

Research Article

A New SVM Multiclass Incremental Learning Algorithm

Yuping Qin,¹ Dan Li,² and Aihua Zhang¹

¹ College of Engineering, Bohai University, Jinzhou 121013, China

² College of Mathematics and Physics, Bohai University, Jinzhou 121013, China

Correspondence should be addressed to Yuping Qin; qlq88888@sina.com

Received 14 July 2014; Accepted 23 August 2014

Academic Editor: Xudong Zhao

Copyright © 2015 Yuping Qin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new support vector machine (SVM) multiclass incremental learning algorithm is proposed. To each class training sample, the hyperellipsoidal classifier that includes as many samples as possible and pushes the outlier samples away is trained in the feature space. When the new samples are added to the classification system, the algorithm reuses the old classifiers that have nothing to do with the new sample classes. To be classified sample, the Mahalanobis distances are used to decide the class of classified sample. If the sample point is not surrounded by any hyperellipsoidal or is surrounded by more than one hyperellipsoidal, the membership is used to confirm its class. The experimental results show that the algorithm has higher performance in classification precision and classification speed.

1. Introduction

SVMs [1], as a new machine learning method based on statistical learning theory, deliver state-of-the-art performance in real world pattern recognition [2, 3] and data mining applications such as text categorization, hand-written character recognition, image classification, and bioinformatics, even into control field [4, 5]. However, the general training method of SVMs will not work when the amount of training samples is too large to be put into the RAM of computer. In order to solve this problem and improve the speed of training SVMs, the incremental learning algorithm has become one of the key techniques for training SVMs on large data sets, especially on multiclass problem.

When using SVMs to deal with multiclassification problems, the most popular four approaches are one-against-one [6], one-against-rest [7], DAGSVM [8], and binary tree SVM [9–11]. Some incremental learning algorithms have been proposed, such as Batch SVM [12, 13], online recursive algorithm [14], divisional training SVMs algorithm [15], and fast incremental learning algorithm [16]. However, these multiclassification approaches are based on binary classifiers, when new samples were added to the classification system, the whole model of the classifier must be retrained. Reference [17] proposed a class-incremental learning algorithm. The algorithm reuses the old models of the classifier, and only

one binary classifier is trained when a new class comes. But it is not suitable for large data set, and new sample set can not include the old class samples. Reference [18] proposed a multiclass incremental learning algorithm based on hypersphere SVMs (HSSVMIL). The algorithm reuses the old models of the classifier, and it is suitable to class-incremental learning and old class sample-incremental learning at the same time. But the distribution of every class sample must be hypersphere shape and the density of the samples is higher in the feature space. Otherwise, the precision of the algorithm is lower. For the disadvantage, [19] proposed a multiclass incremental learning algorithm based on hyper ellipsoidal (HEIL), but the algorithm does not consider the influence of the outlier samples.

In this paper, a Mahalanobis hyperellipsoidal SVM multiclass incremental learning algorithm (MSVMIL) is proposed. To every class sample, the smallest hyperellipsoidal that contains as much samples as possible and pushes the outlier samples away is trained in the feature space. Mahalanobis distances are used to confirm the class of classified sample.

This paper is organized as follows. In Section 2, a review of hypersphere SVM is given. In Section 3, a new multiclass incremental learning algorithm is discussed in detail. In Section 4, experimental results are given on Reuters 21578. Finally, conclusion is outlined.

2. Hyperellipsoidal Support Vector Machine

Given a set of training sample of a class $X = \{x_i\}_{i=1}^N$, where $x_i \in R^d$. Let M be a $d \times N$ sample matrix. Training a hyperellipsoidal $E(a, R)$ in the feature space, where a is the center of the hyperellipsoidal and R is the radius of the hyperellipsoidal. The hyperellipsoidal should contain most of the samples and the radius R is as small as possible. If there are not remote points, then the hyperellipsoidal will contain all samples. If there are remote points, then some samples outside the hyperellipsoidal are allowed, training the smallest hyperellipsoidal that contains most of the samples. When we are uncertain whether there are remote points, nonnegative slack variables ξ_i ($i = 1, 2, \dots, N$) are introduced to allow some samples outside the hyperellipsoidal. Using the method is similar to finding optimal hyperplane to obtain the smallest hyperellipsoidal [19–21]. The formulation is as follows:

$$\begin{aligned} \min_{a, R, \xi_i} \quad & R^2 + C \sum_{i=1}^N \xi_i \\ \text{s.t.} \quad & (x_i - a)^T \Sigma^{-1} (x_i - a) \leq R^2 + \xi_i \\ & \xi_i \geq 0, \quad i = 1, 2, \dots, N, \end{aligned} \quad (1)$$

where C is used to compromise the number of noises out of hyperellipsoidal and the radius of hyperellipsoidal, Σ is covariance matrix of the samples.

To solve the optimization problem above, one can construct the Lagrange function as follows:

$$\begin{aligned} L(R, a, \beta, \gamma, \xi_i) \\ = R^2 + C \sum_{i=1}^N \xi_i \\ - \sum_{i=1}^N \alpha_i \{R^2 + \xi_i - (x_i - a)^T \Sigma^{-1} (x_i - a)\} - \sum_{i=1}^N \beta_i \xi_i, \end{aligned} \quad (2)$$

where α_i ($\alpha_i \geq 0$) and β_i ($\beta_i \geq 0$) are the Lagrange multipliers.

According to the Kuhn-Tucker Theorem (KKT) in optimization theory, the following conditions are satisfied:

$$\begin{aligned} \frac{\partial L}{\partial R} = 2R \left(1 - \sum_{i=1}^N \alpha_i \right) = 0 &\implies \sum_{i=1}^N \alpha_i = 1, \\ \frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 &\implies 0 \leq \alpha_i \leq C, \\ \frac{\partial L}{\partial a} = - \sum_{i=1}^N 2\alpha_i (x_i - a) = 0 &\implies a = \sum_{i=1}^N \alpha_i x_i. \end{aligned} \quad (3)$$

Substituting (3) into (2), the dual optimal problem is obtained as follows:

$$\begin{aligned} \max_{\alpha_i \geq 0} \quad & \sum_{i=1}^N \alpha_i x_i^T \Sigma^{-1} x_i - \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j x_i^T \Sigma^{-1} x_j \\ \text{s.t.} \quad & \sum_{i=1}^N \alpha_i = 1 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N. \end{aligned} \quad (4)$$

The kernel form of (4) is as follows:

$$\begin{aligned} \max_{\alpha_i \geq 0} \quad & \sum_{i=1}^N \alpha_i k(x_i, M) Q^T \Omega^{-2} Q k(x_i, M)^T \\ & - \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j k(x_i, M) Q^T \Omega^{-2} Q k(x_j, M)^T \\ \text{s.t.} \quad & \sum_{i=1}^N \alpha_i = 1 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N, \end{aligned} \quad (5)$$

where $K(\cdot)$ is kernel function, via $K(x_i, x_j) = g(x_i)^T g(x_j)$.

The examples that lie outside or on the margin are the corresponding α_i nonzero. These examples are called support vectors.

The center of the smallest hyperellipsoidal a can be obtained by (6). Consider

$$a = \sum_i \alpha_i g(x_i). \quad (6)$$

The Mahalanobis distance from sample x to the center of hyperellipsoidal is as follows:

$$\begin{aligned} md(x, a) &= \sqrt{(g(x) - a)^T \Sigma^{-1} (g(x) - a)} \\ &= \left(N \left(k(x, M) - \sum_{i=1}^N \alpha_i k(x_i, M) \right) \right. \\ &\quad \left. \times Q^T \Omega^{-2} Q \left(k(x, M) - \sum_{i=1}^N \alpha_i k(x_i, M) \right) \right)^{1/2}. \end{aligned} \quad (7)$$

The radius of the smallest hyperellipsoidal R can be determined by (8), via KKT conditions. Consider

$$\begin{aligned} md(x, a) &< R \quad \alpha_i = 0, \\ md(x, a) &= R \quad 0 < \alpha_i < C, \\ md(x, a) &> R \quad \alpha_i = C. \end{aligned} \quad (8)$$

Remark 1. Mahalanobis distance denotes the distance between a data point and multivariate space centroid, that is, overall mean value.

3. Multiclass Incremental Learning Algorithm

Given a set of training samples $X = \{x_i, T_i\}_{i=1}^l$ and kernel function $K(\cdot)$. Where l is the number of samples, $x_i \in R^n$, $T_i \in \{1, 2, \dots, N\}$, N is the number of classes. $K(\cdot)$ corresponds to inner product in feature space, namely, $K(x_i, x_j) = g(x_i)^T g(x_j)$.

Assume that X_m ($m = 1, 2, \dots, N$) is a subset of A , and all samples in X_m belong to the m th class. For every subset X_m , train the smallest hyperellipsoidal $E(a_m, R_m)$ in feature space, where a_m is the center of the hyperellipsoidal, R_m is the radius of the hyperellipsoidal. SV_m is the support vector set.

If there is a new sample set NX is added to the old classification system, where NX_m ($m = 1, 2, \dots, N, N + 1, \dots, NN$) is a subset of NX and all samples of NX_m belong to the m th class. The multiclass incremental learning algorithm based on hyperellipsoidal SVM is described in detail as follows.

Step 1. For every subset NX_m ($m = N + 1, N + 2, \dots, NN$), let $X_m = NX_m$, training the smallest hyperellipsoidal $E(a_m, R_m)$ with X_m in the feature space and saving support vectors to SV_m .

Step 2. For every subset NX_m ($q = 1, 2, \dots, N$), if $NX_m \neq \Phi$, then $X_m = NX_m + SV_m$, retraining the hyperellipsoidal $E(a_m, R_m)$ with X_m in the feature space and refining SV_m .

For the classified sample x , compute the Mahalanobis distances $md(x, a_m)$ ($m = 1, 2, \dots, NN$) according to (7) in the feature space.

If there is not hyperellipsoidal containing the sample point, via $md(x, a_m) > R_m$ ($m = 1, 2, \dots, NN$), then compute the membership that the sample belongs to the m th class according to (9) and then confirm the class of the sample according to (10). Consider

$$r_m = \frac{R_m}{md(x, a_m)}, \quad (9)$$

$$\text{class} = \arg \min_i r_i. \quad (10)$$

If there are no less than two hyperellipsoidals containing the sample point, compute the membership that the sample belongs to the m th class according to (11) for $md(x, a_m) \leq R_m$ firstly and then confirm the class of the sample according to (10). One has

$$r_m = 1 - \frac{md(x, a_m)}{R_m}. \quad (11)$$

For the classified sample x , the multiclass classification algorithm is described as follows.

Step 1. Compute $md(x, a_m)$ ($m = 1, 2, \dots, NN$) according to (7).

Step 2. If the sample point is contained by only one hyperellipsoidal $E(a_m, R_m)$, the sample belongs to the m th class; go to Step 5. Otherwise, go to Step 3.

TABLE 1: Training set and testing set.

Class	Acq	Eran	Grain	Crude	Trade
Class code	1	2	3	4	5
Original					
Training	403	134	0	0	0
Testing	201	67	0	0	0
1st increment					
Training	200	98	82	0	0
Testing	100	49	41	0	0
2nd increment					
Training	101	68	44	55	0
Testing	51	34	22	31	0
3rd increment					
Training	100	42	0	33	50
Testing	50	21	0	11	25

Step 3. If the sample point is not contained by any hyperellipsoidal, compute the membership that the sample belongs to the m th ($m = 1, 2, \dots, NN$) class according to (9) and then confirm the class of the sample according to (10); go to Step 5; otherwise, go to Step 4.

Step 4. If the sample point is contained by no less than two hyperellipsoidals $E(a_m, R_m)$ ($m \in 1, 2, \dots, NN$), then compute the membership that the sample belongs to the m th class according to (11) and then confirm the class of the sample according to (10); go to Step 5.

Step 5. End.

4. Experiments

Experiments are made on Reuters 21578, in which five categories and 2302 texts are used. 1536 texts are used as training set, and the rest are used as testing set (see Table 1). Information gain is used to reduce feature dimension and the weight of every word is computed according to TF-IDF.

To verify the efficiency of the proposed method, the same task is realized by using HSSVMIL, HEIL, and MSVMIL. The computational experiments were done on a Pentium 1.6 G with 512 MB memory. Kernel function is radial basis function (RBF) $K(x, y) = e^{-\gamma \|x - y\|^2}$, where $\gamma = 0.01$. Penalty parameter of MSVMIL $C = 100$. System parameter of HSSVMIL $\nu = 0.1$.

The macroaverage precision (MAAP), macroaverage recall (MAAR), and macroaverage F_1 (MAAF) are used to evaluate the classification performance of the algorithms. Defined as follows:

$$\text{Macro Average Precision (MAAP)} = \frac{\left(\sum_{i=1}^N P^i\right)}{N},$$

$$\text{Macro Average Recall (MIAR)} = \frac{\left(\sum_{i=1}^N R^i\right)}{N}, \quad (12)$$

$$\text{Macro Average } F_1 \text{ (MAAF)} = \frac{\left(\sum_{i=1}^N F_1^i\right)}{N},$$

TABLE 2: MAAP, MAAR, and MAAF OF three algorithms.

Learning	Algorithm	MAAP (%)	MAAR (%)	MAAF (%)
Original	SSVMIL	79.41	80.23	79.54
	HEIL	81.58	83.44	82.79
	MSVMIL	84.73	85.62	85.57
1st increment	SSVMIL	82.86	77.77	80.32
	HEIL	84.73	82.86	83.41
	MSVMIL	86.82	85.78	85.61
2nd increment	SSVMIL	80.46	78.85	79.60
	HEIL	82.62	81.58	82.12
	MSVMIL	86.30	86.81	86.44
3rd increment	SSVMIL	81.28	79.83	80.79
	HEIL	84.53	82.64	83.71
	MSVMIL	86.29	85.75	85.85

TABLE 3: Training time and testing time of three algorithms.

Learning	Algorithm	Training time (ms)	Testing time (ms)
Original	SSVMIL	157	187
	HEIL	166	154
	MSVMIL	159	162
1st increment	SSVMIL	141	129
	HEIL	167	115
	MSVMIL	143	118
2nd increment	SSVMIL	140	125
	HEIL	158	102
	MSVMIL	142	104
3rd increment	SSVMIL	156	127
	HEIL	179	112
	MSVMIL	158	114

where N is the number of classes, P^i is the precision of the i th class sample, R^i is the recall of i th class sample, and F_1^i is the F_1 value of i th class sample.

In experiments, the original dataset includes two class samples (Acq and Eran), and three times incremental learning are done. The first incremental samples include three classes (Acq, Eran, and Grain). The second incremental samples include four class samples (Acq, Eran, Grain, and Crude). The third incremental samples include four class samples (Acq, Eran, Crude, and Trade). The Macroaverage precision, macroaverage recall, and macroaverage F_1 of three algorithms are given in Table 2. The training time and testing time of three algorithms are given in Table 3.

The experimental results show that the classification precision and recall of MSVMIL are higher than HEIL and MSVMIL. The main reasons are that MSVMIL reduced the volume that surrounds sample points by pushing the outlier samples away and considered the distribution of samples by using Mahalanobis distance. The classification speed of MSVMIL is faster than SSVMIL and is equal basically to HEIL. The training speed of MSVMIL is faster than HEIL and is equal basically to SSVMIL.

5. Conclusion

To solve SVM multiclass incremental learning problem, a novel algorithm based on Mahalanobis hyperellipsoidal SVM is proposed. In the process of incremental learning, only the new samples and the old class support vectors that its class exist in the new samples take part in the training. The history classifiers that have nothing to do with the new samples are reused. In the process of classification, the Mahalanobis distance is used to confirm the classified sample class. The experimental results show, compared with HEIL and SSVMIL, the proposed algorithm has a higher performance in classification precision and classification speed. In fact, data driven [22, 23] is another worth-thinking method, and this idea would be presented in the further work.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This study is partly supported by the National Natural Science Foundation of China (no. 61304149, no. 11171042), the Natural Science Foundation of Liaoning Province in China (no. 201202003), and the education committee project of Liaoning province in China (no. L2014444).

References

- [1] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 2nd edition, 2000.
- [2] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: an overview," *IEEE Transactions on Industrial Electronics*, no. 99, 2014.
- [3] S. Yin, G. Wang, and H. R. Karimi, "Data-driven design of robust fault detection system for wind turbines," *Mechatronics*, vol. 24, no. 4, pp. 298–306, 2014.
- [4] X. Zhao, L. Zhang, P. Shi, and H. R. Karimi, "Robust control of continuous-time systems with state-dependent uncertainties and its application to electronic circuits," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 8, pp. 4161–4170, 2014.
- [5] B. Long, S. Tian, Q. Miao, and M. Pecht, "Research on features for diagnostics of filtered analog circuits based on LS-SVM," in *Proceedings of the Systems Readiness Technology Conference (AUTOTESTCON '11)*, pp. 360–366, Baltimore, Md, USA, September 2011.
- [6] K. Bennett and O. Mangasarian, "Combining support vector and mathematical programming methods for induction," in *Advances in Kernel Methods—SV Learning*, pp. 307–326, 1999.
- [7] S. R. Gunn, "Support vector machines for classification and regression," ISIS Technical Report vol. 14, 1998.
- [8] J. C. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin DAGs for multiclass classification," in *Proceedings of the 13th Annual Neural Information Processing Systems Conference (NIPS '99)*, pp. 547–553, December 1999.
- [9] S. Cheong, S. H. Oh, and S.-Y. Lee, "Support vector machines with binary tree architecture for multi-class classification,"

- Neural Information Processing-Letters and Reviews*, vol. 2, pp. 47–51, 2004.
- [10] A. A. Kotomin, A. S. Kozlov, V. V. Gorovtsov et al., “Regulation of detonation ability of explosive materials used in spacecraft separation systems,” *Solar System Research*, vol. 46, no. 7, pp. 511–518, 2012.
- [11] G. Madzarov, D. Gjorgjevikj, and I. Chorbev, “A multi-class SVM classifier utilizing binary decision tree,” *Informatica*, vol. 33, no. 2, pp. 225–233, 2009.
- [12] S. Rüping, “Incremental learning with support vector machines,” in *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM '01)*, pp. 641–642, December 2001.
- [13] C. Domeniconi and D. Gunopulos, “Incremental support vector machine construction,” in *Proceedings of the IEEE International Conference on Data Mining (ICDM '01)*, pp. 589–592, December 2001.
- [14] G. Cauwenberghs and T. Poggio, “Incremental and decremental support vector machine learning,” in *Proceedings of the Advances in Neural Information Processing Systems (NIPS '00)*, pp. 409–415, December 2000.
- [15] J. Zhang, Z. Li, and J. Yang, “A divisional incremental training algorithm of support vector machine,” in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '05)*, pp. 853–856, August 2005.
- [16] R. Kong and B. Zhang, “Fast incremental learning algorithm for support vector machine,” *Control and Decision*, vol. 20, no. 10, pp. 1129–1136, 2005.
- [17] B.-F. Zhang, J.-S. Su, and X. Xu, “A class-incremental learning method for multi-class support vector machines in text classification,” in *Proceeding of the International Conference on Machine Learning and Cybernetics*, pp. 2581–2585, Dalian, China, August 2006.
- [18] Y. Qin, Q. Leng, X. Meng, and Q. Luo, “A new incremental learning algorithm based on hyper-sphere SVM,” in *Proceedings of the 7th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '10)*, pp. 2340–2343, August 2010.
- [19] X.-K. Wei, G.-B. Huang, and Y.-H. Li, “Mahalanobis ellipsoidal learning machine for one class classification,” in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 3528–3533, 2007.
- [20] Y.-X. Li and Z.-X. Xue, “Maximal margin ellipsoid-shaped multi-class classification algorithm,” *Computer Engineering*, vol. 36, pp. 185–189, 2010.
- [21] X.-K. Wei, G.-B. Huang, and Y.-H. Li, “Mahalanobis ellipsoidal learning machine for one class classification,” in *Proceeding of the 6th International Conference on Machine Learning and Cybernetics (ICMLC '07)*, pp. 3528–3533, Hong Kong, China, August 2007.
- [22] S. Yin, X. Yang, and H. R. Karimi, “Data-driven adaptive observer for fault diagnosis,” *Mathematical Problems in Engineering*, vol. 2012, Article ID 832836, 21 pages, 2012.
- [23] S. Yin, H. Luo, and S. X. Ding, “Real-time implementation of fault-tolerant control systems with performance optimization,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 5, pp. 2402–2411, 2014.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

