Generating the Reduced Set by Systematic Sampling

Chien-Chung Chang and Yuh-Jye Lee
Email: {D9115009, yuh-jye}@mail.ntust.edu.tw

Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Taipei, 106 Taiwan

Abstract. The computational difficulties occurred when we use a conventional support vector machine with nonlinear kernels to deal with massive datasets. The reduced support vector machine (RSVM) replaces the fully dense square kernel matrix with a small rectangular kernel matrix which is used in the nonlinear SVM formulation to avoid the computational difficulties. In this paper, we propose a new algorithm, Systematic Sampling RSVM (SSRSVM) that selects the informative data points to form the reduced set while the RSVM used random selection scheme. This algorithm is inspired by the key idea of SVM, the SVM classifier can be represented by support vectors and the misclassified points are a part of support vectors. SSRSVM starts with an extremely small initial reduced set and adds a portion of misclassified points into the reduced set iteratively based on the current classifier until the validation set correctness is large enough. In our experiments, we tested SSRSVM on six public available datasets. It turns out that SSRSVM might automatically generate a smaller size of reduced set than the one by random sampling. Moreover, SSRSVM is faster than RSVM and much faster than conventional SVM under the same level of the test set correctness.

Keywords: Support vector machine, reduced support vector machine, reduced set, kernel function, systematic sampling.

1 Introduction

For the binary classification problems, SVMs are able to construct nonlinear separating surface (if it is necessary) which is implicitly defined by a kernel function [1, 9]. Nevertheless, there are some major computational difficulties such as large memory usage and long CPU time in generating a nonlinear SVM classifier for a massive dataset. To overcome these difficulties, the reduced support vector machine (RSVM) [5] was proposed. The RSVM replaces the fully dense square kernel matrix with a small rectangular kernel matrix which is used in the nonlinear SVM formulation to avoid the computational difficulties. This reduced kernel technique has been successfully applied to other kernel-based learning algorithms [3, 4].

In this paper, we use a systematic sampling mechanism to select a reduced set which is the most important ingredient of RSVM and name it as Systematic Sampling RSVM (SSRSVM). This algorithm is inspired by the key idea of SVM that the SVM classifier can be represented by support vectors and the misclassified points are a part of support vectors. The SSRSVM randomly selects an extremely small subset as an initial reduced set. Then, a portion of misclassified points are added into the reduced set iteratively based on the current classifier until the validation set correctness is large enough. We tested SSRSVM on six public available datasets [2, 8]. The SSRSVM can generate a smaller reduced set than RSVM without scarifying the test set correctness.

A word about our notations is given below. All vectors will be column vectors unless otherwise specified or transposed to a row vector by a prime superscript $'$. For a vector $x \in R^n$, the plus function x_+ is defined as $(x)_+ = \max\{0, x\}$. The inner product of two vectors $x, z \in R^n$ will be denoted by $x'z$ and the p-norm of x will be denoted by $\|x\|_p$. For a matrix $A \in R^{m\times n}$, A_i is the ith row of A which is *a row vector* in R^n. A column vector of ones of arbitrary dimension will be denoted by e. For $A \in R^{m\times n}$ and $B \in R^{n\times l}$, the kernel $K(A, B)$ maps $R^{m\times n} \times R^{n\times l}$ into $R^{m\times l}$. In particular, $K(x', z)$ is a real number, $K(x', A')$ is a *row* vector in R^m, $K(A, x)$ is a *column* vector in R^m and $K(A, A')$ is an $m \times m$ matrix. The base of the natural logarithm will be denoted by ε.

This paper is organized as follows. Section 2 gives a brief overview of t he reduced support vector machines and discusses some related work. In section 3, we describe how to select the reduced set systematically from the entire dataset. The experimental results are given in section 4 to show the performance of our method. Section 5 concludes the paper.

2 An Overview of the Reduced Support Vector Machines

We now briefly describe the RSVM formulation, which is derived from the generalized support vector machine (GSVM) [7] and the smooth support vector machine (SSVM) [6]. We are given a training data set $\{(x^i, y_i)\}_{i=1}^m$, where $x^i \in R^n$ is an input data point and $y_i \in \{-1, 1\}$ is class label, indicating one of two classes, A_- and A_+, to which the input point belongs. We represent these data points by an $m \times n$ matrix A, where the ith row of the matrix A, A_i, corresponds to the ith data point. We denote alternately A_i (a row vector) and x^i (a column vector) for the same ith data point. We use an $m \times m$ diagonal matrix $D, D_{ii} = y_i$ to specify the membership of each input point. The main goal of the classification problem is to find a classifier that can predict the label of new unseen data points correctly. This can be achieved by constructing a linear or nonlinear separating surface, $f(x) = 0$, which is implicitly defined by a kernel function. We classify a test point x belong to A_+ if $f(x) \geq 0$, otherwise x belong to A_-. We will focus on the nonlinear case which is implicitly defined by a Gaussian kernel function. The RSVM solves the following unconstrained minimization problem

$$\min_{(\bar{v},\gamma)\in R^{\bar{m}+1}} \frac{\nu}{2}\|p(e - D(K(A,\bar{A}')\bar{v} - e\gamma),\alpha)\|_2^2 + \frac{1}{2}(\bar{v}'\bar{v} + \gamma^2), \tag{1}$$

where the function $p(x,\alpha)$ is a very accurate smooth approximation to $(x)_+$ [6], which is applied to each component of the vector $e - D(K(A,\bar{A}')\bar{v} - e\gamma)$ and is defined componentwise by

$$p(x,\alpha) = x + \frac{1}{\alpha}\log(1 + \varepsilon^{-\alpha x}), \alpha > 0. \tag{2}$$

The function $p(x,\alpha)$ converges to $(x)_+$ as α goes to infinity. The reduced kernel matrix $K(A,\bar{A}') \in R^{m\times\bar{m}}$ in (1) is defined by

$$K(A,\bar{A}')_{ij} = \varepsilon^{-\mu\|A_i - \bar{A}_j\|_2^2},\ 1 \leq i \leq m,\ 1 \leq j \leq \bar{m}, \tag{3}$$

where $\bar{A}$ is the reduced set that is randomly selected from A in RSVM [5]. The positive tuning parameter ν here controls the tradeoff between the classification error and the suppression of $(\bar{v},\gamma)$. Since RSVM has reduced the model complexity via using a much smaller rectangular kernel matrix we will suggest using a larger tuning parameter ν here. A solution of this minimization problem (1) for $\bar{v}$ and γ leads to the nonlinear separating surface

$$f(x) = \bar{v}'K(\bar{A},x) - \gamma = \sum_{i=1}^{\bar{m}} \bar{v}_i K(\bar{A}_i,x) - \gamma = 0. \tag{4}$$

The minimization problem (1) can be solved via the Newton-Armijo method [6] directly and the existence and uniqueness of the optimal solution of this problem are also guaranteed. We note that the computational complexity of solving problem (1) is depended on the size of the reduced set which is user prespecified in RSVM [5]. Moreover, the value of $K(A,\bar{A}')_{ij}$ in (3) can be interpreted as the *similarity* between examples A_i and $\bar{A}_j$. Hence the rectangular kernel matrix which is generated by a reduced set records the *similarity* between the entire training set and the reduced set. It seems indicate that if we had a more *representative* reduced set we should have a better classifier. In the next section, we describe how to generate a representative reduced set and apply it to RSVM.

3 Systematic Sampling for RSVM

We now propose a new algorithm to generate the reduced set which is consisting of the *informative* data points. This algorithm is inspired by the key idea of SVM, the SVM classifier can be represented by support vectors and the misclassified points are a part of support vectors. Instead of random sampling the reduced set in RSVM, we start with an extremely small initial reduced set and add a portion of misclassified points into the reduced set iteratively based on the current classifier. We note that there are two types of misclassified points and

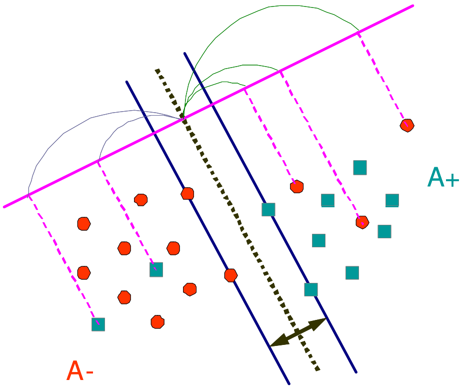

Fig. 1. The main idea of Systematic Sampling RSVM Algorithm

we select them respectively. We showed this idea in Fig. 1. The new reduced kernel matrix can be updated from the previous iteration. We only need to augment the columns which are generated by the new points in the reduced set. We stop this procedure until the validation set correctness is large enough.

Algorithm 3.1 Systematic Sampling RSVM Algorithm

(1) Randomly select an extremely small portion data points, such as $\bar{m} = 5$, from the entire training data matrix $A \in R^{m \times n}$ as an initial reduced set which is represented by $\bar{A}_0 \in R^{\bar{m} \times n}$.
(2) Generate the reduced kernel matrix $K(A, \bar{A}_0')$ and perform RSVM algorithm [5] to generate a tentative separating surface represented by $f(x) = 0$.
(3) Use the separating surface to classify the point which is in the training set but not in the current reduced set. Let $\bar{I}_+$ be the index set of misclassified points of positive example. That is, $\bar{I}_+ = \{i | f(A_i) \leq 0, A_i \in A_+\}$. Similarly, $\bar{I}_- = \{i | f(A_i) > 0, A_i \in A_-\}$.
(4) Sort the set $\bar{I}_+$ by the absolute value of $f(A_{\bar{I}_+})$ and the set $\bar{I}_-$ by $f(A_{\bar{I}_-})$ respectively. We named the resulting sorted sets $\bar{S}_+$ and $\bar{S}_-$.
(5) Partition $\bar{S}_+$ and $\bar{S}_-$ into several subsets respectively such that each subset has nearly equal number of elements. That is, let $\phi \neq \bar{sp}_i \subset \bar{S}_+$, $\forall i, 1 \leq i \leq k$ where k is the number of subsets. $\bar{S}_+ = \bar{sp}_1 \cup \bar{sp}_2 \cup \ldots \cup \bar{sp}_k$ and $\bar{sp}_i \cap \bar{sp}_j = \phi, \forall i \neq j, 1 \leq i, j \leq k$. Similarly, $\bar{S}_- = \bar{sn}_1 \cup \bar{sn}_2 \cup \ldots \cup \bar{sn}_k$ and $\bar{sn}_i \cap \bar{sn}_j = \phi, \forall i \neq j, 1 \leq i, j \leq k$. Then, choose one point from each subset and add these points into $\bar{A}_0$ to generate a new reduced set in place of $\bar{A}_0$.
(6) Repeat *Step* (2) $\sim$ (5) until the validation set correctness has arrived at the threshold which is user pre-specified.
(7) Output the final classifier, $f(x) = 0$.

We showed the numerical results to demonstrate the efficiency of our algorithm in the next section.

Tenfold Test Set Correctness %						
Tenfold Computational Time, *Seconds*						
	Methods					
Dataset Size	SSRSVM		RSVM		SSVM	LIBSVM
$m \times n$	Correctness		Correctness		Correctness	Correctness
	Time *sec.*	$\bar{m}$	Time *sec.*	$\bar{m}$	Time *sec.*	Time *sec.*
Ionosphere	97.43		96.87		96.61	95.16
351×34	0.5620	20	0.6410	35	14.2190	0.1720
Cleveland Heart	86.20		85.94		86.61	85.86
297×13	0.5620	20.6	0.3750	30	7.2500	3.5460
BUPA Liver	74.80		74.87		74.47	73.64
345×6	0.4680	17.8	0.5000	35	10.1560	0.4620
Pima Indians	78.00		77.86		77.34	75.52
768×8	**0.9690**	17.4	1.5160	50	68.1560	26.8440
Mushroom	89.23		89.39		N/A	89.19
8124×22	**74.6870**	79	171.2500	215	N/A	171.4840
Face	98.51		98.39		N/A	98.15
6977×361	**73.8120**	42.2	115.2660	70	N/A	318.9400

Table 1. Tenfold cross-validation correctness results on six public datasets illustrate that the SSRSVM not only keep as good test set correctness as SSVM, RSVM and LIBSVM but less size of reduced set than RSVM. The bold type showed, when processing massive datasets, SSRSVM is faster than the other three methods. The computer ran out of memory while generating the full nonlinear kernel for the Mushroom and Face datasets. $\bar{m}$ denotes the average size of reduced set by running the SSRSVM algorithm. N/A denotes "not available" results because the kernel $K(A, A')$ was too large to store.

4 Experimental Results

All our experiments were performed on a personal computer, which utilizes a 1.47 GHz AMD Athlon(tm)XP 1700 PLUS processor and 256 megabytes of RAM. This computer runs on Windows XP operating system, with MATLAB 6 installed. We implemented the SSRSVM algorithm using standard native MATLAB codes. We used the Gaussian kernel in all our experiments. We test SSRSVM on six public available datasets which five from UC Irvine repository [8] and one from MIT CBCL [2]. In order to give a more objective comparison, we run tenfold cross-validation on each dataset. All parameters in our experiments were chosen for optimal performance on a tuning set, a surrogate for a test set.

The experimental results demonstrated that SSRSVM not only keeps as good test set correctness as SSVM, RSVM and LIBSVM but has less size of reduced set than RSVM. In addition, the results showed, when processing massive datasets, SSRSVM is faster than the other three methods. Table 1 summarizes the numerical results and comparisons of our experiments. It shows a comparison on the testing correctness and time cost among SSRSVM, RSVM, SSVM and LIBSVM algorithms. Observing this table, to run SSRSVM algorithm, the testing correctness is as good as RSVM.

5 Conclusions

In this paper, we propose a Systematic Sampling RSVM (SSRSVM) algorithm that selects the informative data points to form the reduced set while the RSVM used random selection scheme. This algorithm is inspired by the key idea of SVM, the SVM classifier can be represented by support vectors and the misclassified points are a part of support vectors. SSRSVM starts with an extremely small initial reduced set and adds a portion of misclassified points into the reduced set iteratively based on the current classifier until the validation set correctness is large enough. In our experiments, we tested SSRSVM on six public available datasets. It turns out that SSRSVM might automatically generate a smaller size of reduced set than the one by random sampling. Moreover, SSRSVM is faster than RSVM and much faster than conventional SVM under the same level of the test set correctness.

References

1. C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
2. MIT Center for Biological and Computation Learning. Cbcl face database (1), 2000. http://www.ai.mit.edu/projects/cbcl.
3. G. Fung and O. L. Mangasarian. Proximal support vector machine classifiers. In F. Provost and R. Srikant, editors, *Proceedings KDD-2001: Knowledge Discovery and Data Mining, August 26-29, 2001, San Francisco, CA*, pages 77–86, New York, 2001. Asscociation for Computing Machinery. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-02.ps.
4. Yuh-Jye Lee and Wen-Feng Hsieh. ϵ-ssvr: A smooth support vector machine for ϵ-insensitive regression. *IEEE Transactions on Knowledge and Data Engineering*, submitted 2003.
5. Yuh-Jye Lee and O. L. Mangasarian. RSVM: Reduced support vector machines. Technical Report 00-07, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, July 2000. Proceedings of the First SIAM International Conference on Data Mining, Chicago, April 5-7, 2001, CD-ROM Proceedings. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-07.ps.
6. Yuh-Jye Lee and O. L. Mangasarian. SSVM: A smooth support vector machine. *Computational Optimization and Applications*, 20:5–22, 2001. Data Mining Institute, University of Wisconsin, Technical Report 99-03. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/99-03.ps.
7. O. L. Mangasarian. Generalized support vector machines. In A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 135–146, Cambridge, MA, 2000. MIT Press. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-14.ps.
8. P. M. Murphy and D. W. Aha. UCI machine learning repository, 1992. www.ics.uci.edu/∼mlearn/MLRepository.html.
9. V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.