	HD_GSVM	93.8	46.9	46.9	187.5	203.1	265.6	343.8	359.4	468.8	546.9
	GSVM	37.5	46.9	31.3	109.4	109.4	171.9	203.1	281.3	343.8	437.5
Diabetic	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100
	HD_GSVM	7828.1	10937.5	5093.8	9718.8	8531.3	10062.5	10375	10656.3	12343.8	15312.5
	GSVM	37.5	46.9	31.3	78.1	125	187.5	218.8	312.5	390.6	484.4
Breast_cancer	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100
	HD_GSVM	3515.6	3500	3421.9	3390.6	3531.3	2453.1	2718.8	2890.6	3234.4	2937.5
	GSVM	62.5	31.3	31.3	78.1	125	156.3	218.8	296.9	359.4	453.1
Flare_solar	Initial granulation number parameter	GI=10	GI=20	GI=30	GI=40	GI=50	GI=60	GI=70	GI=80	GI=90	GI=100
	HD_GSVM	250	343.8	125	406.3	562.5	796.9	656.3	578.1	578.1	578.1
	GSVM	155.6	155.6	31.3	62.5	93.8	140.6	156.3	328.1	328.1	328.1
Titanic	Initial granulation number parameter	GI=5	GI=10	GI=15	GI=20	GI=25	GI=30	–	–	–	–
	HD_GSVM	155.6	15.6	155.6	31.3	31.3	46.9	–	–	–	–
	GSVM	78.1	15.6	15.6	15.6	15.6	93.8	–	–	–	–

positive and negative actual training samples in these experiments. Figs. 7 and 9 are the hyperplanes obtained on the *Thyroid* dataset when the initial granulation parameter of HD_GSVM and GSVM take 50. We know from Table 2 that the actual granule number of the HD_GSVM last training is 68. In these two figures, we can find amazingly that the samples near the hyperplane of HD_GSVM are more than GSVM obviously. This illustrates that the HD_GSVM method can mine more classification information by hierarchical and dynamical granulation and by combining it with the density and radius of the granule.

4.2. Comparison of testing results

Tables 3 and 4 are the testing accuracy and the training time of the two methods with identical initial granulation number parameters. In Table 3, the GI presents the initial granulation number parameter, Range(▵(Acc)) presents the upper and lower bound of the testing accuracy superiority of HD_GSVM by comparing it with the traditional GSVM and the Average(▵(Acc)) presents the average value of the testing accuracy superiority of HD_GSVM by comparing it with the traditional GSVM. We find that for the

Table 5
Testing accuracy of two methods with the same actual training granules number (%).

Datasets	Methods	Testing accuracy										Range (△(Acc))	Average (△(Acc))
Thyroid	Training granules number	GA=44	GA=36	GA=33	GA=66	GA=68	GA=79	GA=89	GA=90	GA=101	GA=110	[+0.0, +2.9]	+0.76
	HD_GSVM	96.5	97.0	96.3	96.6	98.1	98.2	98.7	98.7	97.9	97.7		
	GSVM	96.2	96.1	93.4	96.3	98.1	97.8	98.2	97	97.5	97.5		
Diabetic	Training granules number	GA=344	GA=388	GA=286	GA=371	GA=351	GA=378	GA=382	GA=387	GA=409	GA=444	[+0.1, +2.5]	+0.96
	HD_GSVM	82.6	83.9	79.8	83.4	83.0	83.7	83.7	83.9	84.7	84.2		
	GSVM	81.3	83.0	79.1	82.8	81.9	83.0	83.5	82.7	84.2	81.8		
Breast_cancer	Training granules number	GA=202	GA=204	GA=205	GA=202	GA=206	GA=177	GA=183	GA=189	GA=199	GA=191	[+1.8, +4.9]	+2.86
	HD_GSVM	92.1	91.4	92.2	93.8	92.7	88.7	87.7	88.3	89.9	88.3		
	GSVM	88.8	89.6	89.6	88.8	89.6	84.9	84.9	86.1	87.7	86.5		
Flare_solar	Training granules number	GA=77	GA=87	GA=53	GA=94	GA=110	GA=125	GA=116	GA=111	GA=111	GA=111	[−9.6, +7.8]	+0.72
	HD_GSVM	64.6	65.3	64.1	65.3	65.3	67.7	67.7	56.6	55.7	55.7		
	GSVM	57.5	57.5	64.7	57.0	57.4	65.1	65.4	65.4	65.4	65.4		
Titanic	Training granules number	GA=20	GA=14	GA=21	GA=22	GA=30	GA=31	–	–	–	–	[−0.6, +1.5]	+0.77
	HD_GSVM	79.2	78.5	79.2	78.5	79.2	78.5	–	–	–	–		
	GSVM	77.6	77.3	77.6	78.4	78.4	79.2	–	–	–	–		

Table 6
Training time of two methods with the same actual training granules number (ms).

Datasets	Methods	Training time									
Thyroid	Training granules number	GA=44	GA=36	GA=33	GA=66	GA=68	GA=79	GA=89	GA=90	GA=101	GA=110
	HD_GSVM	93.8	46.9	46.9	187.5	203.1	265.6	343.8	359.4	468.8	546.9
	GSVM	171.9	78.1	93.8	281.3	296.9	437.5	531.3	562.5	718.8	859.4
Diabetic	Training granules number	GA=344	GA=388	GA=286	GA=371	GA=351	GA=378	GA=382	GA=387	GA=409	GA=444
	HD_GSVM	7828.1	10937.5	5093.8	9718.8	8531.3	10062.5	10375	10656.3	12343.8	15312.5
	GSVM	13046.9	17734.4	8171.9	15828.1	13703.1	16593.8	17671.9	17843.8	20890.6	25890.6
Breast_cancer	Training granules number	GA=202	GA=204	GA=205	GA=202	GA=206	GA=177	GA=183	GA=189	GA=199	GA=191
	HD_GSVM	3515.6	3500	3421.9	3390.6	3531.3	2453.1	2718.8	2890.6	3234.4	2937.5
	GSVM	3234.4	3312.5	3328.1	3234.4	3406.3	2359.4	2656.3	2875	3171.9	2843.8
Flare_solar	Training granules number	GA=77	GA=87	GA=53	GA=94	GA=110	GA=125	GA=116	GA=111	GA=111	GA=111
	HD_GSVM	250	343.8	125	406.3	562.5	796.9	656.3	578.1	578.1	578.1
	GSVM	343.8	468.8	156.3	468.8	921.9	1187.5	843.8	687.5	687.5	687.5
Titanic	Training granules number	GA=20	GA=14	GA=21	GA=22	GA=30	GA=31	–	–	–	–
	HD_GSVM	155.6	15.6	155.6	31.3	31.3	46.9	–	–	–	–
	GSVM	62.5	15.6	15.6	62.5	46.9	46.9	–	–	–	–

Thyroid dataset, the testing accuracy results of HD_GSVM are [0.9%, 4.4%] larger than GSVM, and the average value of it is 2.57%. For the *Diabetic* dataset, the testing accuracy results of HD_GSVM are [9.2%, 19.5%] larger than GSVM, and the average of it is 13.6%. For the *Breast_cancer* dataset, the testing accuracy results of HD_GSVM are [10.4%, 16.0%] larger than GSVM, and the average of it is 18.42%. For the *Titanic* dataset, the testing accuracy results of HD_GSVM are [0.12%, 1.85%] larger than GSVM, and the average of it is 1.13%. However, although the average testing accuracy results of HD_GSVM is 0.75% larger than GSVM, when the initial granulation parameter takes 80 to 100, the testing accuracy results of HD_GSVM is smaller than GSVM. This may be because this dataset is not very suitable to the granulation method of clustering based on European distance.

Because the training time of SVM model is directly determined by the actual training samples size, so the training time of HD_GSVM method is longer than the traditional GSVM method on datasets *Diabetic* and *Breast_cancer*. On other datasets, the training time of HD_GSVM is a little longer than GSVM method. We can find from Table 2 the actual granules number vector [344/388/…/409/444] on *Diabetic* and granules number vector [202/204/…/199/191] on *Breast_cancer* of HD_GSVM are larger than granules number vectors [10/20/…/90/100] of traditional GSVM obviously. So the training time of HD_GSVM is larger than GSVM on these two datasets obviously. The training time of HD_GSVM is longer than traditional GSVM methods slightly, but comparing with the

SVM training directly, the efficiency of HD_GSVM is accepted easily because it improves the generalization performance greatly with high learning efficiency synchronously.

Besides the former identical granulation number parameter setting, we also test when the results of granules number are identical of these two methods. Because the granules number of GSVM is equal to the initial granulation number parameter, in these experiments, we set the initial granulation number parameters of GSVM method identical with the last hierarchical and dynamic granules in Table 2, that the factual training granules numbers are same for these two methods. The testing results are shown in Table 5. the GA present the actual training granules number, and the meaning of Range(△(Acc)) and Average(△(Acc)) are same to the former Table 3.

In this table, we can find that except when GA=53 or GA=111 of dataset *Flare_solar* and G=31 of dataset *Titanic*, the testing accuracy of HD_GSVM are larger than GSVM on all datasets when the granules number of them are identical on other circumstances. Besides, the training time of HD_GSVM is smaller than GSVM for the statistical granulation of traditional GSVM being on all training datasets, but for HD_GSVM method, the first granulation are on all training datasets and the granulation on last granulation level are on smaller granules. So the training time of HD_GSVM on most datasets is smaller than GSVM especially when the granules are large such as *Diabetic* (Table 6).

5. Conclusions

Because SVM is usually not used for large-scale data mining problem because training complexity is highly dependent on the dataset size. This paper presents an improved support vector machine learning model based on hierarchical and dynamical granulation, to solve the low efficiency and generalization performance problem for traditional GSVM. For HD_GSVM, the original data will be mapped into a high-dimensional space by mercer kernel. Then, the data are divided into some granules, and those granules near to approximate hyperplane are extracted and re-granulation in subtle level by their density degree and radius degree hierarchically and dynamically. Finally, the decision hyperplane will be obtained through all the granules at different granulation levels effectively. In the hierarchical and dynamical granulation process, the granulation level of all granules can be dynamically changed at different granulation level continuously. By this method, different classification information on various granulation levels can be used and we can divide granules on various levels, to meet variety needs of practical problems. The HD_GSVM model can improve the generalization performance greatly with high learning efficiency synchronously and it can be used into the large scale application problems such as web page classification, social network, genes or proteins analysis.

Although the HD_GSVM method can improve the generalization performance greatly to solve the large scale classification problem with high learning efficiency. However, similar with traditional GSVM method, the HD_GSVM method only uses a one-way granulation method. In future work, we will investigate the hierarchical and dynamical granulation SVM based on double-direction granulation that contains granulation and combination in hierarchical and dynamical changing process to enhance the generalization performance further for large scale classification problem. Besides, the hierarchical and dynamical granulation parameter also affects the results of HD_GSVM algorithm, this paper not discusses it deeply for simplify and setting, so the hierarchical and dynamical granulation parameter of HD_GSVM will need to be discussed and researched further by the specific different features of datasets.

Acknowledgement

The work described in this paper was partially supported by the National Natural Science Foundation of China (Nos. 61503229, 61273291), Research Project Supported by Shanxi Scholarship Council of China (No. 2012-008), Natural Science Foundation of ShanXi Province (No. 2015021096) and Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi (No. 2015110).

References

[1] V. Vapnik, Statistical Learning Theory, Wiley Press, New York 1998, pp. 493–520.
[2] R.L. Irene, C.S. Cruz, R. Huerta, Hierarchical linear support vector machine, Pattern Recognit. 45 (12) (2012) 4414–4427.
[3] S.F. Ding, J.Z. Yu, H.J. Huang, et al., Twin support vector machines based on particle swarm optimization, J. Comput. 8 (9) (2013) 2296–2303.
[4] T. Harris, Credit scoring using the clustered support vector machine, Expert. Syst. Appl. 42 (2) (2015) 741–750.
[5] J.H. Xu, Multi-label core vector machine with a zero label, Pattern Recognit. 47 (7) (2014) 2542–2557.
[6] W.J. Wang, H.S. Guo, Y.F. Jia, et al., Granular support vector machine based on mixed measure, Neurocomputing 101 (2013) 116–128.
[7] X.B. Kong, X.J. Liu, R.F. Shi, et al., Wind speed prediction using reduced support vector machines with feature selection, Neurocomputing 169 (2015) 449–456.
[8] D. Wang, B. Zhang, P. Zhang, et al., An online core vector machine with adaptive MEB adjustment, Pattern Recognit. 43 (10) (2010) 3468–3482.
[9] S.F. Ding, X.P. Hua, Recursive least squares projection twin support vector machines, Neurocomputing 130 (2014) 3–9.
[10] J.H. Li, C.L. Mei, W.H. Xu, et al., Concept learning via granular computing: a cognitive viewpoint, Inf. Sci. 298 (2015) 447–467.
[11] J.T. Yao. A ten year review of granular computing. in: Proceedings of 2007 IEEE International Conference on Granular Computing, 2007, pp. 734–739.
[12] Y.C. Tang, B. Jin, Y. Sun. et al., Granular support vector machines for medical binary classification problems, in: Proceedings of the IEEE CIBIB. Piscataway, HJ: IEEE Comput. Intell. Soc. 2004, pp.73–78.
[13] E. Osuna, R. Freund, F. Girosi. Training support vector machines: an application to face detection. in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, pp. 130–136.
[14] X.L. Huang, L. Shi, J.A. Suykens. Sequential minimal optimization for SVM with pinball loss 149, 2015, pp. 1596–1603.
[15] C.C. Chang, C.J. Lin. LIBSVM-a library for support vector machines. Available from: http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.
[16] P.Y. Hao, J.H. Chiang, Y.K. Tu, Hierarchically SVM classification based on support vector clustering method and its application to document categorization, Expert. Syst. Appl. 33 (3) (2007) 627–635.
[17] S.X. Cheng, Y.S. Frank, An improved incremental training algorithm for support vector machines using active query, Pattern Recognit. 40 (3) (2007) 964–971.
[18] H.S. Guo, W.J. Wang, C.Q.Men. A novel learning model-kernel granular support vector machine. in: Proceedings of 2009 International Conference on Machine Learning and Cybernetics (ICMLC2009), IEEE Press, 2009, pp. 930–935.
[19] X.L. Tang, L. Zhuang, J. Cai, et al., Multi-fault classification based on support vector machine trained by chaos particle swarm optimization, Knowl. Based Syst. 23 (5) (2010) 486–490.
[20] P.F. Pai, M.F.H, M.C. Wang, A support vector machine-based model for detecting top management fraud, Knowl. Based Syst. 24 (2) (2011) 314–321.
[21] A.K. Deb, Jayadeva, M. Gopal, et al., SVM-based tree-type neural networks as a critic in adaptive critic designs for control, IEEE Trans. Neural Netw. 18 (4) (2007) 1101–1114.
[22] S.N. Pang, N. Kasabov, Encoding and decoding the knowledge of association rules over SVM classification trees, Knowl. Inf. Syst. 19 (1) (2009) 79–105.
[23] J.H. Tan, Infrared gait recognition based on rough set support vector machine, Comput. Eng. Des. 33 (4) (2012) 1542–1546.
[24] M.A. Kumar, M. Gopal, A hybrid SVM based decision tree, Pattern Recognit. 43 (12) (2010) 3977–3987.
[25] S.F. Ding, B.J. Qi, Research of granular support vector machine, Artif. Intell. Rev. 38 (1) (2012) 1–7.
[26] S.F. Ding, H.J. Huang, J.Z. Yu, et al., Research on the hybrid models of granular computing and support vector machine, Artif. Intell. Rev. 43 (6) (2015) 565–577.
[27] W.J. Wang, H.S. Guo, Granular support vector machine learning model, J. Shanxi Univ. (Nat. Sci. Ed.) 32 (4) (2009) 535–540.
[28] W.J. Wang, Z.B. Xu, V.Z. Lu, et al., Determination of the spread parameter in the Gaussian kernel for classification and regression, Neurocomputing 55 (2003) 643–663.
[29] H.S. Guo, W.J. Wang, An active learning-based SVM multi-class classification model, Pattern Recognit. 48 (5) (2015) 1577–1597.
[30] W.J. Wang, J.L. Guo, C.Q. Men, An approach for kernel selection based on data distribution, Lect. Notes Artif. Intell. 5009 (2008) 596–603.
[31] M. Tian, W.J. Wang, An efficient Gaussian kernel optimization based on

centered kernel polarization criterion, Inf. Sci. 322 (2015) 133–149.
[32] K.P. Wu, S.D. Wang, Choosing the kernel parameters for support vector machines by the inter-cluster distance in the feature space, Pattern Recognit. 42 (5) (2009) 710–717.
[33] H.X. Zhang, J. Lu, Semi-supervised fuzzy clustering: a kernel-based approach, Knowl. Based Syst. 22 (6) (2009) 477–481.
[34] F.Y. Lin, C.C. Yeh, M.Y. Lee, the use of hybrid manifold learning and support vector machines in the prediction of business failure, Knowl. Based Syst. 24 (1) (2011) 95–101.
[35] P. Wittek, C.L. Tan, Compactly supported basis functions as support vector kernels for classification, IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 2039–2050.
[36] UCI Machine Learning Repository, 2010. Available from: http://ww.ics.uci.edu/mleam/MLRepository.html.

Guo Husheng obtained his Ph.D. degree from Department of Computer and Information Technology, Shanxi University in 2014. Now he is lecturer at School of Computer and Information Technology, Shanxi University. He has published more than 10 academic papers on machine learning and support vector machine. His research interests include support vector machine, kernel methods and machine learning, etc.

Wang Wenjian obtained her Ph.D. degree from Institute for Information and System Science, Xi'an Jiaotong University in 2004. Now she is a professor and Ph. D. supervisor at School of Computer and Information Technology and Key Laboratory of Computational Intelligence and Chinese Information Processing, Shanxi University. She has published more than 70 academic papers on machine learning, computational intelligence and data mining. Her current research interests include support vector machine, neural networks, machine learning and environmental computations, etc. She received the B.S. degree in computer science from Shanxi University, China, in 1990, the M.S. degree in computer science from Hebei Polytechnic University, China, in 1993, and Ph.D. degree in applied mathematics from Xi'an Jiao Tong University, China, in 2004. She worked as a research assistant at the Department of Building and Construction, The City University of Hong Kong from May 2001 to May 2002. She has been with the Department of Computer Science at Shanxi University since 1993, where she was promoted as Associate Professor in 2000 and as Full Professor in 2004, and now serves as a Ph.D. supervisor in Computer Application Technology and System Engineering. She has published more than 70 academic papers on machine learning, computational intelligence, and data mining. Her current research interests include neural networks, support vector machines, machine learning theory and environmental computations, etc.