Selecting Proper Multi-class SVM Training Methods

Yawen Chen[†], Zeyi Wen[‡], Jian Chen[†*], Jin Huang[§]
[†]ywchenscut@gmail.com, ellachen@scut.edu.cn
South China University of Technology, China
[‡]wenzy@comp.nus.edu.sg
National University of Singapore, Singapore
[§]jinhuang@scnu.edu.cn
South China Normal University, China

Abstract

Support Vector Machines (SVMs) are excellent candidate solutions to solving multi-class problems, and multi-class SVMs can be trained by several different methods. Different training methods commonly produce SVMs with different effectiveness, and no multi-class SVM training method always outperforms other multi-class SVM training methods on all problems. This raises difficulty for practitioners to choose the best training method for a given problem. In this work, we propose a Multi-class Method Selection (MMS) approach to help users select the most appropriate method among one-versus-one (OVO), one-versus-all (OVA) and structural SVMs (SSVMs) for a given problem. Our key idea is to select the training method based on the distribution of training data and the similarity between different classes. Using the distribution and class similarity, we estimate the unclassifiable rate of each multi-class SVM training method, and select the training method with the minimum unclassifiable rate. Our initial findings show: (i) SSVMs with linear kernel perform worse than OVO and OVA; (ii) MMS often produces SVM classifiers that can confidently classify unseen instances.

Introduction

Existing studies have shown that Support Vector Machines (SVMs) (Vapnik 1998) perform well in solving multi-class classification problems (Hsu and Lin 2002). There are two types of approaches to solving multi-class problems using SVMs. One is solving the multi-class problem using one optimization problem, and the other is by decomposing the multi-class problem into several binary SVM training problems. Structural SVMs (SSVMs) solve the multi-class problem as one optimization problem; one-versus-one (OVO) and one-versus-all (OVA) decomposition solve the multi-class problem using several binary SVM training problems.

With several approaches to multi-class SVM training and classification, choosing a proper approach to solve a particular problem is difficult, since different training methods commonly produce SVMs with significantly different effectiveness. Existing studies (Crammer and Singer 2002; Hsu and Lin 2002) have shown that no multi-class SVM training method outperforms other methods in all problems. In this work, we do not concentrate on which method is better in general, but aim to find an approach to help users, especially those who are from other fields, to automatically choose the most appropriate method for a given problem. Our key idea is to select the training method based on the distribution of training data and the similarity between different classes. According to the balance and similarity of the data, we estimate the unclassifiable rate of each multi-class SVM training method, and select the training method with the minimum unclassifiable rate. Our initial findings show: (i) SSVMs with linear kernel perform worse than OVO and OVA; (ii) MMS often produces SVM classifiers that can confidently classify unseen instances.

Our MMS approach

Considering the constraints of each method, we propose a *Multi-class Method Selection (MMS)* approach to help users automatically choose the appropriate multi-class SVM training method for a given problem. Next, we present two measurements for training data, and describe our MMS method.

Measurement of balance

We use the variance to measure the balance of data distribution. We first normalize the number of instances in each class, and then compute the variance V using Equation (1).

$$\begin{aligned} n'_l &= \frac{n_l}{n_1 + n_2 + \cdots + n_K}, l = 1, 2, \cdots, K \\ V &= \frac{\Sigma_{l=1}^{K}(n'_l - \frac{1}{K})^2}{K-1} \end{aligned} \tag{1}$$

where K is the number of classes, n_l is the number of instances in class l, and n'_l is the normalized n_l.

Measurement of similarity

Although other similarity functions will be explored in this work, here we use a radials basis function (RBF) kernel to measure the similarity of each pair of classes in a dataset. We first randomly choose some instances in each class. Then we follow Equation (2) to obtain the similarity $\bar{S}$ of the dataset.

$$\begin{aligned} S(I, J) &= \min\{\exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})\}, i \in I, j \in J \\ \bar{S} &= \frac{\sum_{I \in K, J \in K, I \neq J} S(I, J)}{K} \end{aligned} \tag{2}$$

*Jian Chen is the corresponding author.

where $S(I, J)$ denotes the similarity between class I and class J. $\bar{S}$ is the average similarity of all pair similarities (the number of pairs equals $\frac{K(K-1)}{2}$).

Our MMS approach

Our MMS approach consists of two steps. First, we calculate the balance of the dataset according to Equation (1). If V is large, then the data distribution is very unbalanced. Since OVA suffers from the unbalanced data issue, we choose OVO to solve the multi-class classification. Otherwise, we evaluate the second step.

Second, we calculate the dataset similarity using Equation (2). If $\bar{S}$ is large, the instances in each class are closed which implies that the unclassifiable rate is large. It is known that the unclassifiable rate of OVA is larger than OVO's in general. In OVO, the votes of a new instance for different classes are probably the same which makes this instance assigned to more than one classes. Besides that, OVA may classify the instances to none of the classes. Therefore if $\bar{S}$ is large, we should use OVO. If V and $\bar{S}$ are both small, we choose either OVO or OVA. As for SSVMs with linear kernel, we found in experiments that they always perform the worst, so we will test SSVMs with other kernels.

Experimental results

In this section, we provide our experimental results. We conducted our experiments on several datasets from LIBSVM website (Chang and Lin 2011). Our experiments are based on the SVM library "Mascot" (Wen et al. 2014) which produce the same results as LIBSVM does. For OVO and OVA, we choose the RBF kernel to train the SVMs. We choose linear kernel to train SSVMs which we will improve later.

Test accuracy comparison

In Table 1, the test accuracy of OVO and OVA is overall higher than SSVMs. If we predict labels for the unclassifiable instances ("UINS" in Table 1) by the combination strategy, the performance of OVA and OVO is similar. However, if we do not predict the labels of unclassifiable instances ("NUINS" in Table 1), the test accuracy of OVA decreases more than OVO as shown below the "NUINS". In Table 2, we can see the unclassifiable rate (i.e., $\frac{\text{\# of unclassifiable instances}}{\text{\# of testing instances}}$) using OVO and OVA. While using OVO, instances may belong to more than one class ("MTO" in Table 2). While using OVA, not only instances may belong to more than one class, but also they may belong to none of all the classes ("NC" in Table 2). Table 2 shows that the unclassifiable rate of OVA is higher than that of OVO. When V and $\bar{S}$ increase, the unclassifiable rates of OVO and OVA both increase.

It is because when the V is large, the data distribution is unbalanced. OVA only assigns the instances from one class to be positive and all the remaining instances to be negative, which makes the unbalance even worse. When $\bar{S}$ is large, the classes in the dataset are close to each other, and it is hard to classify the instances which are closed. These lead to increases in unclassifiable rate and decreases in the accuracy of OVA. The above results indicate that although the performance of OVA and OVO is similar, when the V or $\bar{S}$ is large, the unclassifiable rate of OVA will be high, reducing confidence in the predictions made by OVA. Therefore, in these situations, we recommend users to choose OVO method.

Table 1: Comparison of test accuracy (%)

dataset	NUINS		UINS		SSVMs
	OVO	OVA	OVO	OVA	
satimage	91.9	89.45	92	91.9	81.9
pendigits	98.46	97.48	98.48	98.8	90.91
usps	95.57	92.97	95.71	95.86	91.93
letter	97.98	95.44	98.02	97.88	76.96

Table 2: Unclassifiable rate of OVO and OVA

dataset	OVO(%)	OVA(%)		V	$\bar{S}$
	MTO	MTO	NC		
satimage	0.0025	0.25	3.90	0.0050	0.98
pendigits	0.0023	0.29	1.37	1.78e-5	0.00
usps	0.0045	1.10	3.84	0.0008	0.78
letter	0.0008	0.92	2.52	1.83e-6	0.99

Conclusion and future work

In this work, we propose the MMS approach that can help users select the most appropriate method for a particular multi-class problem without trying all the methods. From the experiments, we found SSVMs with linear kernel perform worse than OVO and OVA. Our MMS approach often produces SVM classifiers that can confidently classify unseen instances by taking the unclassifiable rate into account.

In the future, we will search for a faster way to train SSVMs, such that we can use non-linear kernels. Moreover, we plan to explore other ways of measuring balance and similarity, and provide theoretical analysis to our approach.

Acknowledgement

This work is supported by the Special Planning Project of Guangdong Province, China (Grant No. 609055894069).

References

Chang, C.-C., and Lin, C.-J. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2:27:1–27:27. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

Crammer, K., and Singer, Y. 2002. On the learnability and design of output codes for multiclass problems. *Machine learning* 47(2):201–233.

Hsu, C.-W., and Lin, C.-J. 2002. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks* 13(2):415–425.

Vapnik, V. 1998. *Statistical learning theory. 1998*. Wiley, New York.

Wen, Z.; Zhang, R.; Ramamohanarao, K.; Qi, J.; and Taylor, K. 2014. Mascot: fast and highly scalable svm cross-validation using gpus and ssds. In *Data Mining (ICDM), 2014 IEEE International Conference on*, 580–589. IEEE.