Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Science Letters:

Binary tree of posterior probability support vector machines*

Dong-li WANG[1,2], Jian-guo ZHENG[1], Yan ZHOU[†‡2]

([1]*Glorious Sun School of Business and Management, Donghua University, Shanghai 200051, China*)

([2]*College of Information Engineering, Xiangtan University, Xiangtan 411105, China*)

[†]E-mail: sgirld@163.com

Received Feb. 1, 2010; Revision accepted Sept. 1, 2010; Crosschecked Dec. 30, 2010

Abstract: Posterior probability support vector machines (PPSVMs) prove robust against noises and outliers and need fewer storage support vectors (SVs). Gonen *et al.* (2008) extended PPSVMs to a multiclass case by both single-machine and multimachine approaches. However, these extensions suffer from low classification efficiency, high computational burden, and more importantly, unclassifiable regions. To achieve higher classification efficiency and accuracy with fewer SVs, a binary tree of PPSVMs for the multiclass classification problem is proposed in this letter. Moreover, a Fisher ratio separability measure is adopted to determine the tree structure. Several experiments on handwritten recognition datasets are included to illustrate the proposed approach. Specifically, the Fisher ratio separability accelerated binary tree of PPSVMs obtains overall test accuracy, if not higher than, at least comparable to those of other multiclass algorithms, while using significantly fewer SVs and much less test time.

Key words: Binary tree, Support vector machine, Handwritten recognition, Classification
doi:10.1631/jzus.C1000022 **Document code:** A **CLC number:** TP391

1 Introduction

Among many classification methods the support vector machine (SVM) (Cortes and Vapnik, 1995; Dietterich and Bakiri, 1995; Vapnik, 1995; Muller *et al.*, 2001) has demonstrated a superior performance (Hu *et al.*, 2005; Leng *et al.*, 2007; Huang and Zhu, 2010). However, the SVM was originally developed for binary decision problems. To apply SVMs to multiclass problems, there are two main types of approach, i.e., single-machine approach (Vapnik, 1998) and multimachine approach (e.g., Hsu and Lin, 2002, and the references therein). The former is not always practical since it generates a large optimization problem, which leads to time-consuming training. Among the multimachine approaches, one-against-all (OAA) (Vapnik, 1995) and one-against-one (OAO) (KreBel, 1999) are the two most common methods, in which n and $n(n-1)/2$ (n denotes the number of classes) binary SVMs are needed for one classification, respectively. Both OAA and OAO methods are special cases of the error correcting output codes (ECOC) (Dietterich and Bakiri, 1995), in which the main issue is to construct a good ECOC matrix.

To reduce the test computational complexity and avoid the unclassifiable region, a directed acyclic graph SVM (DAGSVM) (Platt *et al.*, 2000) and a binary tree of SVMs (c-BTS) (Fei and Liu, 2006) are proposed. DAGSVM needs only $n-1$ binary SVMs of OAO, while c-BTS needs $\log_{4/3}[(n+3)/4]$ binary SVMs of OAO on average for one classification. Several earlier approaches to multiclass SVM classification also used a decision tree structure. Take a few examples. Takahashi and Abe (2002) proposed four types of decision tree according to the number of classes separated from the remaining classes at each node and separability measure such as Euclidean and Mahalanobis distance. To solve the face recognition problem, Guo *et al.* (2001) constructed a bottom-up binary tree for classification. The new recognition strategy extends the capability of a traditional bipartite framework for solving multiclass problems.

‡ Corresponding author

* Project (Nos. 60874104 and 70971020) supported by the National Natural Science Foundation of China

Recently, in terms of a decision tree of SVM (DTSVM), a problem of multiclass classification was divided into a series of binary classifications using the kernel clustering algorithm (Zhang *et al*., 2007).

2 Posterior probability support vector machines and the extension

For a multiclass problem, we have total l training samples belong to n classes: $(\boldsymbol{x}_1, z_1)$, $(\boldsymbol{x}_2, z_2)$, …, $(\boldsymbol{x}_l, z_l)$, where $\boldsymbol{x}_i \in \mathbb{R}^d$, $z_i \in \{1, 2, \ldots, n\}$. In the case of n=2, Tao *et al*. (2005) modified the canonical SVM to use class posterior probabilities (Xin *et al*., 2002) instead of using hard $\{z_1=1, z_2=-1\}$ labels. Then the PPSVM is aimed to solve the following optimization problem for soft margins:

$$\begin{aligned} &\min\left(\frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+C\sum_{j=1}^{l}\xi_j\right)\\ &\text{s.t.}\quad y_j(\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x}_j+b)\geq y_j^2-y_j^2\xi_j,\\ &\qquad \xi_j\geq 0,\quad j=1,2,...,l, \end{aligned} \tag{1}$$

where C is a predefined positive real number and ξ_j are slack variables. These 'soft labels' are calculated from estimated posterior probabilities as

$$y_j = 2\hat{P}(+\,|\,x_j)-1. \tag{2}$$

The posterior probability can be estimated by any density estimator, such as the windows method and the k-nearest neighbors (k-NN) method (Tao *et al*., 2005; Gonen *et al*., 2008).

The main advantage of using 'soft labels' instead of hard class labels is that the PPSVM performs closer to the Bayesian optimal classifier without knowing the distribution (Tao *et al*., 2005). The posterior probability at a point is the combined effect of a number of neighboring samples for the PPSVMs. This gives a chance to correct the error introduced by both wrongly labeled and noisy/outlier points (Gonen *et al*., 2008). Including 'soft labels' into the classifier also makes a sample surrounded by a number of samples of the same class become redundant, and hence decreases the number of support vectors (SVs). Thus, compared with canonical SVMs, the PPSVM has the superiorities such as higher accuracy, higher efficiency, and is more robust against noises and outliers.

However, the PPSVMs in Tao *et al*. (2005) were designed for two categories of classification. In Gonen *et al*. (2008), the PPSVM was extended to a multiclass case by both single-machine and multi-machine approaches including the OAA and OAO methods. Experiments on 20 datasets showed that PPSVM achieves similar accuracy while storing fewer SVs.

Unfortunately, these extensions suffer from a low classification efficiency and a high computational burden, which will be shown in Section 4. More importantly, the extensions in Gonen *et al*. (2008) intuitively suffer from an unclassifiable region, similar to both the canonical OAO and OAA approaches (Takahashi and Abe, 2002). In this letter, the PPSVM will be extended to a multiclass classification problem using a binary tree structure and boosting with the Fisher ratio separability measure. The theoretical analysis shows that the proposed binary tree of PPSVMs (BTPPSVMs) needs only to train $n-1$ binary PPSVMs in the best situation and needs a decision complexity of $O(\log_2 n)$ binary tests for one classification. Experiments on several benchmark datasets show that the proposed algorithm can obtain classifying accuracy higher than or comparable to those of other multiclass algorithms with fewer support vectors stored.

3 The proposed algorithm

Tree-based classifiers have presented a fast and effective way to structure and solve multiclass classification problems. As discussed above, several earlier approaches have adopted the decision tree structure to solve the multiclass SVM classification. However, there is no published work indicating how to extend the PPSVM to a multiclass case from a binary tree case, and what benefits will be obtained compared to the multiclass canonical SVM and PPSVM.

The efficiency of multiclass SVM methods can be verified in terms of computational complexity and generalization capability. As illustrated earlier, the PPSVMs are robust against noises and outliers with a smaller number of SVs. Hence, it is hoped to reach reduced computational complexity when extended to BTPPSVMs. Note that the total generalization of the binary tree of PPSVMs can be affected by the gener-

alization of each binary PPSVM and the tree structure. Assuming that the decisions of different layers are independent, the final estimate of the generalization error e can be formulated as

$$e = 1 - \prod_{k=1}^{K} (1 - e_k), \tag{3}$$

where e_k is the error of the decision of the kth layer and K is the number of layers. It is easy to conclude that the closer the occurrence of the classification error to the root node, the higher the overall classification accuracy. To maintain high generalization ability, the class pair with the largest separability measure should be separated at the upper nodes of the binary tree. Thus, demand on both computational complexity and generalization capability motivates the research of this work, where PPSVMs are extended to the multiclass case using a binary tree structure and exploiting the Fisher ratio as the class separability measure.

Denote the mean and variance of samples in classes i and j by μ_i, μ_j and σ_i^2, σ_j^2, respectively. The Fisher ratio is given by (Duda and Har, 1973)

$$r_{ij} = \frac{(\mu_i - \mu_j)^2}{\sigma_i^2 + \sigma_j^2}. \tag{4}$$

The Fisher ratio provides a good class separability measure because it is defined as the ratio of the interclass difference to the intraclass spread.

3.1 Procedure

In BTPPSVM, the tree is defined in such a way that each internal node (a PPSVM classifier) determines a hyperplane between an accepted pair belonging to two classes. Here we use the Fisher ratio as the class separability measure to determine the accepted pair. The algorithm is drawn up as follows.

Step 1: Root node

(i) Set layer index k=1, and add all information classes into the root node.

(ii) Calculate the Fisher ratio between class i and class j, r_{ij} (i, j=1, 2, …, n).

(iii) Select the two classes with the largest Fisher ratio separability measure as the accepted pair and separate all other information classes by the hyperplane determined by the currently accepted pair and the PPSVM strategy. Suppose all information classes are divided into two groups Ω_{L}^k and Ω_{R}^k.

Step 2: k-layer branching (k≥2)

(i) If Card$\{\Omega_D^k\} \geq 2$, $D = \{\mathrm{R}, \mathrm{L}\}$, divide Ω_D^k into two subnodes and determine the PPSVM classifier according to (ii)–(iii) in Step 1; here Card{} stands for the cardinality of a set.

(ii) Set k=k+1.

Step 3: If only one class belongs to nodes in the kth layer, stop. Otherwise, go to Step 2.

3.2 Performance analysis

Return to the construction of the BTPPSVMs. Suppose the numbers of the classes in one node and its child nodes are n_0, n_1, and n_2, respectively. In the best situation, we have n_0=n_1+n_2 because the hyperplane is determined by the two clusters, and the two child nodes have no common class. Finally, the binary trees must have n leaf nodes and a total of 2n−1 nodes in the tree. Therefore, there are (2n−1)−n=n−1 internal nodes, each representing a PPSVM classifier.

Now we are in the position to analyze the convergence performance of the proposed algorithms. The root node contains n classes while at the 1st layer, one node on average has n_1=0.5n classes. Furthermore, it is easy to prove that at the kth layer, one node on average has n_k=$0.5^k n$ classes. When n_k=1, the BTPPSVM will converge, in which case we have n_k=$0.5^k n$=1. Therefore, the average convergence performance is $O(\log_2 n)$.

Other strategies such as reassignment according to the estimated probability can also be applied to improve the accuracy. However, as pointed out in Fei and Liu (2006), a larger reassignment threshold leads to an increased number of classifiers and training time without any significant improvement in accuracy. Note that BTPPSVMs are expected to need fewer SVs stored and are robust against outliers due to the PPSVM classifier adopting ‘soft labels’. This will certainly increase the speed of the decision convergence rate and improve the generalization accuracy.

4 Results

In this section, several experiments were performed on multiclass handwritten recognition datasets, i.e., Letter (ftp://ftp.ncc.up.pt/pub/statlog/),

Optdigit (http://www.ics.uci.edu/mlearn/MLRepository.htm), Pendigit (http://www.ics.uci.edu/mlearn/MLRepository.htm), and MNIST (http://yann.lecun.com/exdb/MNIST/), as summarized in Table 1.

Table 1 Summary of multiclass handwritten recognition datasets

Dataset	Number of classes	Number of features	Number of training data	Number of test data
Letter	26	16	15000	5000
Optdigit	10	64	3823	1797
Pendigit	10	16	7494	3498
MNIST	10	49	60000	10000

For comparison, OAO, OAA, and the recently proposed c-BTS using the canonical SVMs were also used. For all experiments, the radial basis function (RBF) kernel was used with parameters C and γ selected according to five-fold cross-validation. The k-NN method was used to estimate the posterior probability with k=11. Tables 2–5 show the results of overall classification accuracy, the number of SVs, the number of binary classifiers used for one classification, and the training and test time for the different approaches, respectively. The experiments were performed on a PC with 2.3 GHz CPU, 2 GB DDR3 memory.

BTPPSVM achieves an overall test accuracy, if not higher than, at least comparable to those of OAO and OAA approaches using either canonical SVMs or PPSVMs (Table 2). In general, OAO obtains the best accuracy, and the test accuracy of BTPPSVM is about 0.1% higher than that of c-BTS. Specifically, BTPPSVM ranks second among the six approaches, and obtains an accuracy equal to that of the canonical OAO approach on average. The three algorithms using PPSVMs need to store significantly fewer SVs (Table 3). In general, compared to the canonical OAO algorithm and posterior probability OAO, BTPPSVM needs only to store 60% and 78% of SVs, respectively. It is well known that the computational complexity of a binary tree is $O(n_{SV})$, where n_{SV} denotes the number of support vectors for one classification. This reduces the test time and computational complexity of BTPPSVM. Moreover, in most cases BTPPSVM reduces the number of binary classifiers compared with both OAO and c-BTS (Table 4). This is because the Fisher ratio is used as the class separability measure. Both c-BTS and BTPPSVM use significantly fewer classifiers than the OAO approach for one classification. Combined with the reduced number of SVs, this enables BTPPSVM to achieve a much higher classification efficiency and less computational complexity, and a faster test convergence rate (Table 5).

Table 2 Overall accuracy of different approaches

Dataset	Overall accuracy (%)					
	Canonical SVM			PPSVM		
	OAO	OAA	c-BTS	OAO	OAA	BTPPSVM
Letter	97.96	97.88	97.92	98.11	97.73	97.92
Optdigit	98.46	98.55	98.46	98.62	98.46	98.62
Pendigit	98.26	97.40	98.11	98.42	98.38	98.19
MNIST	97.49	97.12	97.24	97.04	97.16	97.43
Average	98.04	97.74	97.93	98.05	97.93	98.04

Table 3 Percentage of support vectors of different approaches

Dataset	Percentage of support vectors (%)					
	Canonical SVM			PPSVM		
	OAO	OAA	c-BTS	OAO	OAA	BTPPSVM
Letter	59.5	67.5	36.9	42.1	45.5	27.4
Optdigit	39.9	32.9	28.1	32.5	31.0	26.8
Pendigit	44.3	53.7	40.8	37.8	42.2	31.6
MNIST	22.9	28.6	14.5	15.2	22.4	14.1
Average	41.7	45.7	30.1	31.9	35.3	25.0

Table 4 Number of binary classifiers for one classification

Dataset	Number of binary classifiers					
	Canonical SVM			PPSVM		
	OAO	OAA	c-BTS	OAO	OAA	BTPPSVM
Letter	325	26	268	325	26	172
Optdigit	45	10	36	45	10	32
Pendigit	45	10	30	45	10	24
MNIST	45	10	36	45	10	32

Table 5 Average training and test time for the dataset Letter

Algorithm	Training time (s)	Test time (s)
Canonical SVM		
OAO	112.59	4.07
OAA	86.31	3.86
c-BTS	101.66	4.71
PPSVM		
OAO	97.36	3.92
OAA	85.74	2.49
BTPPSVM	80.38	2.02

The average training and test time for the six different algorithms on the Letter dataset are compared (Table 5), showing that the proposed BTPPSVM

needs the shortest test time. Specifically, compared with the OAO and OAA of PPSVMs, BTPPSVM saves about 48.5% and 18.9% test time, respectively. This illustrates the higher classification efficiency of BTPPSVM from one perspective. On the other hand, compared with the canonical multiclass SVMs, the test time of BTPPSVM is reduced by 50.4%, 47.7%, and 57.1%, respectively. This is mainly because BTPPSVM uses significantly fewer classifiers than both the canonical and posterior probability OAO approaches while storing an obviously reduced number of SVs. For brevity, we omit the results for the other three datasets, which are similar.

To assess the effect of the *k*-NN method over the test error rate and the number of SVs for BTPPSVM, we varied the neighborhood from 1 (i.e., the canonical case) to 11 on the Letter dataset (Fig. 1). Both the test error rate and the number of SVs decrease as the neighborhood increases. The results of the other three datasets have similar patterns, and thus are omitted.

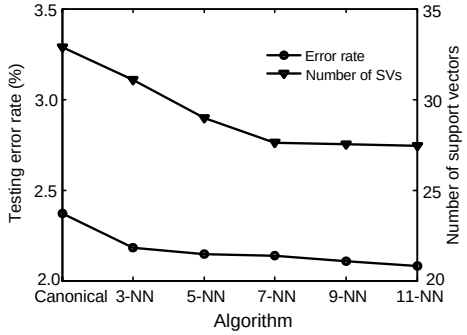

Fig. 1 Effect of the *k*-NN method over the test error rate and the number of support vectors on the Letter dataset

5 Conclusions

A binary tree of posterior probability support vector machines has been proposed for the multiclass classification problem. Fisher ratio separability measure has been adopted to determine the tree structure and accelerate the classifying speed. The proposed algorithm needs only to train $n-1$ binary PPSVMs in the best situation and needs decision complexity of $O(\log_2 n)$ binary tests for one classification. Experiments on several benchmark datasets show that the proposed BTPPSVMs can obtain classifying accuracy higher than or comparable to other multiclass algorithms, while using fewer SVs and reduced test time.

References

Cortes, C., Vapnik, V., 1995. Support-vector networks. *Mach. Learn.*, **20**(3):273-297. [doi:10.1007/BF00994018]

Dietterich, T.G., Bakiri, G., 1995. Solving multiclass learning problems via error-correcting output codes. *J. Artif. Intell. Res.*, **2**(1):263-286.

Duda, R.O., Har, P.E., 1973. Pattern Classification and Scene Analysis. Wiley, New York.

Fei, B., Liu, J., 2006. Binary tree of SVM: a new fast multiclass training and classification algorithm. *IEEE Trans. Neur. Netw.*, **17**(3):696-704. [doi:10.1109/TNN.2006.872343]

Gonen, M., Tanuğur, A.G., Alpaydin, E., 2008. Multiclass posterior probability support vector machines. *IEEE Trans. Neur. Netw.*, **19**(1):130-139. [doi:10.1109/TNN.2007.903157]

Guo, G., Li, S.Z., Chan, K.L., 2001. Support vector machines for face recognition. *Image Vis. Comput.*, **19**(9-10):631-638. [doi:10.1016/S0262-8856(01)00046-4]

Hsu, C.W., Lin, C.J., 2002. A comparison of methods for multi-class support vector machines. *IEEE Trans. Neur. Netw.*, **13**(2):415-425. [doi:10.1109/72.991427]

Hu, Z.H., Cai, Y.Z., Li, Y.G., Xu, X.M., 2005. Data fusion for fault diagnosis using multi-class support vector machines. *J. Zhejiang Univ.-Sci.*, **6A**(10):1030-1039. [doi:10.1631/jzus.2005.A1030]

Huang, P., Zhu, J., 2010. Multi-instance learning for software quality estimation in object-oriented systems: a case study. *J. Zhejiang Univ.-Sci. C (Comput & Electron.)*, **11**(2):130-138. [doi:10.1631/jzus.C0910084]

KreBel, U.H.G., 1999. Pairwise classification and support vector machine. *In*: Schölkopf, B., Burges, C.J., Smola, A.J. (Eds.), Advances in Kernel Methods: Support Vector Learning. MIT Press, Cambridge, MA.

Leng, B., Qin, Z., Li, L.Q., 2007. Support vector machines active learning for 3D model retrieval. *J. Zhejiang Univ.-Sci. A*, **8**(12):1953-1961. [doi:10.1631/jzus.2007.A1953]

Muller, K.R., Mika, S., Ratisch, G., Tsuda, K., Scholkopf, B., 2001. An introduction to kernel-based learning algorithms. *IEEE Trans. Neur. Netw.*, **12**(2):181-201. [doi:10.1109/72.914517]

Platt, J., Cristianini, N., Shawe-Taylor, J., 2000. Large margin DAGSVM's for multiclass classification. *Adv. Neur. Inform. Process. Syst.*, **12**:547-553.

Takahashi, F., Abe, S., 2002. Decision-Tree-Based Multiclass Support Vector Machine. Proc. 9th Int. Conf. on Neural Information, p.1418-1422.

Tao, Q., Wu, G.W., Wang, F.Y., Wang, J., 2005. Posterior probability support vector machines for unbalanced data. *IEEE Trans. Neur. Netw.*, **16**(6):1561-1573. [doi:10.1109/TNN.2005.857955]

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer Verlag, New York.

Vapnik, V., 1998. Statistical Learning Theory. John Wiley & Sons, New York.

Xin, D., Wu, Z.H., Pan, Y.H., 2002. Probability output of multi-class support vector machines. *J. Zhejiang Univ.-Sci.*, **3**(2):131-134. [doi:10.1631/jzus.2002.0131]

Zhang, L., Zhou, W.D., Su, T.T., Jiao, L.C., 2007. Decision tree support vector machine. *Int. J. Artif. Intell. Tools*, **16**(1):1-15. [doi:10.1142/S0218213007003163]