RANSAC-SVM for Large-Scale Datasets

Kenji NISHIDA,   Takio KURITA
Neuroscience Research Institute
National Institute of Advanced Industrial Science and Technology(AIST)
Central 2, 1-1-1 Umezono,
Tsukuba, IBARAKI 305-8568 JAPAN
kenji.nishida@aist.go.jp takio-kurita@aist.go.jp

Abstract

Support Vector Machines (SVMs), though accurate, are still difficult to solve large-scale applications, due to the computational and storage requirement. To relieve this problem, we propose RANSAC-SVM method, which trains a number of small SVMs for randomly selected subsets of training set, while tuning their parameters to fit SVMs to whole training set. RANSAC-SVM achieves good generalization performance, which close to the Bayesian estimation, with small subset of the training samples, and outperforms the full SVM solution in some condition.

1 Introduction

Although support vector machines (SVMs)[1, 2, 3] provide accuracy and generalization performance, applying them to large-scale problems are still difficult, since they require large computation and storage with the number of training vectors. Many previous work tried to relieve these difficulties, for example, Chapelle[4] examined the effect of optimization of SVM algorithm, which reduces the computation complexity from $O(n^3)$ (for naive implementation) to about $O(n_{sv}^2)$ (n_{sv} stands for the number of support vector), and Keerthi[5] proposed to reduce the number of kernels with forward stepwise selection and attained the computational complexity to $O(nd^2)$, where d stands for the number of selected kernels. Lin[6] proposed to select a random subset of the training set, though it could not reduce the number of basis functions to attain the accuracy close to full SVM solution. Demir[7] applied RANSAC[8, 9] algorithm to reduce the number of training samples for Relevance-Vector Machine (RVM) for remote sensing data. However, the classification accuracy was deteriorated against the Full-RVM solution in their experiment. Though the reason of the deterioration was not clear since detailed training procedure for Ransac-RVM was not described in [7], we suspect that the insufficient hyper-parameter setting for RVM caused the deterioration of classification accuracy.

In this paper, we propose RANSAC-SVM, in which random subsets of the training set is selected and small SVMs are trained for the subsets while hyper parameters are tuned to fit the SVMs to the full training set. Finally, the SVM with the lowest classification error for the full training set is adopted as the optimal classifier. We also employed Genetic Algorithm (GA) [10] to optimize the subsets of training samples from randomly sampled subsets. RANSAC-SVM achieves the accuracy close to the Bayesian estimation which gives the theoretical upper bound, and attains almost same accuracy of the full SVM solution with small subset of training samples.

The algorithm of RANSAC-SVM is described in next section followed by the description of its background, such as SVM and RANSAC. In section 3, we present our experimental results on artificial data. The consideration on computational requirements are discussed in section 4.

2 RANSAC-SVM

Since our RANSAC-SVM is based on the hyper parameter search on kernel-SVM and RANSAC (random sampling and consensus) algorithm, we first briefly describe these algorithms, then, we describe how to determine RANSAC-SVM at the end of this section.

2.1 Kernel-SVM and Hyper Parameter

When the classification function is given as

$$y = \mathrm{sign}(\boldsymbol{w}^{*T}\phi(\boldsymbol{x}) - h^*) \tag{1}$$

where function $sign(u)$ is a sign function, which outputs 1 when $u > 0$ and outputs -1 when $u \leq 0$, $\boldsymbol{w}$

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

stands for a weight vector of the input, and h stands for a threshold. $\phi(\boldsymbol{x})$ stands for a non-linear projection of an input vector, such as $\phi(\boldsymbol{x_1})^T\phi(\boldsymbol{x}_2) = K(\boldsymbol{x_1}, \boldsymbol{x}_2)$. K is called a *Kernel Function* and usually selected a simple function, such as Gaussian function

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(\frac{-||\boldsymbol{x}_1 - \boldsymbol{x}_2||^2}{G}\right). \qquad (2)$$

A SVM determines a separating hyperplane with maximal margin (distance), which is the distance between the separating hyperplane and a nearest sample. If the hyperplane is determined, there exists a parameter to satisfy $t_i\boldsymbol{w}^T\phi(\boldsymbol{x}) \geq 1, \ \ i = 1, \ldots, N$. This means that the samples are separated by two hyperplane H1: $\boldsymbol{w}^T\phi(\boldsymbol{x}_i) - h = 1$, H2: $\boldsymbol{w}^T\phi(\boldsymbol{x}_i) - h = -1$, and no samples exist between them. The distance between separating hyperplane and these hyperplanes are defined as $1/\|\boldsymbol{w}\|$.

A soft-margin SVM allows some training samples to violate the hyperplanes H1 and H2. When a distance from the H1 (or H2) is defined as $\xi_i/\|\boldsymbol{w}\|$ for the violating samples, the sum $\sum_{i=1}^{N}\xi_i/\|\boldsymbol{w}\|$ should be minimized. Therefore, a soft-margin SVM is defined as an optimization problem of the following evaluation function

$$L(\boldsymbol{w}, \boldsymbol{\xi}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i \qquad (3)$$

under a constraint

$$\xi_i \geq 0, t_i(\boldsymbol{w}^T\phi(\boldsymbol{x})_i - h) \geq 1 - \xi_i, (i = 1, \ldots, N) \quad (4)$$

where t_i stands for the correct class label for an input vector x_i, and C stands for a **cost** parameter for violating hyperplane H1 (or H2). Solving this problem with an optimal solution $\boldsymbol{\alpha}^*$, the classification function can be redefined as

$$y = \text{sign}(\sum_{i \in S}\alpha_i^* t_i K(\boldsymbol{x}_i, \boldsymbol{x}) - h^*). \qquad (5)$$

The samples are grouped with α_i^*; a sample x_i is classified correctly when $\alpha_i^* = 0$, when $0 < \alpha_i^* < C$ the sample x_i is also classified correctly and it locates on the hyperplane H1 (or H2) as a support-vector, if $\alpha_i^* = C$ the sample x_i becomes a support-vector but it locates between H1 and H2 with $\xi \neq 0$.

Two hyper-parameters C and G are usually assigned to minimize the cross-validation error for the compatibility between accuracy and generalization performance, and grid-search is often used to find optimal pair of C and G.

2.2 RANSAC

The RANSAC[8, 9] is an algorithm for robust fitting of models in the presence of many data outliers. The algorithm is described as follows.

Given a fitting problem with parameters $\boldsymbol{a}$, $(\boldsymbol{a}|a_1, \ldots, a_l)$, the estimation algorithm is described as follows:

1. selects N data items at random
2. estimates parameter $\boldsymbol{a}$
3. finds how many data items (of M) fit the model with parameter vector $\boldsymbol{a}$ within a user given tolerance. Call this K.
4. if K is big enough, accept fit and exit with success.
5. repeat step 1 to step 4, L times
6. fail if K does not satisfy the requirement after L iteration.

This algorithm assumes (1) The parameters can be estimated from N data items, (2) There are M data items in total.

When we apply RANSAC to the classification problem, we can take classification error E for consensus measure, instead of user given tolerance K.

2.3 Determining RANSAC-SVM

RANSAC-SVM applies the RANSAC algorithm to the training set for SVM (support vector machine). A number of small subsets are extracted from the training set by random sampling, then a SVM is trained for each small subset with the parameters to minimize the classification error on the full set of the training set. Finally, the SVM with the smallest error is selected by means of the best consensus. We also employed the multi point crossover method in genetic algorithm to optimize the sampled subsets. The brief algorithm for RANSAC-SVM with multi point crossover for Gaussian kernel is described below:

```
• Let $\boldsymbol{X}$ be the training set with $N$ samples, $\boldsymbol{X_l}$ $l = 1, 2, \ldots, L$ be the randomly sampled subsets
  of $\boldsymbol{X}$ with $M$ samples, and $p = 1, 2, \ldots, P$ be the generation of genetic algorithm.
• For $p = 1$ to $P$
    – For $l = 1$ to $L$
      Determine SVM classifier $Z_l$ for a subset $\boldsymbol{X_l}$ while searching hyper parameters $C_{opt}$ and
      $G_{opt}$ with following steps
        1. for $c = C_{min}$ to $C_{max}$,
           $g = G_{min}$ to $G_{max}$,
        2. Determine SVM classifier $Z_l$ with $c$ and $g$,
        3. predict whole training set $\boldsymbol{X}$ and compute classification error $E(Z_l(\boldsymbol{X}))$,
```

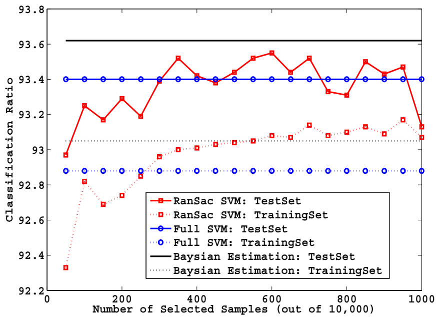

Figure 1. Classification Ratio against the Size of Subset

- sort sampled subsets according to the classification error $E(Z_l(\boldsymbol{X}))$,
- Determine the subsets for next generation by crossover operation, such as
  1. randomly select two (one pair of) subsets according to the classification error $E(Z_l(\boldsymbol{X}))$ as selection weights,
  2. Generate two (one pair of) subsets by randomly selecting samples from two subsets selected in previous step,
  3. repeat step 1 and 2, until L subsets are generated.

- Adopt Z_l with smallest E_l as a final classifier Z.

3 Experimental Results

3.1 Experiment with Artificial Data

We evaluated RANSAC-SVM with the artificial data in this experiment, the data are generated according to the mixture of four Gaussian distribution, two of which generate class 1 samples and others generates class 0 samples. 10,000 samples are generated for training set, and 10,000 samples are independently generated for test set. The contour in the figure indicates Bayesian estimate of class boundary, and classification ratio with Bayesian estimate are 93.05% for training set and 93.62% for test set. Classification ratio of full SVM solution, in which the parameters are determined by five-fold cross-validation, is 92.9% for training and 93.4% for test set.

Classification Ratio for Test set Figure 1 shows the classification ratio against the size of subset. Subset size varies 50 to 500 by 50, and one SVM with the highest classification ratio for full training set (10,000 samples) is selected out of one hundred crossover generation with one hundred training subsets to evaluate the test set. The result shows that the RANSAC-SVM attains almost same classification performance against Full SVM from small subset (0.2% degradation at 100 samples), and often outperform the Full SVM. Actually, the highest classification ratio of 93.5%, which is close to Bayesian estimation, for test set is attained by RANSAC-SVM at 600 samples.

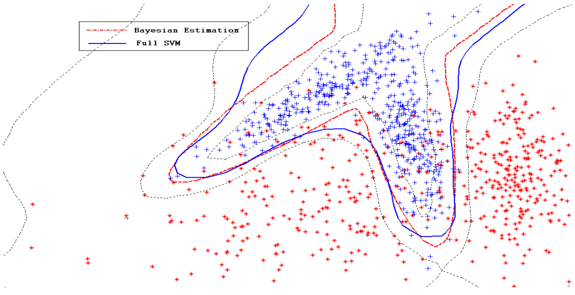

Figure 2. Classification Boundary for Full SVM

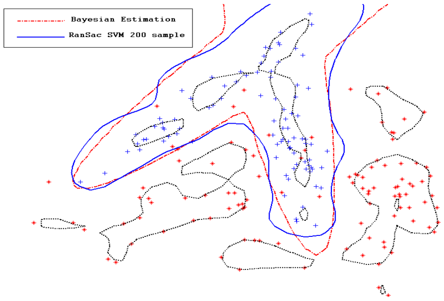

Figure 3. Classification Boundary for RANSAC-SVM (200 sample)

Classification Boundary Figure 2 shows the classification boundary for Full SVM solution, and figure 3 for RANSAC-SVM with subset size of 200. The classification boundary for Full SVM is determined by 1,791 support vectors (SVs), and classification boundary for 200 sample RANSAC-SVM is determined by 120 SVs (60% of the samples are assigned to SVs).The result also shows that 200 sample RANSAC-SVM gives a fairly good approximation of Bayesian estimation, in spite of the small number of SVs.

Number of Support Vectors Figure 4 shows the number of support vectors (SVs) against the size of sub-

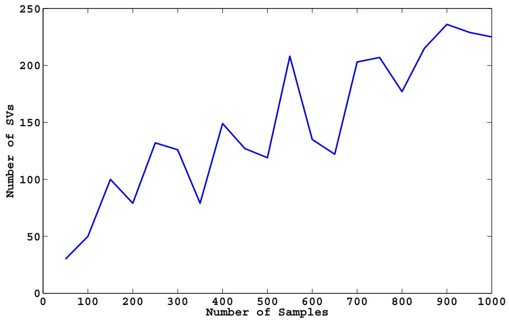

Figure 4. Number of Support Vectors against the Size of Subset

Table 1. Classification Ratio for MNIST

	Training set	Test set
Full SVM	99.99%	99.71%
RANSAC-SVM	99.55%	99.47%

set. The number of SVs in full SVM is 1,791 for the reference. The optimal classification ratio attained in a broad area of the subset size, and the number of SVs at subset size 600 was 135, which attained the highest classification ratio. This result indicates that RANSAC-SVM can provide accurate classifier with much smaller model compared with full SVM, therefore it can improve the generalization performance.

3.2 Experiment on MNIST data

The performance of RANSAC-SVM is examined on the the MNIST database of handwritten digits[11]. The 60,000 training samples are relabeled to the class of digit three and others for making the problem to two-class classification, and the subset size is determined as 13,844 from several trials. Table 5 shows the classification ratio of 60,000 training samples) and the classification ratio of 10,000 test samples. The parameters for full SVM are determined by 5-fold cross-validation. The result shows fairly good classification accuracy with RANSAC-SVM.

4 Discussions on Required Computation

Popular SVM libraries employ $O(n_{sv}^2)$ algorithm, therefore, reduction of number of SVs is important in large-scale problems. According to the results in figure 1 and figure 4, RANSAC-SVM reduces the computation requirement to about 1/170 for one randomly sampled subset.

5 Conclusion and Future Work

We proposed RANSAC-SVM which achieves high accuracy and generalization performance with a small number of support vectors. With reducing a number of SVs, RANSAC-SVM reduces the computation requirement at the same time. Therefore, we consider it is effective to determine classifiers for large-scale datasets. We are going to adopt particle swarm optimization (PSO) to improve the performance of optimal parameter search.

Acknowledgment

This work was partially supported by KAKENHI (19500163).

References

[1] V.N.Vapnik, *Statistical Learning Theory*, John Wiley & Sons (1998).

[2] B.Scholkopf, C.J.C.Burges, A.J.Smola, *Advances in Kernel Methods - Support Vector Learning*, The MIT Press, 1999.

[3] N.Cristianini, J.S-Taylor, *An Introduction to Support Vector Machines and other kernel-based learning methods*, Cambridge University Press, 2000.

[4] O. Chapelle, "Training a Support Vector machine in the Primal", in *Large-Scale Kernel Machines*, pp.29-50, The MIT Press, 2007.

[5] S.S. Keerthi, O. Chapelle, D. DeCoste, "Building SVMs with Reduced Classifier Complexity", in *Large-Scale Kernel Machines*, pp.251-274, The MIT Press, 2007.

[6] K.-M. Lin and C.-J. Lin, "A study on reduced support vector machines", *IEEE Transactions on Neural Networks*, Vol.14, pp.1449-1459, 2003.

[7] B. Demir, S. Erturk, "Hyperspectral Image Classification Using Relevance Vector Machines", *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS*, VOL. 4, NO. 4, pp. 586-590, OCTOBER 2007.

[8] M. A. Fischler, R. C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", *Comm. of the ACM*, Vol. 24, pp. 381-395, 1981.

[9] R.B.Fisher," The RANSAC (Random Sample Consensus) Algorithm", `http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/FISHER/RANSAC/#fischler` .

[10] D.E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning", Addison-Wesley, 1966.

[11] Y. Lecun, C.Cortes, "THE MNIST DATABASE of handwritten digits", `http://yann.lecun.com/exdb/mnist/` .