A Multi-subject Classification Algorithm Based on SVM Geometric Interpretation

Chi Tang

College of All Media, Liaoning Vocational University of Technology, Jinzhou 121013, China
15604060972@163.com

Abstract. A new multi-subject classification algorithm based on support vector machine(SVM) is proposed. For each class of training samples, a minimum convex shell surrounding as many samples as possible is constructed in the feature space by using soft SK algorithm, and finally the multi-subject classifier composed of multiple convex shells is obtained. For the sample to be classified, its classes are determined according to the convex hulls in which it are located. If it is not in any convex hull, firstly, the membership degree is determined by the distance that from it to the centroid of each class sample, and then its class to which it belongs is determined according to the membership degree. The classification experiments are carried out on the standard dataset Reuters 21578, and the classification performance is compared with hyperellipsoid method and hypersphere method. The experimental results show that compared with hyperellipsoid method and hypersphere method, the proposed algorithm can ensure the inheritability of the classifier and the classification accuracy is significantly improved, which effectively solves the influence of sample distribution shape on classification performance.

Keywords: Multi-subject classification, Convex hull, Schlesinger-Kozinec algorithm, Support vector machine

1 Introduction

1.1 A Subsection Sample

The multi-subject is when a single piece of data is associated with multiple subjects. The multi-subject classification is to construct a multi-label classifier by training the given multi-label sample set, and to effectively predict the class label set of unknown class samples. Due to the rapid increase of multi-subject classification application field, it has attracted people's attention, and has become a research hotspot in the field of machine learning[1-6].

The main research achievements of multi-subject classification include SVM method[7,8], neural network method[9], K-nearest neighbor method[10], decision tree method[11,12] and so on. Most of these methods transform multi-subject classification problems into multiple binary classification problems. The training speed is slow when the data set is large. The classification efficiency is low when the number of labels is large. The classification accuracy is affected when the data set is unbalanced. In addition, the scalability and inheritance of these algorithms are poor. When a new class sample is added, the classifier needs to be retrained.

Convex hull technique is an effective method to solve the problem of multi-subject class classification. Literature[13] proposes a hypersphere SVM multi-subject classification algorithm, which uses the optimal hypersphere to separate all kinds of samples to the maximum extent, and determines the category of samples to be classified by judging the hypersphere where the samples are located. However, this algorithm is only suitable to the case where each class of samples is distributed in hypersphere shape. Literature[14] proposes a hyperellipsoid SVM multi-subject classification algorithm, which uses the optimal hyperellipsoid to separate all kinds of samples to the maximum extent, and determines the category of samples to be classified by judging the hyperellipsoid where the samples are located. However, this algorithm is only suitable for the case where each class of samples is distributed in hyperellipsoid shape. Literature[15] proposes a multi-subject classification algorithm based on hypercuboid, which uses the optimal hypercube to separate all kinds of samples to the maximum extent, and determines the category of the samples to be classified by judging the hypercube where the samples are located. However, this algorithm is only applicable to the case that each class of samples has a hypercube shape distribution.

SVM is a binary classification model. The intuitive geometric interpretation of SVM is to realize the separation of convex shells of two classes of training samples at the maximum boundary, The optimal hyperplane is the hyperplane that maximizes the margin. Soft Schlesinger-Kozinec (SK) algorithm[16] is an algorithm to calculate the optimal hyperplane, which has the advantages of high calculation accuracy and easy application. Based on geometric interpretation of SVM and SK algorithm, this paper presents a multi-subject classification algorithm.

In the second part of this paper, the geometric interpretation of SVM and SK algorithm are introduced. In the third part, convex shell construction algorithm and multi-subject classification algorithm are described. The fourth part gives the experimental results and analysis on the standard, and finally draws the conclusion.

2 SVM Geometry Interpretation and SK Algorithm

2.1 SVM Geometry Interpretation

Let a linearly separable training set $X = \{\mathrm{x}_\mathrm{i}, y_i\}_{i=1}^{l}$ be known, where $x_i \in R^n, y_i \in \{1, -1\}$, and l is the number of training samples. Make convex shells of positive and negative class sets respectively, find the closest points c and d of these two convex shells, the vertical bisector H of line segment cd is the SVM that correctly divides the training set X into two parts, as shown in Fig.1.

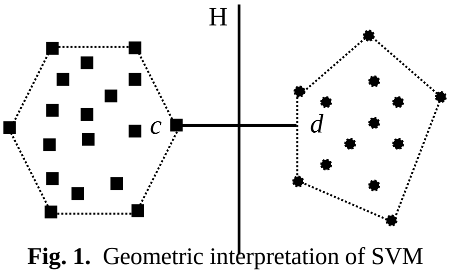

Fig. 1. Geometric interpretation of SVM

2.2 SVM Geometry Interpretation

SK algorithm is a typical algorithm for solving the nearest point between convex shells. The algorithm is described as follows:

Input： X, Y, ε , where X and Y are two classes of sample sets, and ε is the stop precision.

Output： $f(x) = w^* \cdot x + b$

Step1: Pick $x^* \in X, y^* \in Y$.

Step2: Calculate mx according to Eq.(1), my according to Eq.(2), and m according to Eq.(3). If $\left\| x^* - y^* \right\| - m < \varepsilon$, go to Step 4, otherwise go to Step 3.

$$mx = min\{ \frac{(x_i - y^*)\cdot(x^* - y^*)}{\left\| x^* - y^* \right\|} \mid x_i \in X \} \tag{1}$$

$$my = min\{ \frac{(y_j - x^*)\cdot(y^* - x^*)}{\left\| x^* - y^* \right\|} \mid y_j \in Y \} \tag{2}$$

$$m = min\{ mx, my \} \tag{3}$$

Step3: If $mx \le my$, λ is calculated according to Eq.(4), and then x^* is updated according to Eq.(5). Otherwise, first calculate μ according to Eq.(6), and then update y^* according to Eq.(7). Go to Step 2.

$$\lambda = min\{ 1, \frac{(x^* - y^*)\cdot(x^* - x_t)}{\left\| x^* - x_t \right\|^2} \} \tag{4}$$

Where, x_t is the sample in sample set X with the smallest subscript satisfying Eq.(2).

$$\boldsymbol{x^* = x^*(1-\lambda)+\lambda x_t} \tag{5}$$

$$\boldsymbol{\mu = min\{1, \frac{(y^* - x^*)\cdot(y^* - y_t)}{\left\|y^* - y_t\right\|^2}\}} \tag{6}$$

Where, $\boldsymbol{y_t}$ is the sample in sample set $\boldsymbol{Y}$ with the smallest subscript satisfying Eq.(1).

$$\boldsymbol{y^* = y^*(1-\mu)+\mu y_t} \tag{7}$$

Step4:Comput $f(x) = w^* \cdot x + b$, where $w^* = x^* - y^*$ and $b = (\| y^* \|^2 - \| x \|^2)/2$.

The SK algorithm first takes a sample $\boldsymbol{x^*}$ and $\boldsymbol{y^*}$ from $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively, and then finds the nearest point $\boldsymbol{x_t}$ or $\boldsymbol{y_t}$ to the vector $\boldsymbol{x^*}$ - $\boldsymbol{y^*}$, and the nearest point is obtained by calculating $\boldsymbol{mx}$ and $\boldsymbol{my}$. Taking $\boldsymbol{mx}$ as an example, suppose we find that x_t meets the conditions, i.e.

$$\boldsymbol{mx = \frac{(x_t - y^*)\cdot(x^* - y^*)}{\left\|x^* - y^*\right\|}} \tag{8}$$

Let θ represent the Angle between two vectors, and simplify Eq.(8) to get Eq.(9) according to the definition of the inner product.

$$\boldsymbol{mx = \frac{(x_t - y^*)\cdot(x^* - y^*)}{\left\|x^* - y^*\right\|} = \frac{\left\|x_t - y^*\right\|\cdot\left\|x^* - y^*\right\|\cdot cos\theta}{\left\|x^* - y^*\right\|} = \left\|x_t - y^*\right\|\cdot \mathbf{cos}\theta} \tag{9}$$

The geometric meaning of $\boldsymbol{mx}$ can be obtained from Eq.(9), as shown in Fig.2

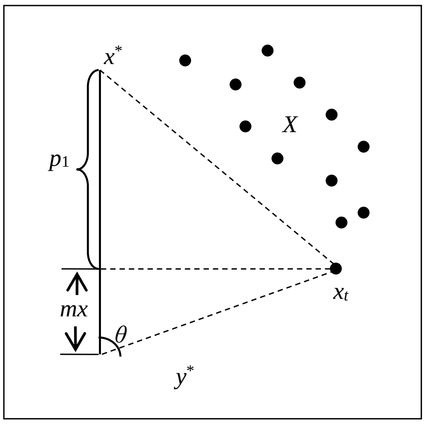

Fig.2. The geometric meaning of mx

As can be seen from Fig.2, mx is the projection length of the vector x_t - y^* on the vector x^* - y^*, where x_t is the point with the smallest projection length in the set of X. θ is the angle between the vector x_t - y^* and x^* - y^*. In fact, x_t is the point in the set of X that gives $x^* - x_i$ the greatest projection length in the x^* - y^* direction (i.e. p_1 in Fig. 2). It can be seen that the purpose of Step 2 in the algorithm is to find the point x_t or y_t in X and Y that makes the projection on x^* - y^* the largest, and select the larger one to verify the stop condition. If the condition is satisfied, the algorithm stops and the classification hyperplane $f(x)$ is obtained. Otherwise, perform local adjustments and updates. If $m = mx$, then y^* is fixed and $x^*(x^* = (1-\lambda)x^* + \lambda x_t)$ is updated. If $m = my$, then x^* is fixed and $y^*(y^* = (1-\mu)y^* + \mu y_t)$ is updated. The value λ (or μ) is determined according to Eq. (4) (or Eq.(6)) to ensure that the distance between the new x* and y* is minimal.

3 Multi-subject Classification Algorithm

3.1 Convex Shell Training Algorithm

Let a multi-subject sample set $X = \{x_i, E_i\}_{i=1}^{l}$ be given, where l is the number of samples, $x_i \in R^n$, $E_i = \{y_i^j\}_{j=1}^{p}$, $y_i^j \in \{1,2,\cdots,N\}$, N is the number of classes in sample set X, and $p(p \le N)$ is the number of classes of the sample x_i.

Let X_m be the subset of samples in X that contains the m-th $m(m=1,2,\cdots,N)$, the algorithm for constructing convex shells in feature space is as follows:

Step 1: Use the m-th class samples X_m（$1 \le m \le N$） as positive class and other samples as negative class.

Step 2: Use SK algorithm to find sample set C_m in $X - X_m$ that is not in the convex shell of the m-th class.

Step 3: Find the point closest to the convex shell of the m-th class in C_m and construct a hyperplane, which will cut off some points in C_m. Then, find the point closest to the convex shell of the m-th class from the remaining points of C_m, construct the hyperplane, and cut off some points in C_m again. Repeat the process until C_m is empty.

Step 4: Construct a convex shell $H_m = \{H_m^1, H_m^2, \cdots, H_m^{N_m}\}$ of the m-th class with the resulting hyperplanes, where, N_m is the number of hyperplanes in the convex shell of the m-th class, and H_m^i is the i-th($1 \le i \le N_m$) hyperplane of the convex shell of the m-th class.

Step 5: For each class of training sample, repeat Step 1 to Step4 to obtain N convex shells.

3.2 Multi-subject Classification Algorithm

For classified sample x, determine whether x is in the i-th convex shell $H_m = \{H_m^1, H_m^2, \cdots, H_m^{N_m}\}$ according to Eq.(10). If $f_{H_m^i}(x) \ge 0$ for every hyperplane

$H_m^i (1 \le i \le N_m)$, then x in the m-th convex shell, x belong to the m-th class. If x is not in any convex shell, its class is determined by its distance to the centroid of each class of sample.

$$f_{H_m^i}(x) = w_m^i.x + b_m^i \quad (i = 1,2,\cdots N_m) \tag{10}$$

The classification process is described in detail as follows:

Step 1: Determine whether x is in the m-th $(m = 1,2,\cdots,N)$ convex shell H_m according to Eq.(10).

Step 2: If x is not in any convex shell, go to Step 4, otherwise go to Step 3.

Step3: $class = \{i \mid \text{x is in the i - th convex shell}, 1 \le i \le N\}$,go to Step5.

Step 4: Firstly, the centroid of each class of sample is calculated according to Eq.(11), and then the distances between x and these centroids are calculated according to Eq.(12), finally, the class of x is determined according to Eq.(13).

$$\alpha_m = \frac{1}{l_m}\sum_{i=1}^{l_m}\phi(x_m^i) \tag{11}$$

$$[d_m(x)]^2 = \left\|\phi(x) - a_m\right\|^2 \tag{12}$$

$$class = \min_m d_m(x) \tag{13}$$

Step5: Stop.

4 Experimental Results and Analysis

The standard data set Reuters 21578 was used for performance analysis in the experiment, from which 665 texts were selected, with a total of 6 categories, and the maximum number of classes of a text was 3. Among them, 431 were used as training sample set, and 234 were used as test sample set (see Table 1). The text is preprocessed to form a high-dimensional word space vector, and the information gain method is used to reduce the feature dimension, and the weight of the word is calculated using tf-idf.

Table 1. Training set and testing set

Name	oat	corn	cotton	soybean	wheat	rice
Code	1	2	3	4	5	6
Training	9	168	44	79	204	44
Testing	5	84	22	40	101	23

In the experiment, macro-averaging precision, recall, F_1 value and micro-averaging precision, recall, F_1 value are used as evaluation indicators.

$$precision\,(P) = N_c / N_a \tag{14}$$

$$recall\,(R) = N_c / N_r \tag{15}$$

$$F_1 = (2 * P * R)/(P + R) \tag{16}$$

Where, N_c is the number of classes in which the sample is correctly classified, N_a is the number of classes in which the sample is classified, and N_r is the number of classes actually existing in the sample.

$$averaging\ precision\,(AP) = (\sum P)/n \tag{17}$$

$$averaging\ recall\,(AR) = (\sum R)/n \tag{18}$$

$$averaging\ F_1(AF) = (\sum F_1)/n \tag{19}$$

When n is the total number of test samples, they are called macro-averaging precision(MAAP), macro-averaging recall(MAAR) and macro-averaging F_1 (MAAF) respectively. When n is the number of samples with the same number of classes, they are called micro-averaging precision(MIAP), micro-averaging recall(MIAR) and micro-averaging F_1 (MIAF) respectively.

Literature[14] gives the performance comparison between hypersphere method and hyperellipsoid method, and literature [15] gives the performance comparison between hypersphere method and hypercube method. Among the three methods, hyperellipsoid method has the best classification performance. In order to compare the performance of the method, the proposed method, hypersphere method and hyperellipsoid method are used for classification experiments on the data set respectively. The experimental running environment is I5-6500 CPU 3.20GHz, 8GB memory, windows8.1 operating system.

Table 2 shows the comparison of MAAP, MAAR and MAAF of the proposed method, hypersphere method and hyperellipsoid method. Table 3 shows the comparison of MIAP, MIAR and MIAF of the proposed method, hypersphere method and hyperellipsoid method. Table 4 shows the comparison of training time and testing time of the proposed method, hypersphere method and hyperellipsoid method.

Table 2. Comparison of MAAP, MAAR and MAAF

Algorithm	MAAP(%)	MAAR(%)	MAAF(%)
Hypersphere method	78.38	77.92	77.52
Hyperellipsoid method	82.34	80.96	81.84
Proposed method	85.68	88.55	86.33

Table 3. Comparison of MIAP, MIAR and MIAF

Algorithm	The number of subjects	MIAP(%)	MIAR(%)	MIAF(%)
Hypersphere method	1	71.34	73.91	72.14
	2	83.33	55.32	63.93
	3	100.00	50.00	65.00
Hyperellipsoid method	1	75.78	78.95	76.68
	2	87.12	63.33	73.93
	3	66.67	66.67	66.67
Proposed method	1	82.25	84.88	83.45
	2	91.22	82.76	78.66
	3	88.89	75.32	76.42

Table 4. Comparison of training time and testing time

Algorithm	Training (ms)	Testing (ms)
Hypersphere method	128	119
Hyperellipsoid method	159	108
Proposed method	176	131

It can be seen from the experimental results that the precision and recall of the proposed method are significantly improved compared with hyperellipsoidal method and hypersphere method. The reason is that the proposed method is not affected by the shape of the sample distribution, and the shape of the convex shell is similar to that of the sample distribution. The space enclosed by a convex shell is smaller than the space enclosed by a hyperellipsoid. The training speed and classification speed of the proposed method are slightly lower than that of hyperellipsoid method and hypersphere method, because there is only one classifier of each class in hyperellipsoid method and hypersphere method, and only one class of samples participate in the training, while the classifier of each class in the proposed algorithm is composed of multiple sub-classifiers and all samples participate in the training.

5 Conclusion

Based on geometric analysis and SK algorithm of support vector machine, a multi-subject classification algorithm of support vector machine is proposed. The convex shell training algorithm and multi-subject classification algorithm are described in detail, and the classification experiments are carried out on the standard data set Reuters 21578. The experimental results show that the proposed method can effectively avoid the influence of sample distribution shape on the classification accuracy, and the classification accuracy of the proposed method is significantly improved compared with hyperellipsoid method and hypersphere method, the proposed method has good scalability and inheritance, that is, if a new class sample is added, the original classifier does not need to be retrained. The further research work is how to quickly construct the smallest convex shell enclosing a class of samples, and make the hyperplane forming the convex shell as few as possible, so as to improve the training speed and classification speed.

Acknowledgements. This study was funded by the Education Committee Project of Liaoning Province, China (grant number LJKMZ20221883).

References

[1] Zhang, M. L., Zhou, Z, H.: A review on mutil-label learning algorithms. IEEE Transactions on Knowledge & Data Engineering. 26, 1819-1837(2014)

[2] Al-salemi B., Ayob M., Noah S.:Feature ranking for enhancing boosting-based multi-label text categorization. Expert Systems with Applications. 113, 531-543(2018)

[3] Liu, Y., Wen K., Gao Q., et al.:SVM based multi-label learning with missing labels for image annotation. Pattern Recognition. 78,307-317(2018)

[4] Xu, X. S., Jiang, Y., Peng. L, et al.:Ensemble approach based onconditional random field for multi-label image and video annotation. In: Proceedings of the 19th International Conference on Multimedea. Scottsdale, AZ, USA: ACM. pp.1377-1380(2011)

[5] Sun, L., Zu, C., Shao, W., et al.:Reliability-based robust multi-atlas label fusion for brain MRI segmentation. Artificial Intelligence in Medicine. 96,12-24(2019)

[6] Gong, K. L., Zhai, T. T., Tang, H. C.:An online active learning algorithm for multi-label classification. Journal of Shandong University(Engineering Science). 52(2), 80-88(2022)

[7]Feng, P., Qin, D., Ji, P., et al.:Multi-label learning algorithm with SVM based association. High Technology Letters. 25(1), 97-104(2019)

[8]Sun, Z., Liu, X., Hu, K.,et al.:An Efficient Multi-Label SVM Classification Algorithm by Combining Approximate Extreme Points Method and Divide-and-Conquer Strategy.IEEE Access. 8, 170967-170975(2020)

[9]Zhuang, N., Yan, Y., Chen,S.: Multi-label learning based deep transfer neural network for facial attribute classification. Pattern Recogn. 80,225–240(2018)

[10]Zhang, M. L., Zhou, Z. H.: ML-KNN: a lazy learning approach to multi-label learning. Pattern Recognition. 40(7), 2038-2048(2007)

[11] Vens, C., Struyf, J., Schietgat, L.,et al.:Decision trees for hierarchical multi-label classification. Machine Learning. 73(2), 185-214(2008)

[12] Guan, X. Q., Liang, J. Y., Qian, Y. H., et al.:A multi-view OVA model based on decision tree for multi-classification tasks. Knowledge-Based Systems. 138, 208-219(2017)

[13] Qin, Y. P., Wang, X. K., Li, X. N., et al.:Study on multi- class text classification algorithm based on hyper- sphere support vector machines.Computer Engineering and Applications. 44(19), 166- 168(2008)

[14] Qin, Y. P., Wang, Y,, Lun. S, X.:Multi-label Text Classification Algorithm Based on Hyper Ellipsoidal SVM. Computer Science. 40(11A), 98-100(2013)

[15] Qin. Y. P., Cheng, X. Y., Li, X. N., etal.:An effective multi-label classification algorithm based on hypercube. In: Proceedings of the Sixteenth International Conference on Intelligent Computing, Bari, Italy, pp.476–483(2020).

[16] Franc V., Hlaváč V.:An iterative algorithm learning the maximal margin classifier. Pattern Recognition. 36(9),1985-1996(2003)