Extending Granular Ball Twin Support Vector Machine for Multi-class Classification

Hung.C Nguyen[1][0009-0003-2194-3876] and Quang-Thinh Bui[2][0000-0002-7357-2574]

[1] HUTECH University, Ho Chi Minh City, Vietnam
[2] Tien Giang University, Tien Giang, Vietnam
buiquangthinh@tgu.edu.vn

Abstract. Multi-class classification is a key area in machine learning, with Support Vector Machines (SVMs) and their variant, Twin SVM (TSVM), showing significant effectiveness in various applications. Recently, Granular Ball Twin SVM (GBTSVM) was introduced, combining Granular Computing with TSVM to reduce data noise and enhance performance, and it has been applied in several prior studies to multi-label classification tasks. This paper proposes Granular Ball Twin Support Vector Machine for Multi-class Classification (GBTSVM_MC), a novel method to extend GBTSVM for multi-class tasks. GBTSVM_MC employs two decomposition strategies, One-vs-One (OvO) and One-vs-Rest (OvR), to handle multi-class datasets. It leverages TSVM's strong classification ability and the computational efficiency of granular balls while improving computation for multi-label issues. To experimental evaluation of the performance of GBTSVM_MCC on diverse datasets along with additional testing on large datasets, thereby verifying the scalability and practical applicability of the model.

Keywords: Key Word: Multi-class Classification, Support Vector Machine, Twin Support Vector Machine, Granular Ball Computing, Noise Resistance, Large Scale Dataset

1 Introduction (đã sửa)

In the field of machine learning, classification algorithms have made significant strides, with Support Vector Machines (SVMs) standing out due to their effectiveness in handling multidimensional data and their strong theoretical foundation in statistical learning theory. SVMs have demonstrated success in diverse applications, including disease diagnosis, anomaly detection, and image classification. Originally designed for binary classification, SVMs solve a large quadratic programming problem (QPP) to find the optimal separating hyperplane. However, this design limits their ability to directly address multiclass scenarios and large-scale datasets, prompting the development of advanced variants such as Twin Support Vector Machines (TSVMs). TSVMs enhance efficiency by solving two smaller QPPs instead of a single large one, achieving computation speeds up to four times faster than traditional SVMs. Building on this, the Granular Ball Twin Support Vector Machine (GBTSVM) incorporates granular

computing to represent data as granular balls rather than individual points, improving noise immunity and computational efficiency.

The motivation to extend binary classification methods to multiclass settings arises from the growing complexity of real-world datasets, where data often span multiple categories, such as in image segmentation or medical diagnosis. Traditional binary classifiers struggle with challenges like the curse of dimensionality and high computational costs when applied to multiclass problems, necessitating efficient strategies such as One-vs-One (OvO) and One-vs-Rest (OvR). To address these issues, several multiclass models for SVMs have emerged. Among them, the Granular Ball K-Class Twin Support Vector Classifier (GB-TWKSVC) tackles multiclass problems by combining GBTSVM with the One-versus-One-versus-Rest (OvOvR) strategy. Despite its advancements, there remains a need for more adaptive and computationally efficient approaches to handle large-scale datasets effectively.

This paper proposes a novel framework, Granular Ball Twin Support Vector Machine for Multi-class Classification (GBTSVM_MC), which extends GBTSVM to multiclass tasks by employing OvO and OvR decomposition strategies instead of the more complex OvOvR approach. Our method integrates the computational efficiency of TSVM with the robust data representation of granular balls to tackle key challenges in multiclass classification, such as scalability to large-scale data and noise immunity. The contributions of this work are outlined as follows:

- **Improved Computational Efficiency for Large-Scale Data**: GBTSVM_MC builds on the TSVM framework by breaking down the classification problem into smaller QPPs and uses a fine-grained granular ball representation to reduce the number of input data points. This combined approach significantly lowers computational costs, making the model well-suited for large-scale datasets, as evidenced by experiments on datasets like Global Cancer, which includes up to 50,000 samples.
- **Flexible Multiclass Processing with OvO and OvR Strategies**: Unlike GB-TWKSVC, which relies on the OvOvR strategy, GBTSVM_MC adopts OvO and OvR methods to decompose multiclass problems into manageable binary subproblems. This flexibility enables the model to adapt to diverse dataset structures, enhancing both accuracy and training speed, as shown by comparative results on various UCI and Kaggle datasets.
- **Enhanced Noise Robustness and Scalability**: By using granular balls as input units, the model improves resilience to noise and outliers, which is critical for real-world applications with imperfect data. Additionally, extensive testing on subsets of the Global Cancer dataset (ranging from 1,000 to 50,000 samples) confirms the model's scalability and practical applicability across different datasets and domains.

The structure of this paper is organized as follows. Section 2 provides a comprehensive background and mathematical foundation of related models, including TSVM, GBTSVM, and GB-TWKSVC, which form the basis for the advancements we propose. Section 3 elaborates on the GBTSVM_MC methodology, covering granular ball generation, multiclass classification strategies, and algorithm design. Section 4 presents the experimental setup and results, highlighting the efficiency and scalability of

GBTSVM_MC across multiple datasets. Finally, Section 5 discusses the implications of our findings and identifies potential directions for future research.

2 Background and Related work

2.1 Granular Ball Twin Support Vector Machines (GBTSVM)

GBTSVM (Granular Ball Twin Support Vector Machines) was proposed by A. Quadir based on the idea of combining the stability of the granular ball computing method and the superior computational efficiency of TSVM. This idea helps the model increase its noise tolerance, thanks to performing calculations based on particles instead of individual data points. At the same time, the computational time is significantly reduced because only a small quadratic programming problem (QPP) needs to be solved for each label, instead of having to handle a large QPP problem for the entire data set.

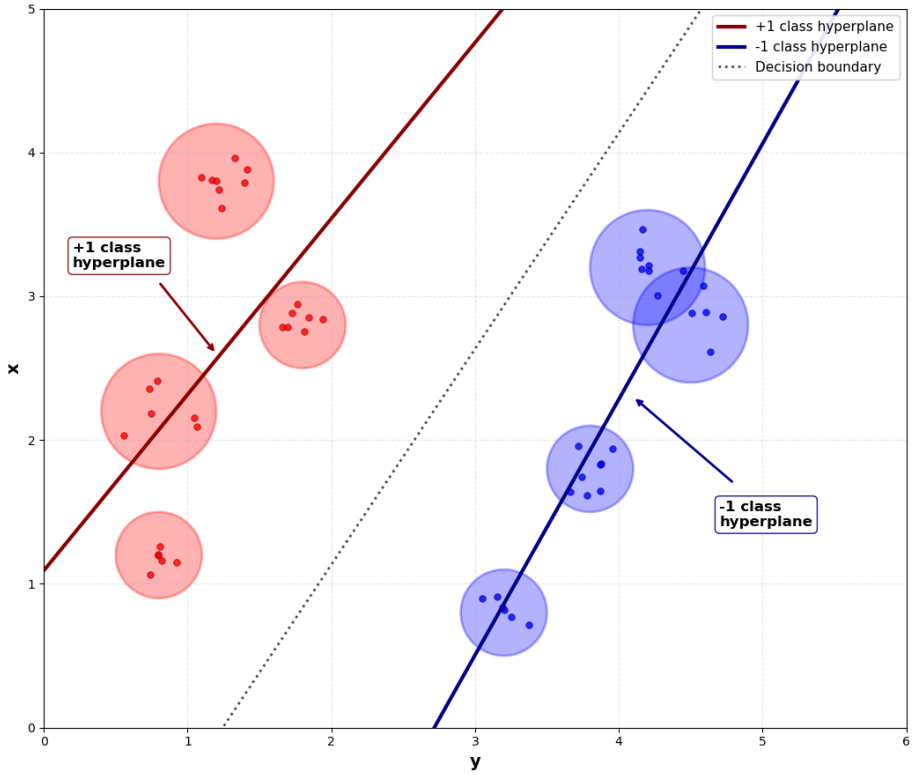

Fig. 1. Visualization of GBTSVM

The mechanism of Granular Ball Computing is a data processing technique to solve the scalability problem with high-dimensional data, by using Granular Balls (GB) defined by the center and radius instead of the entire data points, to reduce the amount of information to be processed and increase the computational efficiency.

To explain it mathematically, let the set of generated GBs be denoted as $S = GB_i, i = 1,2,\dots,p = \big((c_i, r_i), y_i\big), i = 1,2,\dots,p$, where c_i denotes the center, r_i denotes the

radius, and y_i is the label of the $i-th$ GB. This mechanism is represented in the form of an optimization formula as follows:

$$\begin{aligned} &min\ \lambda_1 \frac{n}{\sum_{i=1}^{p} GB_i} + \lambda_2 p \\ &s.t.\ pur\big(GB_j\big) \geq T, j = 1,2,\dots,p \end{aligned} \tag{1}$$

where λ_1 and λ_2 are weighting coefficients, and T is the purity threshold, which represents the proportion of samples in the GB with the same label that are in the majority. After determining the GBs for each layer, the center and radius of these GBs will be provided as input to the GBTSVM in the following manner:

$$\begin{aligned} &\min_{w_1,\, b_1} \frac{1}{2} \parallel C_1 w_1 + e_1 b_1 \parallel^2 + d_1 e_2^T \xi_2, \\ &\text{s.t.} -(C_2 w_1 + e_2 b_1) + \xi_2 \geq e_2 + R_2, \xi_2 \geq 0. \end{aligned} \tag{2}$$

$$\begin{aligned} &\min_{w_2,\, b_2} \frac{1}{2} \parallel C_2 w_2 + e_2 b_2 \parallel^2 + d_2 e_1^T \xi_1, \\ &\text{s.t.} (C_1 w_2 + e_1 b_2) + \xi_1 \geq e_1 + R_1, \xi_1 \geq 0. \end{aligned} \tag{3}$$

where C_1 and C_2 are the center matrix of GBs for layers +1 and -1, and R_1 and R_2 are the radius of GBs.

By applying the Lagrange method and utilizing the K.K.T conditions, deriving the dual form of problem $\min_{w_1,\, b_1} \frac{1}{2} \parallel C_1 w_1 + e_1 b_1 \parallel^2 + d_1 e_2^T \xi_2,\ \text{s.t.} -(C_2 w_1 + e_2 b_1) + \xi_2 \geq e_2 + R_2, \xi_2 \geq 0.$ (2) as follows:

$$\begin{aligned} &\max_{\alpha}\ \alpha^T (e_2 + R_2) - \frac{1}{2} \alpha^T G (H^T H + \delta I)^{-1} G^T \alpha, \\ &\text{s.t. } 0 \leq \alpha \leq d_1 e_2. \end{aligned} \tag{4}$$

where $G = [C_1 \quad e_1], H = [C_2 \quad e_2]$ and δI is the regularization term to avoid the matrix inversion problem.

Similarly, the Wolfe dual for $\min_{w_2,\, b_2} \frac{1}{2} \parallel C_2 w_2 + e_2 b_2 \parallel^2 + d_2 e_1^T \xi_1,\ \text{s.t.} (C_1 w_2 + e_1 b_2) + \xi_1 \geq e_1 + R_1, \xi_1 \geq 0.$ (3) can be derived as follows:

$$\begin{aligned} &\max_{\gamma}\ \gamma^T (e_1 + R_1) - \frac{1}{2} \gamma^T H (G^T G + \delta I)^{-1} H^T \gamma, \\ &\text{s.t. } 0 \leq \gamma \leq d_2 e_1. \end{aligned} \tag{5}$$

As a result, the vectors $u_1 = [w_1; b_1]$representing the +1 class and $u_2 = [w_2; b_2]$ representing the −1 class can be set up as follows:

$$u_1 = -(H^T H + \delta I)^{-1} G^T \alpha \tag{6}$$

$$u_2 = (G^T G + \delta I)^{-1} H^T \gamma \tag{7}$$

From here, when a new data point x is classified based on its closest distance to two hyperplanes as described below:

$$\text{class}(x) = \arg\min_{i\in\{1,2\}} \frac{|w_i^T x + b_i|}{\|w_i\|} \tag{8}$$

2.2 Twin k-class Support Vector Machines (Twin-KSVC)

Aiming to apply multi-label classification to TSVM, Twin-KSVC was also introduced by Yitian Xu. Based on the idea of the "1-versus-1-versus rest" (OvOvR) method, two separate sets of samples are selected from k classes and treated as core partitions. This algorithm maps the remaining samples to an intermediate region between these non-parallel hyperplanes, resulting in a ternary output system {-1, 0, +1}.

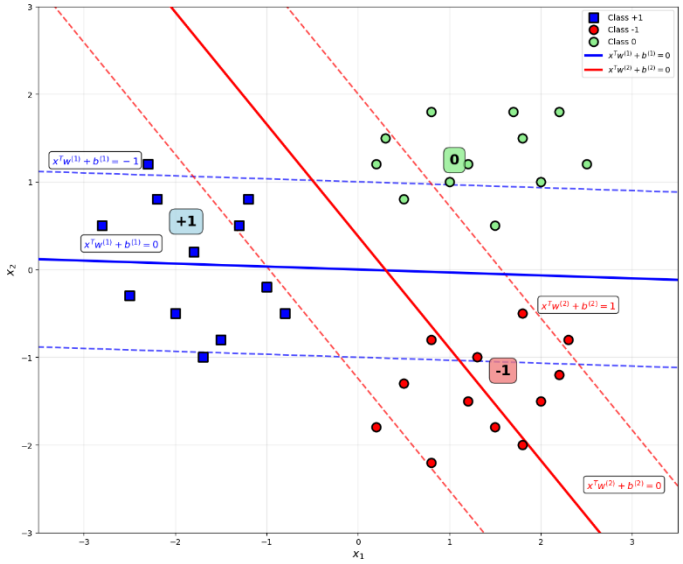

Fig. 2. Visualization of Twin-KSVC

Consider $A \in \mathbb{R}^{n_1\times d}$ and $B \in \mathbb{R}^{n_2\times d}$ representing two sets of centroid samples labeled +1 and -1 respectively. The remaining samples are represented by $C \in \mathbb{R}^{n_3\times d}$ and labeled 0. The mechanism for computing these hyperplanes is obtained by solving the refined QPP pair as follows:

$$\begin{aligned} &\min_{w_1,\, b_1} \frac{1}{2} \| Aw_1 + e_1 b_1 \|^2 + c_1 e_2^T \xi + c_2 e_3^T \eta, \\ &\text{s.t. } -(Bw_1 + e_2 b_1) + \xi \geq e_2, \xi \geq 0, \\ &-(Cw_1 + e_3 b_1) + \eta \geq (1-\varepsilon) e_3, \eta \geq 0 \end{aligned} \tag{9}$$

$$\begin{aligned} &\min_{w_2,\, b_2} \tfrac{1}{2} \| Bw_2 + e_2 b_2 \|^2 + c_3 e_1^T \xi^* + c_4 e_3^T \eta^*, \\ &\text{s.t. } -(Aw_2 + e_1 b_1) + \xi^* \geq e_1, \xi^* \geq 0, \\ &-(Cw_2 + e_3 b_1) + \eta^* \geq (1-\varepsilon) e_3, \eta^* \geq 0 \end{aligned} \tag{10}$$

where η and $\eta^* \in \mathbb{R}^{n_3 \times 1}$, and $\xi \in \mathbb{R}^{n_2 \times 1}$ and $\xi^* \in \mathbb{R}^{n_1 \times 1}$

By using the lagrange function and the Karush–Kuhn–Tucker (KKT) condition, the dual formulation of

$$\begin{aligned} &\min_{w_1,\, b_1} \tfrac{1}{2} \| Aw_1 + e_1 b_1 \|^2 + c_1 e_2^T \xi + c_2 e_3^T \eta, \\ &\text{s.t. } -(Bw_1 + e_2 b_1) + \xi \geq e_2, \xi \geq 0, \\ &-(Cw_1 + e_3 b_1) + \eta \geq (1-\varepsilon) e_3, \eta \geq 0 \end{aligned} \tag{9}$$

and

$$\begin{aligned} &\min_{w_2,\, b_2} \tfrac{1}{2} \| Bw_2 + e_2 b_2 \|^2 + c_3 e_1^T \xi^* + c_4 e_3^T \eta^*, \\ &\text{s.t. } -(Aw_2 + e_1 b_1) + \xi^* \geq e_1, \xi^* \geq 0, \\ &-(Cw_2 + e_3 b_1) + \eta^* \geq (1-\varepsilon) e_3, \eta^* \geq 0 \end{aligned}$$

(10) is established as shown below:

$$\begin{aligned} &\max_{\gamma} -\tfrac{1}{2} \gamma^T T (U^T U)^{-1} T^T \gamma + e_4^T \gamma, \\ &\text{s.t. } 0 \leq \gamma \leq E. \end{aligned} \tag{11}$$

$$\begin{aligned} &\max_{\rho} -\tfrac{1}{2} \rho^T P (V^T V)^{-1} P^T \rho + e_5^T \rho, \\ &\text{s.t. } 0 \leq \rho \leq E^*. \end{aligned} \tag{12}$$

for

$$\begin{aligned} &\max_{\gamma} -\tfrac{1}{2} \gamma^T T (U^T U)^{-1} T^T \gamma + e_4^T \gamma, \\ &\text{s.t. } 0 \leq \gamma \leq E. \end{aligned}$$

(11), the symbols are defined as $U = [A \quad e_1], V = [B \quad e_2], G = [C \quad e_3], T = [V \quad G], \gamma = [\alpha \quad \beta], e_4 = [e_2 \quad e_3(1-\epsilon)]$ and $E = [c_1 e_2 \quad c_2 e_3]$ and likewise with

$$\begin{aligned} &\max_{\rho} -\tfrac{1}{2} \rho^T P (V^T V)^{-1} P^T \rho + e_5^T \rho, \\ &\text{s.t. } 0 \leq \rho \leq E^*. \end{aligned}$$

(12).

Therefore, the solution vector $u = [w_1; b_1]$ and $u^* = [w_2; b_2]$ can be obtained as:

$$u = -(U^\top U + \delta I)^{-1} (V^\top \alpha + G^\top \beta) \tag{13}$$

$$u^* = -(V^\top V + \delta I)^{-1} (U^\top \alpha + G^\top \beta) \tag{14}$$

where δI is regularization term with δ is a very small positive number.

For a new checkpoint x_i, to determine its class label through the decision function as follows:

$$f(x_i) = \begin{cases} 1, if x_i^\top \mu_1 + e b_1 > -1 + \epsilon \\ -1, if x^\top \mu_2 + e b_2 < 1 - \epsilon \\ 0, \text{otherwise} \end{cases} \tag{15}$$

This Twin-KSVC formula provides an optimal solution for multi-class classification, significantly improving computational efficiency and accuracy when processing complex datasets.

2.3 Granular Ball K-Class Twin Support Vector Classifier (GB-TWKSVC)

GB-TWKSVC, introduced by M. A. Ganai, is an improved version of GBTSVM, which moves from binary to multi-class classification using the OvOvR strategy, incorporating GBs representation to increase noise immunity and computational efficiency. In OvOvR, for each pair of classes, two non-parallel hyperplanes are constructed to separate the two selected classes, while the other classes are placed in an "epsilontube" region around the hyperplanes.

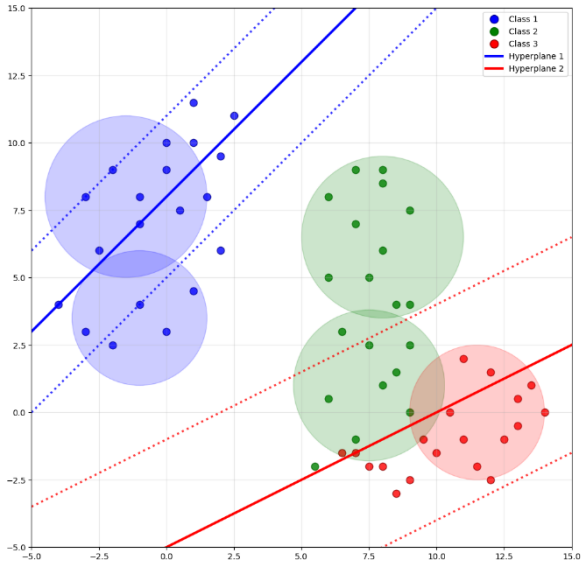

Fig. 3. Visualization for GB-TWKSVC

In GB-TWKSVC, the training dataset is denoted as $\tilde{I}$, which consists of GBs sets I_k corresponding to each class k.

$$\tilde{I} = \cup_{k=1}^{K} I_k \tag{16}$$

Then, to solve the QPP problem, GB-TWKSVC must improve GBTSVM to suit OvOvR is as follows:

$$\begin{aligned} \min_{w_i,b_i,\eta_i,\xi_j,\xi_r} \frac{1}{2}c_1(\| w_i \|^2 + b_i^2) + \frac{1}{2}\eta_i^T\eta_i + c_2 e_j^T\xi_j + c_3 e_r^T\xi_r, \\ \text{s.t. } C_i w_i + e_i b_i = \eta_i, \\ -(C_j w_i + e_j b_i) + \xi_j \geq e_j + R_j, \xi_j \geq 0, \\ | C_r w_i + e_r b_i | + \xi_r \geq e_r + R_r, \xi_r \geq 0. \end{aligned} \tag{17}$$

$$\begin{aligned} \min_{w_j,b_j,\eta_j,\xi_i,\xi_r} & \frac{1}{2}c_4(\| w_j \|^2 + b_j^2) + \frac{1}{2}\eta_j^T\eta_j + c_5 e_i^T\xi_i + c_6 e_r^T\xi_r, \\ & \text{s.t. } C_j w_j + e_j b_j = \eta_j, \\ & (C_i w_j + e_i b_j) + \xi_i \geq e_i + R_i, \xi_i \geq 0, \\ & | C_r w_j + e_r b_j | + \xi_r \geq e_r + R_r, \xi_r \geq 0. \end{aligned} \tag{18}$$

Similar to the Twin-KSVC approach, with R_1, R_2, R_3 being the set of radii of the corresponding class, the dual formulation of

$$\begin{aligned} \min_{w_i,b_i,\eta_i,\xi_j,\xi_r} & \frac{1}{2}c_1(\| w_i \|^2 + b_i^2) + \frac{1}{2}\eta_i^T\eta_i + c_2 e_j^T\xi_j + c_3 e_r^T\xi_r, \\ & \text{s.t. } C_i w_i + e_i b_i = \eta_i, \\ & -(C_j w_i + e_j b_i) + \xi_j \geq e_j + R_j, \xi_j \geq 0, \\ & | C_r w_i + e_r b_i | + \xi_r \geq e_r + R_r, \xi_r \geq 0. \end{aligned} \tag{17}$$

and

$$\begin{aligned} \min_{w_j,b_j,\eta_j,\xi_i,\xi_r} & \frac{1}{2}c_4(\| w_j \|^2 + b_j^2) + \frac{1}{2}\eta_j^T\eta_j + c_5 e_i^T\xi_i + c_6 e_r^T\xi_r, \\ & \text{s.t. } C_j w_j + e_j b_j = \eta_j, \\ & (C_i w_j + e_i b_j) + \xi_i \geq e_i + R_i, \xi_i \geq 0, \\ & | C_r w_j + e_r b_j | + \xi_r \geq e_r + R_r, \xi_r \geq 0. \end{aligned}$$

(18) is formulated as follows:

$$\begin{aligned} \max_{\alpha} \quad & -\frac{1}{2}\gamma^T V(H^T H)^{-1}V^T\alpha + e_4^T\gamma, \\ \text{s.t.} \quad & 0 \leq \alpha \leq F \end{aligned} \tag{19}$$

$$\begin{aligned} \max_{\rho} \quad & -\frac{1}{2}\rho^T R(G^T G)^{-1}R^T\rho + e_4^T\rho, \\ \text{s.t.} \quad & 0 \leq \rho \leq F^*. \end{aligned} \tag{20}$$

for $\max_{\alpha} \; -\frac{1}{2}\gamma^T V(H^T H)^{-1}V^T\alpha + e_4^T\gamma$, s.t. $0 \leq \alpha \leq F$ (19), where $H = [A \quad e_1], G = [B \quad e_2], J = [C \quad e_3], V = [G \quad J], \gamma = [\alpha \quad \beta], e_4 = [e_2 + R_2 \quad e_3(1-\epsilon) + R_3]$ and $F = [c_1 e_2 \quad c_2 e_3]$ and the same calculation for $\max_{\rho} \; -\frac{1}{2}\rho^T R(G^T G)^{-1}R^T\rho + e_4^T\rho$, s.t. $0 \leq \rho \leq F^*$. (20).

In this way, the decision vectors $\vartheta_1 = [w_1; b_1]$ and $\vartheta_2 = [w_2; b_2]$ can be calculated by the formula:

$$\vartheta_1 = -(H^\top H)^{-1}(G^\top\alpha + J^\top\mu) \tag{21}$$

$$\vartheta_2 = -(H^\top H)^{-1}(G^\top\alpha + J^\top\mu) \tag{22}$$

The classification of new sample $z \in R^n$ is done based on the closest distance to the hyperplanes:

$$h(z) = \min_{1,2}\{\delta^1(z), \delta^2(z)\} \tag{23}$$

where $\delta^1(z) = | z^T\omega^{(1)} + b^{(1)} |, \delta^2(z) = | z^T\omega^{(2)} + b^{(2)} |$ with $|.|$ represents the normal distance of point z to the hyperplanes.

3 Proposed Model: Twin Support Vector Machine for Multi-class Classification (GBTSVM_MCC)

3.1 Model Overview

The Granular Ball Twin Support Vector Machine for Multi-class Classification (GBTSVM_MCC) introduces an innovative approach to handle multi-class problems by integrating Twin Support Vector Machines (TSVM) with Granular Ball Computing. This model extends the binary classification framework of GBTSVM to multi-class scenarios through two well-established decomposition strategies: One-vs-One (OvO) and One-vs-Rest (OvR). By utilizing granular balls as input units, defined by their center and radius, GBTSVM_MCC enhances noise resistance and computational efficiency, making it suitable for large-scale datasets.

In the OvO strategy, GBTSVM_MCC trains $K(K-1)/2$ binary GBTSVM classifiers for a dataset with K classes, each addressing a unique pair of classes (e.g., class i vs. class j), using GBs specific to those classes. The final classification for a new data point is determined by a voting mechanism, where the class with the most votes across all pairwise classifiers is selected. Conversely, the OvR strategy trains K binary GBTSVM classifiers, each distinguishing one class from all others (e.g., class i vs. not class i), again leveraging granular balls for the respective groups. The decision for a new data point is based on the highest confidence score or distance from the separating hyperplanes among the classifiers. These strategies ensure flexible and efficient multi-class classification tailored to diverse dataset structures.

To visually illustrate the two proposed methods, Figure 4 shows the comparison between the OvO and OvR methods when applied to the GBTSVM model. In the figure, the 3-class dataset is represented as GBs corresponding to each data class. For the OvO method, the non-parallel hyperplanes pass through the corresponding regions representing each binary classification pair. As for the OvR method, it is represented by non-parallel hyperplanes, in which one plane represents a class chosen as the positive class and the remaining planes represent the separation of that class from the set of all remaining classes.

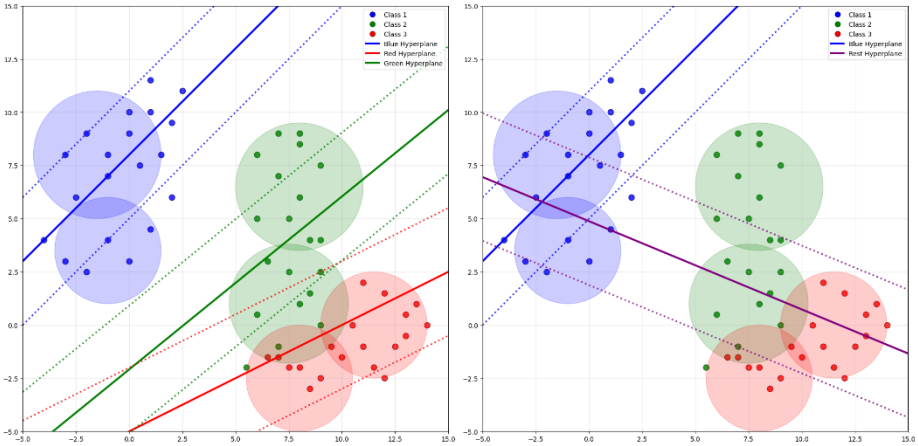

Fig. 4. Comparison of multi-class classification strategies: (Left) One-vs-One (OvO), (Right) One-vs-Rest (OvR). The colored GBs represent decision for three-class classification problem.

3.2 Granular Ball Generation

The first stage of the proposed model is to generate Granular Ball using the K-means algorithm. This process supports the generation of input data for the system. The calibration is performed based on two main parameters: purity threshold and minimum number in GB. The algorithm is implemented in the following steps.

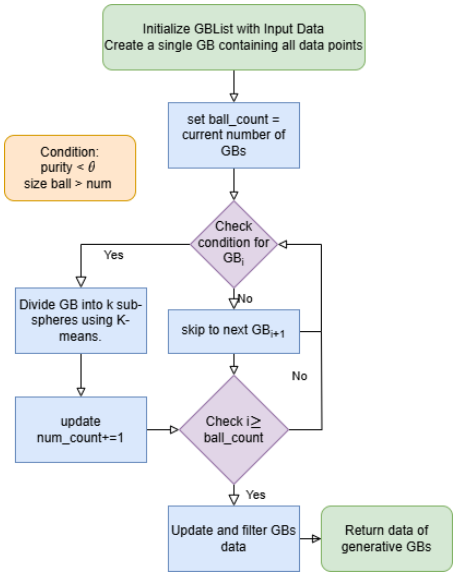

Fig. 5. Process of the GB generation.

1. Create an initial GBs containing a single GB, where this GB contains the entire set of data points.

2. Loop through each GB in the GBs, check the following two conditions:
   a. purity < θ
   b. number of points in the GB > num
   If either condition is not met, move on to the next GB.
3. If both the above conditions are true, perform the partition of the GB into k sub-GBs by applying the K-means algorithm. Then, update the GBs by replacing the original GB with the new sub-GBs and update the total number of GBs in the list.
4. Continue performing steps 2 and 3 until there are no more GBs in the GBs that satisfy the condition for partitioning.
5. Calculate and update the center, radius, label for each final GB, then filter and return the list of generated GBs.

The algorithm of granular ball generation is shown in detail as follows:

Algorithm 1 Granular Ball Generation

Input: Dataset $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^{N}$,where $\boldsymbol{x}_i \in R^d$ and label y_i
Purity threshold θ, minimum ball size n_{min}

Output: List of granular balls $\mathcal{G} = \{GB_j\} = \{(c_j, r_j, l_j)\}_{j=1}^{M}$

1: Initialize $\mathcal{G} \leftarrow \{GB_1 = \mathcal{D}\}$
2: **for** each GB_j in $\mathcal{G}$ **do**
3: **if** $purity(GB_j) <$ θ **and** $|GB_j| > n_{min}$ **then**
4: $\mathcal{G} \leftarrow (\mathcal{G}\{GB_j\}) \cup \mathrm{K-Means(GB_j, k)}$
5: **else**
6: **contiune**
7: **end for**
8: **for** each GB_j in $\mathcal{G}$ **do**
9: $c_j \leftarrow \frac{1}{|GB_j|}\sum_{\boldsymbol{x}_i \in GB_j} \boldsymbol{x}_i$
10: $r_j \leftarrow \frac{1}{|GB_j|}\sum_{\boldsymbol{x}_i \in GB_j} |\boldsymbol{x}_i - c_j|_2$
11: $l_j \leftarrow arg\ \max_{y} |\{\boldsymbol{x}_i \in GB_j : y_i = y\}|$
12: end for
13: $\mathcal{G} \leftarrow \{GB_j \in \mathcal{G} : purity(GB_j) \geq \theta \wedge |GB_j| \geq n_{min}\}$
14: return $\mathcal{G}$

Each granular ball is characterized by a centroid and a radius. The centroid of the granular ball, denoted by c_j, is computed as the average of all feature vectors belonging to that granular ball, forming a d-dimensional vector with d being the number of features. The radius r_j of the granular ball is defined as the average of the L2 (Euclidean) distances from each data point to the centroid cj. In the final step, each granular ball is labeled according to the class that is the majority in the set of data points belonging to that ball, i.e., the label l_j is the label that appears most frequently in the granular ball.

3.3 Multi-class Classification Strategy

OvO Method.

OvR Method.

3.4 Time Complexity and Algorithm

4 Experimental Results

4.1 Experimental setup

The experiment was conducted on a personal computer with an Intel Core i5-11300H CPU configuration at 3.20GHz, 24GB RAM and Windows 11 operating system. The programming environment used Python 3.11.7 with supporting libraries such as scikit-learn, numpy and pandas. The datasets were randomly divided into 80% for the training set and 20% for the test set, ensuring the same ratio between classes. The GBTSVM_MC model fine-tuning process was performed through a 5-fold cross-validation method on the training set. The main hyperparameters tuned are listed in **Table 1**:

Table 1. List of Parameters, Symbols, and Tuning Intervals

Parameters	Symbols	Tuning
Regularization coefficients	c	0.01, 0.025, 0.05, 0.075, 0.1, 0.25, 0.5
Margin tolerance limit	ϵ	0.01, 0.025, 0.05, 0.075, 0.1
Granular ball size	num	1, 2, 3, 4, 5
Purity threshold	pur	0.85, 0.895, 0.925, 0.97, 0.985

4.2 Dataset

1. To evaluate the performance of GBTSVM_MC, experiments were conducted on 10 multi-class datasets sourced from the UC Irvine Machine Learning Repository[1] and Kaggle[2]. These datasets were carefully selected from various domains and exhibit diversity in sample size, feature dimensionality, and number of classes. Detailed information about each dataset is summarized in **Table 2**. This diverse selection enables a comprehensive assessment of GBTSVM_MC's capabilities across different levels of complexity, including variations in the number of samples, features, and classes.

Table 2. Overview of multi-class datasets utilized in this experiments

Dataset	Samples	Features	Classes
Hayes-roth	132	5	3
Iris	150	4	3
Teaching Evaluation	151	5	3
Image-segmentation	210	19	7
Seeds	210	7	3
Glass	214	9	6
Ecoli	327	7	5
Dermatology	358	4	6
Balance	625	4	3

[1] https://archive.ics.uci.edu
[2] https://www.kaggle.com

Global Cancer	50000	13	5

Specially, the Global Cancer dataset obtained from Kaggle is utilized to assess the scalability and computational efficiency of GBTSVM_MC. This dataset is partitioned into three different subset sizes—1,000, 10,000, and 50,000 samples—to systematically evaluate the algorithm's performance and computational scalability across varying data volumes.

4.3 Result

Table 3.

Dataset		GBTSVM_MC(OvO) (c_1, c_2, ϵ_1, ϵ_2, num, pur)	GTBSVM_MC(OvR) (c_1, c_2, ϵ_1, ϵ_2, num, pur)	GB-TWKSVC (c_1, c_2, ϵ_1, ϵ_2, num, pur)	Twin-KSVC (c_1, c_2, ϵ_1, ϵ_2)
Hayes-roth	Accuracy (%)	74.19	**74.2**	53.57	57.06
	Time(s)	0.0352	0.0130	0.0732	0.0741
	Parameters	0.01, 0.01, 0.01, 0.01, 3, 0.895	0.01, 0.01, 0.01, 0.01, 3, 0.895	0.01, 0.01, 0.025, 0.025, 2, 0.97	0.01, 0.01, 0.075, 0.05
Iris	Accuracy (%)	86.67	93.33	**98.34**	95.32
	Time(s)	0.0189	0.0195	0.0332	0.0352
	Parameters	0.01, 0.01, 0.01, 0.01, 2, 0.985	0.01, 0.01, 0.01, 0.01, 2, 0.97	0.01, 0.01, 0.025, 0.025, 3, 0.985	0.01, 0.01, 0.025, 0.05
Teaching Evalua-tion	Accuracy (%)	62.5	68.75	**76.43**	69.35
	Time(s)	0.0261	0.0333	0.0525	0.0543
	Parameters	0.01, 0.01, 0.01, 0.01, 3, 0.97	0.01, 0.01, 0.01, 0.01, 3, 0.925	0.01, 0.01, 0.025, 0.025, 3, 0.97	0.01, 0.01, 0.025, 0.025
Image-segmen-tation	Accuracy (%)	**92.85**	90.47	90.43	89.65
	Time(s)	0.0840	0.0530	0.3967	0.5487
	Parameters	0.01, 0.01, 0.01, 0.01, 1, 0.97	0.01, 0.01, 0.01, 0.01, 2, 0.925	0.01, 0.01, 0.05, 0.01, 2, 0.925	0.01, 0.01, 0.025, 0.025
Seeds	Accuracy (%)	88.09	**100**	98.23	95.12
	Time(s)	0.0312	0.0266	0.0560	0.0624
	Parameters	0.01, 0.01, 0.01, 0.01, 2, 0.97	0.01, 0.01, 0.01, 0.01, 2, 0.97	0.01, 0.01, 0.025, 0.01, 3, 0.985	0.01, 0.01, 0.05, 0.025
Glass	Accuracy (%)	82.14	**84.52**	75.34	67.93
	Time(s)	0.1615	0.0525	0.3205	0.4710
	Parameters	0.01, 0.01, 0.01, 0.01, 2, 0.925	0.01, 0.01, 0.01, 0.01, 2, 0.985	0.01, 0.01, 0.01, 0.01, 3, 0.925	0.01, 0.01, 0.05, 0.025
Ecoli	Accuracy (%)	82.53	82.52	**92.54**	87.32
	Time(s)	0.2992	0.0849	0.8083	1.728
	Parameters	0.01, 0.01, 0.01, 0.01, 3, 0.85	0.01, 0.01, 0.01, 0.01, 3, 0.85	0.01, 0.01, 0.025, 0.025, 2, 0.085	0.01, 0.01, 0.01, 0.025
Derma-tology	Accuracy (%)	**98.5**	96.26	91,75	81.21
	Time(s)	0.1075	0.0708	0.4103	0.719
	Parameters	0.01, 0.01, 0.01, 0.01, 4 0.985	0.01, 0.01, 0.01, 0.01, 4, 0.895	0.01, 0.01, 0.01, 0.01, 2, 0.97	0.01, 0.01, 0.075, 0.05
Balance	Accuracy (%)	**94.21**	92.48	89.62	87.52

	Time(s)	**0.0533**	0.0733	0.1405	0.2676
	Parameters	0.01, 0.01, 0.01, 0.01, 4, 0.97	0.01, 0.01, 0.01, 0.01, 1, 0.85	0.01, 0.01, 0.01, 0.01, 2 0.97	0.01, 0.01, 0.01, 0.01
Global Cancer – 1K	Accuracy (%)	**63.07**	62.24	60.16	57.58
	Time(s)	0.0395	0.0505	0.0518	0.2628
	Parameters	0.01, 0.01, 0.01, 0.01, 4, 0.97	0.01, 0.01, 0.01, 0.01, 4, 0.97	0.01, 0.01, 0.01, 0.01, 4, 0.97	0.01, 0.01, 0.01, 0.025
Global Cancer – 10K	Accuracy (%)	**61.53**	60.85	57.38	54.65
	Time(s)	0.8314	1.5107	1.6211	3.2571
	Parameters	0.01, 0.01, 0.01, 0.01, 3, 0.985	0.01, 0.01, 0.01, 0.01, 2, 0.97	0.01, 0.01, 0.01, 0.01, 2, 0.925	0.01, 0.01, 0.05, 0.05
Global Cancer – 50K	Accuracy (%)	50.28	**50.71**	49.23	42.78
	Time(s)	80.2156	158.9295	155.5041	186.4782
	Parameters	0.01, 0.01, 0.01, 0.01, 1, 0.85	0.01, 0.01, 0.01, 0.01, 2, 0.895	0.01, 0.01, 0.01, 0.01, 2, 0.895	0.01, 0.01, 0.1, 0.1

4.4 Discussion

5 Conclusion

Acknowledgments. A third level heading in 9-point font size at the end of the paper is used for general acknowledgments, for example: This study was funded by X (grant number Y).

Disclosure of Interests. It is now necessary to declare any competing interests or to specifically state that the authors have no competing interests. Please place the statement with a third level heading in 9-point font size beneath the (optional) acknowledgments, for example: The authors have no competing interests to declare that are relevant to the content of this article. Or: Author A has received research grants from Company W. Author B has received a speaker honorarium from Company X and owns stock in Company Y. Author C is a member of committee Z.

References

2. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
3. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016)
4. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
5. Author, F.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
6. LNCS Homepage, http://www.springer.com/lncs, last accessed 2023/10/25