Information Sciences 382–383 (2017) 292–307

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Finding the samples near the decision plane for support vector learning

Fa Zhu[a,b,*], Jian Yang[a], Junbin Gao[c], Chunyan Xu[a], Sheng Xu[d], Cong Gao[e]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, PR China
[b] Centre for Quantum Computation & Intelligent Systems and the Faculty of Engineering and Information Technology, University of Technology, Sydney NSW 2007, Australia
[c] Discipline of Business Analytics, University of Sydney Business School, University of Sydney, NSW 2006, Australia
[d] Department of Geomatics Engineering, University of Calgary, Canada
[e] Department of Computer Science, University of Regina, Canada

ARTICLE INFO

Article history:
Received 14 March 2016
Revised 20 October 2016
Accepted 13 December 2016
Available online 14 December 2016

Keywords:
Data pre-processing
Extended nearest neighbor chain
SVM
Subset selection

ABSTRACT

The decision plane of support vector machine (SVM) is decided by a few support vectors (SVs) in the training set. If there exist overlapping regions among different classes, SVs mainly locate in the overlap regions. A number of approaches have been proposed to find the samples in overlapping regions to condense the training set. However, the performance of these approaches would degrade if there is no overlapping region in the training set. In this paper, the extended nearest neighbor chain is proposed to find samples near the decision plane to avoid degrading performance in the cases of no overlapping region between different classes. Experimental results demonstrate that the proposed method performs better than the previous ones on artificial synthetic datasets as well as benchmark datasets. Additionally, the proposed method can obtain a higher compression ratio than previous ones.

1. Introduction

SVM has very good generalization ability [4,28]. It has been used in a wide range of applications, such as object detection, image denoising, and object tracking etc. [21,31,32]. In SVMs, we need to solve quadratic programming (QP) whose time complexity and space complexity are both too high for a large scale dataset. It is imperative to explore strategies to speed up SVMs for practical applications. Prior work in speeding up SVMs can be categorized into: algorithmic improvements [8,11,23,26], new model establishment [9,10,17,18,27], and data pre-processing [5,12,13,20,24,25,29,33,34]. The aim of algorithmic approach is to make QP solver become faster. The decomposition algorithm [19] and Sequential Minimal Optimization (SMO) algorithm [22] are both classical algorithmic methods. The aim of new model establishment is to avoid solving QP, such as core vector machines [9,27], proximal support vector machine [17,18], and twin support vector machines [10]. The aim of data pre-processing is to identify those samples, which would become SVs, before SVM training. If all non-support vectors are removed, the performance would not degrade. After data pre-processing, we only need to train SVM on a smaller subset. Thus, the model would become simpler. Additionally, a smaller dataset is also beneficial for tuning the parameters, which plays an important role in SVMs. Furthermore, data pre-processing does not conflict with other two speedup strategies and can be combined with the two former strategies.

* Corresponding author.
E-mail address: zhufag@gmail.com (F. Zhu).

http://dx.doi.org/10.1016/j.ins.2016.12.019
0020-0255/

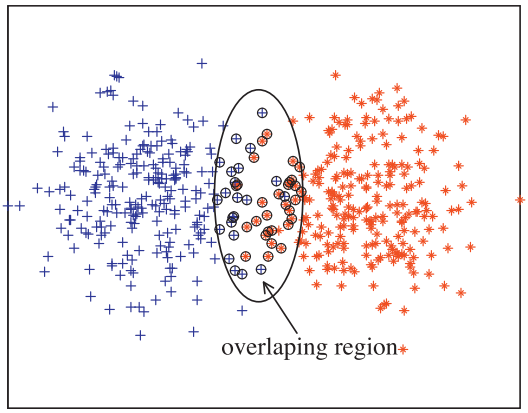

Fig. 1. The overlapping region between different classes.

Generally, SVs locate in the overlapping regions which are between different classes. Shin and Cho [24] proved that merely preserving the samples in the overlap regions, the performance of SVM does not degrade. Unfortunately, it is invalid for training sets without overlapping regions. In this paper, we find the samples near the decision plane instead of that in overlapping regions. The proposed approach can also work well for the training sets without overlapping regions.

The paper is organized as follows. The related work on condensing training set for SVMs is reviewed in Section 2. A novel method is proposed to find the samples near decision plane in Section 3. Section 4 provides the validation of the proposed method on artificial synthetic datasets as well as benchmark datasets. Discussion and conclusive remarks are provided in the last section.

2. Related work

Lee and Mangasarian [13] chose a subset by random selection to represent original training set and proposed Reduced Support Vector Machine (RSVM). Lin and Lin [16] proved that the performance of RSVM is unstable. De Almeida et al. [5] clustered the training set by K-means algorithm and each cluster was represented by its center. Koggalage and Halgamuge [12] also utilized cluster algorithm to condense training set, but they reserved boundary samples of each cluster instead. Zheng and Wang [25,29] utilized cluster algorithm to condense training set too. The performance of those clustering-based methods depends on the cluster algorithm. However, the performance of cluster algorithm is usually unstable and the number of clusters is difficult to choose.

For binary-class or multi-class SVMs, support vectors locate in the overlapping regions. In two-class problem, the overlapping region is a hypothetical region where positive samples and negative samples mix together. In Fig. 1, pluses and stars belong to different classes, while circles are the samples locating in the overlapping region.

According to neighborhood properties, Shin and Cho [24] defined two measures: Neighbors_Entropy and Neighbors_Match, as follows.

$$Neighbors\_Entropy(\boldsymbol{x}_i, k) = \sum_{j=1}^{J} P_j \cdot log_J\left(1/P_j\right) \tag{1}$$

$$Neighbors\_Match(\boldsymbol{x}_i, k) = \frac{\left|\left\{\boldsymbol{x}_j \middle| label\left(\boldsymbol{x}_i^j\right) = label(\boldsymbol{x}_i), \boldsymbol{x}_j \in kNN(\boldsymbol{x}_i)\right\}\right|}{k} \tag{2}$$

where J represents the number of the classes, P_j is defined as k_j/k (k is the number of nearest neighbors, and k_j is the number of the neighbors belonging to class j). $kNN(\boldsymbol{x}_i)$ is the k-nearest neighbors list of $\boldsymbol{x}_i$ and consists of $\boldsymbol{x}_i^j$, $j = 1, \ldots, k$. $label(\boldsymbol{x}_i^j) = label(\boldsymbol{x}_i)$ means that both $\boldsymbol{x}_i$ and $\boldsymbol{x}_i^j$ belong to the same class. He only reserved each sample which satisfies $Neighbors\_Entropy(\boldsymbol{x}_i, k) > 0$ and $Neighbors\_Match(\boldsymbol{x}_i, k) \geq 1/J$ simultaneously. This method is called neighborhood property–based pattern selection (NPPS) algorithm in the rest of the paper.

Panda et al. [20] proposed a scoring function and the corresponding normalized score as follows.

$$c\left(\boldsymbol{x}_i, \boldsymbol{x}_i^j\right) = \exp\left(-\left(\left\|\boldsymbol{x}_i - \boldsymbol{x}_i^j\right\|_2^2 - \tau_i\right)/\gamma\right) \tag{3}$$

$$S_{\boldsymbol{x}_i} = \frac{1}{\#\boldsymbol{x}_i} \sum_{\boldsymbol{x}_i^j \in kNN(\boldsymbol{x}_i)} c\left(\boldsymbol{x}_i, \boldsymbol{x}_i^j\right) \tag{4}$$