Support Vector Selection and Adaptation

Gülşen Taşkın Kaya
Computational Science and Engineering
Istanbul Technical University, Turkey
gulsen@be.itu.edu.tr

Okan K. Ersoy
School of Electrical and Computer Engineering
Purdue University, USA
ersoy@purdue.edu

Mustafa E. Kamaşak
Computer Engineering
Istanbul Technical University, Turkey
kamasak@itu.edu.tr

Abstract

The linear Support Vector Machine (SVM) is a very attractive approach in classification of linearly separable data. In order to handle non-separable data, SVM with a non-linear kernel is used. In nonlinear SVM, if a better kernel function is chosen, a higher classification performance is obtained. However, it is generally hard to decide which kernel type is optimal to be used with the given data, especially if the structure of the data are not known in advance. Moreover, it takes more computational time than the linear SVM. In order to overcome or reduce these difficulties, a new method based on support vector selection and adaptation (SVSA) is introduced and applied to both classification of synthetic and Colorado remote sensing data.

1. Introduction

SVM is based on determining an optimum hyperplane that separates the data into two classes with the maximum margin [1]. The hyperplane is obtained from the solution of a constrained quadratic programming (QP) problem. With linearly separable data, the support vectors exist at the margin. Classification is performed subsequently not by using the support vectors further, but by using the hyperplane dependent on the Lagrange coefficient corresponding to support vector. Only the support vectors have non-zero coefficients.

In nonlinear SVM, input space of data is transformed into a higher dimensional feature space by using a nonlinear kernel function, followed by linear SVM.

However, there are some difficulties with the nonlinear SVM approach [2]. As the training data grows in size, the constraint part for solving the QP problem becomes large, is very memory expensive, and decomposition methods become necessary to decompose the problem to parts and to solve the corresponding parts iteration by iteration [3].

Moreover, choosing of kernel function that best separate the classes is generally a crucial task for classification of linearly nonseparable classes. Even if the kernel function is determined for the data classified, finding of its parameter is also another issue in nonlinear SVM. In order to overcome this problem, cross validation algorithms are used for determining the parameters needed, but it also takes extra computational time.

Support vector selection and adaptation is a new method especially for nonlinearly separable data without choosing any kernel function; therefore, the problems caused by choosing and using kernel type in nonlinear SVM are gradually reduced, and what is more, the competitive classification performance with the SVSA are obtained in classification compared to nonlinear SVM.

Only the support vectors of linear SVM, which can be considered as the most important vectors closest to the decision boundary are used in SVSA, and some of which are selected with respect to their contribution to overall classification accuracy, which are called reference vectors. Afterwards, they are adaptively moved by Learning Vector Quantization (LVQ) with respect to training data. At the end of the adaptation process, the reference vectors are finalized, and used in classification with K-Nearest Neighbor (KNN) method.

In addition, a hybrid SVSA method is generated for classification of certain types of data. In the hybrid SVSA, both SVSA and linear SVM are used for classification depending on a given threshold value that can be determined by using the classification performance of training data.

During implementation, since the results of the linear SVM are also available, by utilizing this information, the hybrid model was generated by using consensus between the results of the linear SVM and the results of the SVSA.

2. Selection and Adaptation (SVSA)

Let $X = \{(x_1, \bar{x}_1), \ldots, (x_N, \bar{x}_N)\}$ represent the training data with $x_i \in R^p$, and the class labels $\bar{x}_i \in \{1, \ldots, M\}$. N, M and p denote the number of training samples, the number of classes and the number of features, respectively. After applying the linear SVM to the training data, the support vectors are obtained as

$$S = \left\{(s_i, \bar{s}_i) \;\middle|\; (s_i, \bar{s}_i) \in X \quad i = 1, \ldots, k\right\} \qquad (1)$$

$$T = \left\{(t_i, \bar{t}_i) \;\middle|\; (t_i, \bar{t}_i) \in X \setminus S \quad i = 1, \ldots, N-k\right\} \qquad (2)$$

where k is the number of support vectors, S is the set of support vectors with the class labels $\bar{s}$, and T is the set of training data vectors with the class labels $\bar{t}$, excluding the support vectors.

In the selection stage, the support vectors in the set S are classified with respect to the set T by using the KNN algorithm [4]. The labels of the support vectors are obtained as:

$$\bar{s}_i^p = \left\{\bar{t}_l \;\middle|\; l = \arg \min_{1 \le j \le N-k} \left\{\left\| s_i - t_j \right\|\right\} \quad i = 1, \ldots, k\right\} \qquad (3)$$

where $\bar{s}_i^p$ is the predicted label of i^{th} support vector.

Then, the misclassified support vectors are removed from the set S. The remaining support vectors are called reference vectors and constitute the set R:

$$R = \left\{(s_i, \bar{s}_i) \;\middle|\; (s_i, \bar{s}_i) \in S \text{ and } \bar{s}_i^p = \bar{s}_i \quad i = 1, \ldots, k\right\} (4)$$

The aim of selection process is to select the support vectors that describe the distinction of the classes as much as possible in the training set.

The reference vectors to be used for classification are next adaptively modified based on the training data in a way to increase the distance between the neighboring reference vectors with different class labels. The main idea of adaptation is that a reference vector causing a wrong decision should be further away from the current training vector, and the nearest reference vector with the correct class should be closer to the current training vector. Adaptation is achieved by using the LVQ algorithm [5,6] as described below.

Let x_j be one of the training samples with label y_j [7]. Assume that $\mathbf{r_w}(t)$ is the nearest reference vector to x_j with label y_{r_w}. If $y_j \neq y_{r_w}$ then the adaptation is applied as follows:

$$\mathbf{r_w}(t+1) = \mathbf{r_w}(t) - \eta(t)(\mathbf{x_j} - \mathbf{r_w}(t)) \qquad (5)$$

On the other hand, if $\mathbf{r}_l(t)$ is the nearest reference vector to x_j with label y_{r_l} and $y_j = y_{r_l}$ then

$$\mathbf{r}_l(t+1) = \mathbf{r}_l(t) + \eta(t)(\mathbf{x_j} - \mathbf{r}_l(t)) \qquad (6)$$

where $\eta(t)$ is a descending function of time called the learning rate. It is also adapted in time by

$$\eta(t) = \eta_0 e^{-t/\tau} \qquad (7)$$

where η_0 is the initial value of η, and τ is a time constant.

The adapted reference vectors are used for classification of the training and testing datasets. For this purpose, the KNN method is applied to classify the samples with respect to the reference vectors. The Euclidian distances from the input vector to the reference vectors are calculated, and classification is done based on the majority class of the K nearest reference vectors.

3. Hybrid SVSA

It is known that linear SVM gives the best classification accuracy for linearly separable data. According to the results obtained with some experiments done with both SVSA and SVM, it was observed that the SVSA as well as nonlinear SVM are not efficient classifiers especially with linearly separable data and very nonlinearly separable; therefore, the hybrid SVSA was developed.

For this purpose, the perpendicular distance to the hyperplane obtained by the linear SVM for each data sample is calculated based on the Euclidian distance. If the distance is greater than a given threshold, the data is classified by the linear SVM; otherwise the SVSA algorithm is applied. The schema for hybrid SVSA is shown in Figure 1.

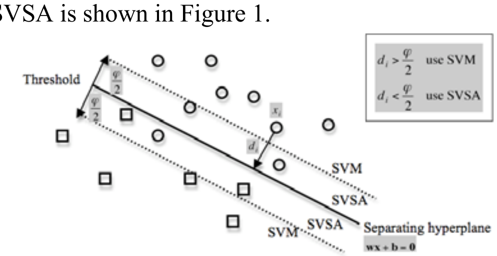

Figure 1. Schema of hybrid SVSA.

4. Experiments with Synthetic Data

In our experiments, we first generated different types of synthetic data with different types of nonlinearity in order to compare the classification performance of the proposed method with the SVM. Four types of example were generated as banana shaped data and the data created by using given mean vectors and covariance matrices in a way to provide nonlinearity [8]. All the datasets are shown in Figure 2.

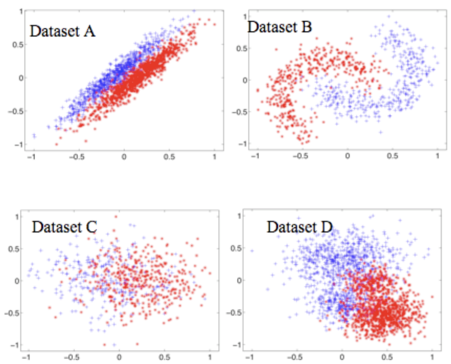

Figure 2. The synthetic datasets used in the experiments.

Scaling data before applying SVM is an important step [9]. The main advantage of scaling is to avoid features in larger numeric ranges, and dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during computations. In this work, each feature of a data vector was linearly scaled to the range [-1,+1] before doing experiments.

For nonlinear SVM, cross-validation within the original datasets was utilized to provide a nearly unbiased estimate of the prediction error rate. The performance of classifying the datasets was evaluated using 10-fold cross-validation [10]. There are two parameters while using RBF kernels: kernel parameter γ and penalty parameter C. These were also estimated by cross-validation.

After tuning of parameters for kernels and scaling of data, support vectors are determined by the linear SVM. The selection and adaptation process is applied with respect to the training data, and adapted reference vectors are obtained for each dataset in Figure 2.

In the selection stage, each support vector is first classified by 1-KNN with respect to the training data excluding the support vectors, and then the misclassified support vectors are excluded from the set of reference vectors. During the adaptation, the remaining support vectors called reference vectors are adapted based on all the training data by means of the LVQ.

Table 1. The classification accuracies on the testing data with respect to all the methods.

Dataset	METHODS				
	SVM	**N-1**	**N-2**	**SVSA**	**Hybrid**
A	93.9	93.8	93.6	92.3	94.0
B	83.8	97.0	82.7	95.9	96.6
C	72.0	70.5	72.0	69.4	72.1
D	84.3	87.3	83.9	86.3	86.5

In Table 1, the methods N-1 and N-2 corresponds to nonlinear SVM with RBF and polynomial kernel, respectively. As observed in Table 1, the classification accuracy of nonlinear SVM depends on the choice of the kernel type. For example, nonlinear SVM with radial basis kernel is better than nonlinear SVM with polynomial kernel in terms of classification performance.

According to the results obtained by applying all the algorithms to all the datasets, it was observed that the classification performance of the hybrid method was better than all the other methods with linearly separable data and the data with extreme nonlinearity. If the data was not linearly separable, the SVSA was competitive with the nonlinear SVM and better than the linear SVM in terms of classification accuracy.

5. Experiment 2 : Colorado Dataset

Classification was performed with the Colorado dataset [11] consisting of four data sources: Landsat MSS data (four spectral data chanels), elevation data (in 10-m contour intervals, one data channel), slope data ($0-90^0$ in 1^0 increments, one data channel), and aspect data ($1-180^0$ in 1^0 increments, one data channel).

Each channel comprised an image of 135 rows and 131 columns, and all channels were spatially co-registered in Colorado. It has ten ground-cover classes which are listed in Table 2. One class is water; the others are forest types. It is very difficult to distinguish among the forest types using Landsat MSS data alone since the forest classes show very similar spectral response.

Table 2. Training and testing samples of the Colorado dataset.

Class #	Information Class	Training Size	Testing Size
1	Water	408	195
2	Colorado Blue Spruce	88	24
3	Mountane/ Subalpine meadow	45	42
4	Aspen	75	65
5	Ponderosa Pine 1	105	139
6	Ponderose Pine/Douglas Fir	126	188
7	Engelmann Spruce	224	70
8	Douglas Fir/White Fir	32	44
9	DouglasFir/PonderosaPine/Aspen	25	25
10	Douglas Fir/White Fir/Aspen	60	39
	Total	1188	831

45 experiments were done with the Colorado dataset for binary classification, and the overall training and testing classification accuracy of the binary classification are shown in Table 3.

Table 3. Classification accuracies for training and testing data of Colorado.

Data	SVM	NSVM(1)	NSVM(2)	SVSA	Hybrid
Training	92.15	94.99	87.10	**96.88**	96.30
Testing	78.87	79.92	77.17	82.95	**84.36**

We obtained higher classification accuracies with the SVSA in comparison to the linear and nonlinear SVM. The performance of the hybrid SVSA was also better than the SVSA method in terms of overall classification accuracy. The classification performance of each binary class is shown in the Technical Report [12]. The performance of hybrid SVSA depends on choice of a threshold value determined as to give maximum classification performance in training data.

If speed performance of the SVSA method is considered, it takes a longer time than SVM because of adaptation of support vectors in addition to getting support vectors. On the other hand, it requires less time than nonlinear SVM since our method do not contain time consuming kernel processes. The advantage of the SVSA method is that the classification performance of nonlinear SVM can be reached with faster calculations.

6. Conclusion

In this study, we addressed the problem of classification of synthetic and remote sensing data using the proposed support vector selection and adaptation method and hybrid SVSA method in comparison to linear and nonlinear SVM.

The SVSA method consists of selection of the support vectors which contribute most to the classification accuracy and adaptation of them based on the class distributions of the data. It was shown that the SVSA method gives competitive classification performance in comparison to the linear and nonlinear SVM with both synthetic data and real world data.

With linearly separable data, it was observed the linear SVM is better than other methods in terms of classification accuracy. The hybrid model (hybrid SVSA) was developed to improve classification performance further with such data. In the hybrid SVSA, both linear SVM and SVSA are used to classify the data based on a given threshold value. It was observed that the hybrid SVSA is quite effective in classification of such data.

7. References

[1] A. Shmilovici, *The Data Mining and Knowledge Discovery Handbook*, Springer, 2005.

[2] Yue Shihong, Li Ping, Hao Peiyi, "Svm Classification :Its Contents and Challenges", *Appl. Math. J. Chinese Univ. Ser. B*, Vol.18(3), 332-342, 2003.

[3] John Platt, "Using Analytic QP and Sparseness to Speed Training of Support Vector Machines", *MIT Press*, 557-563, 1999.

[4] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13(1), pp.21–27, 1967.

[5] T. Kohonen, "Learning vector quantization for pattern recognition," *Tech. Rep., TKK-F-A601, Helsinki University of Technology,* 1986.

[6] T. Kohonen, J. Kangas, J. Laaksonen, and K. Torkkola, "Lvqpak: A software package for the correct application of learning vector quantization algorithms," *Neural Networks, IJCNN., International Joint Conference*, vol. 1, pp. 725–730, 1992.

[7] N. G. Kasapoğlu and O. K. Ersoy, "Border Vector Detection and Adaptation for Classification of Multispectral and Hyperspectral Remote Sensing", *IEEE Transactions on Geoscience and Remote Sensing*, Vol: 45-12, pp: 3880-3892, December 2007.

[8] R.P.W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D.M.J. Tax, S. Verzakov, "A Matlab Toolbox for Pattern Recognition", *Delft University of Technology, PRTools4.1,* 2007.

[9] Jun Cai and Yanda Li, "Classification of Nuclear Receptor Subfamilies with RBF Kernel in Support Vector Machine", *Springer-Verlag Berlin Heidelberg,* 680–685, 2005.

[10] Bengio, Y., Grandvalet, Y, "No unbiased estimator of the variance of k-fold cross-validation", *Journal of Machine Learning Res*, Vol.5 pp. 1089-1105, 2004.

[11] J. A. Benediktsson, P. H. Swain, O. K. Ersoy, "Neural Network Approaches versus Statistical Methods in Classification of Multisource Remote Sensing-Data," *IEEE Transactions Geoscience and Remote Sensing,* Vol. 28, No. 4, pp. 540-552, July 1990.

[12] G. Taşkın Kaya and O. K. Ersoy, "Support Vector Selection and Adaptation for Classification of Remote Sensing Images", *TR-ECE-09-02*, Purdue University Technical Report, February 2009.