A New Double Bagging via the Support Vector Machine with Application to the Condition Diagnosis for the Electric Power Apparatus

Faisal M. Zaman Hideo Hirose *

Abstract—**The aggregation of multiple unstable classifiers often leads us to reduce the misclassification rates substantially in many applications and benchmark classification problems. We propose here a new variant of the double bagging, where we use the support vector machine as the additional classifier built on the out-of-bag samples. The underlying basic classifier is the decision tree. We use four kernel types; linear, polynomial, radial basis and sigmoid kernels, expecting the new classifier perform better. The major advantages of the proposed method is that, 1) it has robustness against many messy real data cases, 2) the generation of support vectors in the first phase facilitate the decision tree to classify the objects with higher accuracy, resulting a significant reduction in misclassification rates in the second phase. We also used subsamples with bootstrap samples, where 50% samples are used for the training samples without replacement, expecting larger out-of-bag samples. We have applied the proposed method to a real case, the condition diagnosis for the electric power apparatus; the feature variables are the maximum likelihood parameters in the generalized normal distribution, and these variables are composed from the partial discharge patterns of electromagnetic signals by the apparatus. Comparing to other well-known ensemble classifiers, the double bagging with the support vector machine classifier with radial basis kernel performs best among all the classifiers.**

Keywords: ***Support vector machine, double bagging, CART, condition diagnosis, electric power apparatus***

1 Introduction

The support vector machine (SVM) is a new and promising classification and regression technique proposed by Vapnik and his group at AT&T Bell Laboratories [5]. The SVM learns a separating hyperplane to maximize the margin and to produce a good generalization ability [4]. Recent theoretical research work has solved the existing difficulties of using the SVM in practical applications [14], [19]. The capability of SVM to have competitive generalization error than other classification methods and ensemble methods has also been checked [18], [8].

The idea of the SVM ensemble has been proposed in [23]. They used the boosting technique to train each individual SVM and took another SVM for combining several SVMs. In [15] authors proposed to use the SVM ensemble based on the bagging and boosting techniques. In bootstrapping (bagging), each individual SVM is trained over the randomly chosen training samples via the bootstrap technique. In boosting, the training samples for each individual SVM is chosen according to updating the probability distribution (related to error) for samples. Then, the independently trained several SVMs are aggregated in various ways such as the majority voting, the least square error based weighting, and the double-layer hierarchical combining. In [22] authors used a novel aggregation rule SEN (selective ensemble) in constructing LS-SVM ensemble. In [17] authors used subsampling to build SVM ensembles to increase the diversity of the ensemble. In this paper we have used SVM as the additional classifier model in an ensemble method called the double bagging. In double bagging an additional classifier model is built on the out-of-bag samples and then this model is trained on both the inbag samples and test set to extract additional predictors for both in building the ensemble and testing it in the test set. As the SVM is a maximum margin classifier, which construct optimum separating hyperplane between the classes (for binary classification), we intended to use it in the first phase of the ensemble to attain the support vectors consisting the discriminative information between the classes and then use them as the additional predictors to constructs the decision tree ensemble in the second phase. These support vectors are also used in the testing the decision tree ensembles. This procedure ensures a possibility of maximum separation of the classes so that the decision tree ensemble can perform more accurately in discriminating the classes.

In this paper we have applied the double bagging via SVM in classifying the type of partial discharge (PD)

*Manuscript received December 15, 2008. The authors would like to thank Dr. Okabe and Mr. Tsuboi for their cooperation and valuable comments. This research was supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (20510159), and by Tokyo Electric Power Co.
Correspondence: Department of Systems Design and Informatics, Kyushu Institute of Technology, Fukuoka 820-8502, Japan Tel/Fax: +81(948)29-7711/7709, Email: hirose@ces.kyutech.ac.jp

patterns in a model gas insulated switch gear (GIS) as a typical electric power apparatus. For condition monitoring purposes, it is considered to be important to identify the type of defects when monitoring discharge activities inside an insulation system. In the paper [10] authors first proposed to use the decision tree as a classification tool for diagnosing because it provides the if-then-rule in visible form, and thus we may have a possibility to connect the physical phenomena to the observed signals. In [11] authors used several ensemble methods in classifying the defect patterns in the electric power apparatuses. In [16] authors applied a SVM ensemble for fault diagnosis, based on the genetic algorithm (GA). They used the GA in order to find more accurate and diverse ensemble.

The paper is organized as follows. In section 2, we have introduced the SVM with a non-mathematical introduction and mathematical formulation, and then we have introduced some popular kernels used in the SVM. In section 3 we have introduced the double bagging and give a brief description of the implementation of the double bagging via the SVM. In section 4, the main topic of this paper, we described the new double bagging via the SVM. In section 5 we have described the characteristics and extraction method used for the GIS data of the experiments in this paper. In section 6 the experimental setup of the study is explained, where we have compared the performance of the double bagging (with subbagging) SVM, with other ensemble methods, such as the bagging, the adaboost.M1, the logitboost and th double bagging (with subbagging) with LDA and k-NN. In section 7 the results of the experiments are explained and discussed. In section 8, conclusion of the study is stated.

2 Support Vector Machine (SVM)

The SVM models were originally defined for the classification of linearly separable classes of objects. Such an example is presented in Figure 1. For these two-dimensional objects that belong to two classes (class $+1$ and class -1), it is easy to find a line that separates them perfectly. For any particular set of two-class objects, an SVM finds the unique hyperplane having the maximum margin (denoted with δ in Figure 1). The hyperplane H_1 defines the border with class $+1$ objects, whereas the hyperplane H_2 defines the border with class -1 objects. Two objects from class $+1$ define the hyperplane H_1, and three objects from class -1 define the hyperplane H_2. These objects, represented inside circles in Figure 1, are called the support vectors. A special characteristic of the SVM is that the solution to a classification problem is represented by the support vectors that determine the maximum margin hyperplane.

The SVMs aim at minimizing an upper bound of the generalization error through maximizing the margin between the separating hyperplane and the data. This can be regarded as an approximate implementation of the structural risk minimization (SRM) principle, which endows with good generalization performances independent of underlying distributions [14]. The SVMs algorithms are based on parametric families of separating hyperplanes of different Vapnik-Chervonenkis dimensions (VC dimensions). The SVMs can effectively and efficiently find the optimal VC dimension and an optimal hyperplane of that dimension simultaneously to minimize the upper bound of the expected risk. Usually the classification decision function in the linearly separable problem is represented by

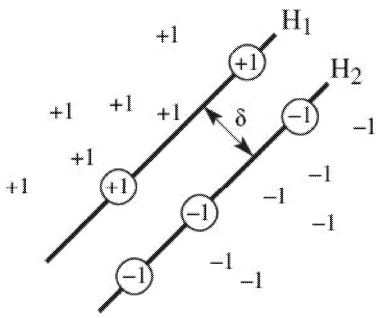

Figure 1: Maximum Separation Hyperplane.

$$f_{w,b} = sign(w \cdot x + b).$$

Thus, to find a hyperplane with minimum VC dimension, we need to minimize the norm of the canonical hyperplane $||w||$. Also the distance between the hyperplane H_1 and H_2 showed in Figure 1 is,

$$\delta = \frac{2}{||w||}.$$

Consequently, minimizing the norm of the canonical hyperplane $||w||$ is equivalent to maximizing the margin δ between H_1 and H_2, Figure 1. The purpose of implementing SRM for constructing an optimal hyperplane is to find an optimal separating hyperplane that can separate the two classes of training data with maximum margin. In Figure 1, the support vectors construct these optimal hyperplanes. Hence the optimal hyperplane separating the training data of two separable classes is the hyperplane that satisfies,

$$Minimize : F(w) = \frac{1}{2} w^T w, \; y_i(w \cdot x_i + b) \geq 1.$$

This is a convex, quadratic programming (QP) problem with linear inequality constraints. It is hard to solve the inequality constraint optimization problem directly. The most common way to deal with optimization problems with inequality constraints is to introduce Lagrange multipliers to convert the problem from primal space to dual space and then solve the dual problem. For the linearly non-separable case, the minimization problem needs to

be modified to allow the misclassified data points. This modification results in a soft margin classifier that allows but penalizes errors by introducing a new set of variables $\xi_{i=1}^{l}$ as the measurement of violation of the constraints.

$$Minimize : F(w) = \frac{1}{2}w^T w + C(\sum_{i=1}^{L}\xi_i)^k,$$

$$y_i(w^T\phi(x) + b) \geq 1 - \xi_i,$$

where C and k are used to weight the penalizing variables ξ_i, and $\phi(\cdot)$ is a nonlinear function which maps the input space into a higher dimensional space. Minimizing the first term in the above QP is corresponding to minimizing the VC dimension of the learning machine and minimizing the second term in QP controls the empirical risk. Therefore, in order to solve problem the QP, we need to construct a set of functions, and implement the classical risk minimization on the set of functions. Here, a Lagrangian method is used to solve the above problem. Then the QP can be written as after introducing L non-negative Lagrangian multipliers $\alpha_1, \alpha_2, \ldots, \alpha_L$,

$$Maximize : L(\alpha),$$

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{L}\alpha_i - \sum_{i=1}^{L}\sum_{j=1}^{L}\alpha_i\alpha_j y_i y_j \phi(x)^T\phi(x_i)^T,$$

subject to

$$\sum_{i=1}^{L}\alpha_i y_i = 0; \sum_{i=1}^{L}\alpha_i \leq C;\ \sum_{i=1}^{L}\alpha_i \geq 0.$$

After the optimum Lagrange multipliers α_i have been determined, we can compute the optimum coefficient vector w^* and the optimal offset b^*. The solution is given by

$$f(x) = sign(\sum_{i=1}^{L} y_i\alpha_i^*(x) + b^*),$$

where $\alpha_i^*(x) = \alpha_i y_i K(x, x_i)$, and $K(x, x_i) = \phi(x) \cdot \phi(x_i)$. ($K(x, x_i)$ can be simplified by kernel trick [20]).

2.1 Kernels used in SVM

In this subsection, we present the most used SVM kernels. These functions are usually computed in a high-dimensional space and have a nonlinear character.

Linear (dot) kernel: The inner product of x_i and x_j defines the linear (dot) kernel

$$K(x_i, x_j) = x_i \cdot x_j.$$

This is a linear classifier, and it should be used as a test of the nonlinearity in the training set, as well as a reference for the eventual classification improvement obtained with nonlinear kernels.

Polynomial Kernel: The polynomial kernel is a simple and efficient method for modeling nonlinear relationships:

$$K(x_i, x_j) = (1 + x_i \cdot x_j)^d.$$

Gaussian Radial Basis Function: Radial basis functions (RBF) are widely used kernels, usually in the Gaussian form:

$$K(x_i, x_j) = \exp(\frac{||x - \mu||^2}{2\sigma^2}).$$

The parameter σ controls the shape of the separating hyperplane.

Exponential Radial Basis Function:

$$K(x_i, x_j) = \exp(\frac{||x - \mu||}{2\sigma^2}).$$

Neural (tanh, sigmoid) kernel: The hyperbolic tangent (tanh) function, with a sigmoid shape, is the most used transfer function for artificial neural networks. The corresponding kernel has the formula:

$$K(x_i, x_j) = \tanh(ax_i \cdot x_j + b).$$

Anova Kernel: A useful function is the anova kernel, whose shape is controlled by the parameters γ and d:

$$K(x_i, x_j) = (\sum \exp(\gamma(x_i - x_j)))^d.$$

In this paper we have used linear, polynomial, Gaussian radial basis function and sigmoid kernel.

3 Double Bagging

Drawing a random sample of size N from the empirical distribution, a bootstrap sample of size N covers approximately 2/3 of the observations of the learning sample. The observations, which are not in the bootstrap sample, are called out-of-bag sample and may be used for estimating the misclassification error or for improved class probability estimates. In the double bagging framework proposed by Hothorn and Lausen [12], the out-of-bag sample is used to perform an additional classifier. In the setup of Hothorn and Lausen the double-bagging uses the values of the linear discriminant functions trained on the out-of-bag sample as additional predictors for bagging classification trees only. The discriminant variables are computed for the bootstrap sample and a new classifier is constructed using the original variables as well as the discriminant variables. In the experiments of [12] the double-bagging was performed as follows:

(1) Draw B random samples $L^{(1)}, ...L^{(B)}$ with replacement from the training set L and let $X^{(b)}$ denote the matrix of predictors $x_1^{(b)}, \ldots, x_N^{(b)}$ from $L^{(b)}$.

(2) Compute an LDA $Z^{(b)}$, using the out-of-bag sample $L^{-(b)}$, that gives a matrix $W^{(b)}$ where the columns are the coefficients of the linear discriminant functions.

(3) Construct the combined classifier C using the original variables as well as the discriminant variables of the bootstrap sample $(L^{(b)}, X^{(b)}W^{(b)})$

(4) Iterate steps (2) and (3) for all B bootstrap samples.

A new observation x is classified by, ' average ' rule using the predictions of all classifiers $C((x, xZ(b)), (L(b), X(b)W(b)))$ for $b = 1, 2, ...B$. Using the out-of-bag sample for the LDA, the coefficients are of the discriminants are estimated by an independent sample thus it avoiding the overfitted discriminant variables in the tree growing process. Furthermore it ensures that the training sample for the LDA is small and therefore the LDA becomes less stable and in the typical situation bagging can lead to stabilization. In double bagging instead of the LDA the other stable classifiers like, Nearest Neighbor (NN), Linear Logistic and SVM can be used as the additional classifier models.

4 Double Bagging with SVM

The underlying idea of double bagging is in the spirit of Breiman [3], " Instead of reducing the dimensionality, the number of possible predictors available to the classification trees is enlarged and the procedure is stabilized by bootstrap aggregation ". In this algorithm a classifier model is constructed for each bootstrap sample using an additional set of observations, the out-of-bag sample. The prediction of this classifier is computed for the observations in the bootstrap sample and is used as additional predictors for a classification tree. The trees implicitly select the most informative predictors. The procedure is repeated sufficiently often and a new observation is classified by averaging the predictions of the multiple trees. So we see that performance of the double bagging solely depends on how much informative (or discriminative) are the additional predictors built on the out-of-bag samples. Keeping this in mind we used the SVM as the additional classifier model as SVC (support vector classifier) are maximum margin classifier, i.e., the support vectors construct the separating hyperplane with the maximal margin between the classes (for example in 2-class problem), it has an extra advantage regarding automatic model selection in the sense that both the optimal number and locations of the support vectors are automatically obtained during training [21]. So in the double bagging the use of SVM will ensure that the additional predictors (the support vectors) extracted after training the SVM models on the inbag samples, constructed on out-of-bag samples, will be the observations with maximal margin between the classes. Henceforth it will allow us the decision tree based on the combined training sample (i.e., the bootstrap samples and the support vectors) to split the data more accurately and therefore will have an improved performance.

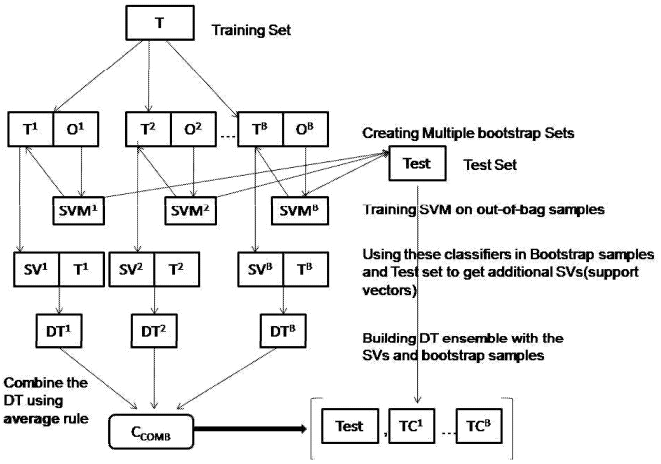

Figure 2: Architecture of Double Bagging via SVM

So we see from Figure 2 that in the first phase of training step SVMs are constructed using the out-of-bag samples, then to get additional predictors, these SVMs are used in the bootstrap samples to get the support vectors (SV^b). In the second phase an ensemble of decision tree(DT^b) is built using these SVs and the bootstrap samples (T^b). The SVMs are also used in the test set to enlarge the size of the test set by the test support vectors (TSV^b). Then these TSVs are included with test set as the additional predictors.

The SVM has been known to show a good generalization performance and is easy to learn exact parameters for the global optimum [4]. Because of these advantages, their ensemble may not be considered as a method for improving the classification performance greatly. However, since the practical SVM has been implemented using the approximated algorithms in order to reduce the computation complexity of time and space, a single SVM may not learn exact parameters for the global optimum. Sometimes, the support vectors obtained from the learning is not sufficient to classify all unknown test examples completely. So, we cannot guarantee that a single SVM always provides the global optimal classification performance over all test examples. This allows us to use the SVM in bagging; as in bagging the base classifiers should be unstable to get better performance. In addition to this as in the double bagging, in the first phase, the SVCs built on out-of-bag samples, are smaller in size will have a rather unstable performance, will result in as the perfect unstable predictor for the bagging procedure in the second phase of the method.

As the success of the double bagging mostly lie on the classifier model build on the out-of-bag samples, to en-

sure large out-of-bag samples we used subsamples instead of the bootstrap samples, i.e., use 50% of each sample without replacement; we call this the double subbagging. This modification ensures that the learning samples for the additional classifier model always contain half of the observations of the training sample.

As our data consist of three classes we need to modify the classification strategy of each SVM. Another method is called the one-against-one method [24]. When the number of classes is k, this method constructs $k(k-1)/2$ SVM classifiers. The ijth SVM is trained from the training samples where some examples contained in the ith class have, " $+1$ ", labels and other examples contained in the jth class have, " -1 " labels. The class decision is performed in the following way. The decision is based on the " max wins " voting strategy, in which $k(k-1)/2$ binary SVM classifiers will vote for each class, and the winner class will be the class having the maximum votes.

5 Data

The data used in the experiments in this paper is a transformed version of the electromagnetic signals measured by the sensors in the substations, since the stochastic signals measured cannot be used as they are of too abundant information, they are once transformed into ϕ-V-n (phase resolved PD) patterns. Then generalized normal distribution fitting [9] is used in order to acquire accurate diagnosis of the faults. We assume three classes for possible abnormal conditions in the GIS; 1) the metal is attached on the high voltage side conductor (abridged by HV from now on), 2) the metal is attached on the earth side tank (TK), and 3)the metal is freely movable (FR). The numbers of the observed samples are, 150, 377, 126, for HV, TK, FR. Here the dataset consist of MLEs for 4 parameter (2 parameters for phase 0-180 and 2 parameters for phase 180-360) of the generalized normal distribution (GND) fitted to the observed PD patterns, and these are used as feature variables.

6 Experimental Setup

To evaluate the efficacy of the proposed double bagging via SVM ensemble we have performed three different ensemble methods, bagging [2], adaboost.M1 [6] and logitboost [7], with the double bagging (with subbagging) with LDA, 5-NN and 10-NN classifier models. We have used CART [1] in bagging, double bagging and adaboost.M1 and decision stump (DS) [13] in adaboost.M1 and logitboost as the base classifier. We used here 2-node decision stump in case of adaboost.M1 and logitboost and 3-node decision stump in case of adaboost.M1. Since DS is more efficient as a weak classifier to be used in boosting algorithms, we used it in the experiments. As there are three classes in the data set, we used here 3-node DS and 2-node DS. The results are shown in Table 1. In double bagging with SVM we have used four kernels (as stated in section 2.1) linear, polynomial, radial basis function and sigmoid. The main idea behind this is to check which kernel produces better diagnosis results. In the experiments we have split the dataset into two independent parts, one for training, the training set (50% of the dataset) and the test set (remaining 50% of the data). We perform this splitting 5, 10, 25 and 50 times in order to avoid the dependence on the splitting. The ensemble size in case of bagging and double bagging is B=50 and 100; in case of adaboost.M1 and logitboost it is M = 25, 50 and 100. We report here only the better performing ensembles. The notations used in the paper for the classifiers:

CART: Single CART,

BCART: Bagged CART,

DBLDA: Double bagging with LDA

DB5NN: Double bagging with 5-NN

DB10NN: Double bagging with 10-NN

DSBLDA: Double subbagging with LDA

DSB5NN: Double subbagging with 5-NN

DSB10NN: Double subbagging with 10-NN

DBLINSV: Double bagging with linear kernel SVM

DBPOLYSV: Double bagging with polynomial kernel SVM

DBRBFSV: Double bagging with RBF kernel SVM

DBSIGSV: Double bagging with sigmoid kernel SVM

DSBLINSV: Double subbagging with linear kernel SVM

DSBPOLYSV: Double subbagging with polynomial kernel SVM

DSBRBFSV: Double subbagging with RBF kernel SVM

DSBSIGSV: Double subbagging with sigmoid kernel SVM

ADACART: Adaboost.M1 CART

ADADS2: Adaboost.M1 Decision Stump with 2-node

ADADS3: Adaboost.M1 Decision Stump with 2-node

LOGITDS: LogitBoosted Decision Stump

7 Results

In this section we present the results of the experiments. We have reported here the lowest test errors of the classifiers, with the training error and the corresponding en-

semble size (B or M) with the number of random partitions (R). The best result is printed in bold. In Table 1 we have presented the diagnosis results of CART, BCART, ADACART, ADADS2, ADADS3 and LOGITDS.

In Table 1 we see that the performance of BCART is (misclassification error 4.5%) better than single CART and adaboost.M1 and logitboost. We see that 3-node DS has highest the prediction accuracy among the boosted algorithms. We see here also that the best results are occurred with random partitions 5 and 10, while for the boosted algorithms, nothing can be deduced on the optimum ensemble size (M).

Table 1: Misclassification error of BCART, ADACART, ADADS2, ADADS3 and LOGITDS.

data	GND fitted		
classifiers	train error	test error	condition
CART	0.061773	0.087461	
BCART	0.00000	0.04587	B=50, R =5
ADACART	0.015337	0.051988	M=100, R =10
ADADS2	0.068712	0.097248	M=50, R =10
ADADS3	0.012577	0.047095	M=100, R =5
LOGITDS	0.044192	0.071976	M=25, R =5

In Table 2 we have presented the results of DB5NN, DB10NN, DBLDA, DSB5NN, DSB10NN and DSBLDA. Here we see that DBLDA has the highest accuracy than the other classifier (accuracy 96.23%); though DB5NN has 96.21% accuracy. We see here that the accuracy has increased (or misclassification error is decreased) than the best result of Table 1 which was acquired by BCART (accuracy 95.5%). For the number of partitions (R), we can say that for these classifiers the optimum values is R = 50. Unlike Table 1, we can say that for these classifiers the better performing ensemble size is B = 100. We also see that with the introduction of subsamples (double subbagging) the accuracy is increased for double subbagging with 5-NN and 10-NN.

Table 2: Misclassification error of double bagging (and subbagging) with LDA, 5-NN and 10-NN Data GND fitted Data.

data	GND fitted		
classifiers	train error	test error	condition
DB5NN	0.000000	0.039083	B=100, R =50
DB10NN	0.000000	0.045872	B=100, R =50
DBLDA	0.000000	0.037676	B=100, R =50
DSB5NN	0.002147	0.03792	B=100, R =25
DSB10NN	0.003436	0.043425	B=100, R =50
DSBLDA	0.001104	0.04159	B=100, R =50

In Table 3 we have presented the results of the double bagging (and subbagging) via the SVM. We see here that the better performing SVM for this data is the RBF, as the DBRBF and DSBRBF have the lowest misclassification error (0.03211, 0.02935) among all the classifiers here. The main reason could be for the success of the RBF kernel to perform very well is that, we used the Gaussian RBF instead of the exponential RBF and as the features of this dataset are the fitted parameters of generalized normal distribution, the kernel function mapped the features in the best way than the other kernel methods. We see in this table that all the classifiers instead of DSBPOLYSV and DSBLINSV produced error nearly the same or lower than the classifiers in Table 1 and 2. For these classifiers the better ensemble size can be set to B = 100 for subsampled ensembles and B = 50 for bootstrapped ensembles.

Table 3: Misclassification error of double bagging (and subbagging) with LIN-SVM, POLY-SVM, RBF-SVM and SIGMA-SVM.

data	GND fitted		
classifiers	train error	test error	condition
DBLINSV	0.000000	0.03792	B=50, R =10
DBPOLYSV	0.000000	0.034250	B=100, R =5
DBRBFSV	0.000000	0.03211	B=50, R =10
DSIGMSV	0.00000	0.03795	B=100, R =5
DSBLINSV	0.00721	0.04220	B=100, R =10
DSBPOLYSV	0.00521	0.03852	B=100, R =5
DSBRBFSV	0.00243	0.02935	B=100, R =10
DSIGMSV	0.002147	0.03552	B=100, R =5

8 Conclusions

CART searches for partitions in the multivariate samples space, which may be seen as higher-order interactions or homogeneous subgroups defined by some combination binary splits of the predictors. On the contrary the SVC construct the optimum separating hyperplane which maximize the margin between the classes (in binary classification). To build an ensemble of classifier with better generalization performance we combine these two methods.

A new SVM ensemble method has been proposed in this study, being a variant of another ensemble method named double bagging, where the SVM is used to construct additional classifier models using an independent sample than the training sample (the out-of-bag sample) to enhance the generalization performance of the ensemble method. Then these additional predictors are combined with the CART to build the ensemble.

The new method is used to detect the defects in the insulation system in order to model a better diagnosis system for the electric power apparatus. The proposed method outperformed other ensemble methods such as bagging, adaboost.M1, logitboost and double bagging with LDA and k-NN ($k = 5$ and 10), in the experiments.

References

[1] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*, Belmont, CA:Wadsworth, 1984.

[2] L. Breiman, *Bagging predictors*, Machine Learning, 24(2):123–140,1996.

[3] L. Breiman, *Statistical modeling: the two cultures*, Statist. Sci., 16(3),199–231 (with discussion), 2001.

[4] C. J. C. Burges, *A Tutorial on Support Vector Machines for Pattern Recognition*, Data Min. Knowl. Discov., 2, 121–167 1998.

[5] C. Cortes and V. Vapnik, *Support-Vector Networks*, Mach. Learn., 20, 273–297 1995.

[6] Y. Freund and R. Schapire, *Experiments with a New boosting algorithm*, Machine Learning: Proceedings to the Thirteenth International Conference, Morgan Kaufmann, San Francisco, 148–156, 1996.

[7] J. Friedman, T. Hastie, and R. Tibshirani, *Additive logistic regression: a statistical view of boosting*, Annals of Statistics, 28, 337–407(with discussion), 2000.

[8] T. V. Gestel, J.Suykens, B.Baesens, S. Viaene,J. Vanthienen, G. Dedene, B. D. Moor, and J. Vandewalle, *Benchmarking least squares support vector machine classifiers* Machine Learning, 2001.

[9] H. Hirose, S. Matsuda and M. Hikita, *Electrical Insulation Diagnosing using a New Statistical Classification Method*, In the Proceedings of 8th Internal Conference on Properties and Applications of Dielectric Materials (ICPADM2006), 698–701, 2006.

[10] H. Hirose, M. Hikita, S. Ohtsuka, S. Tsuru and J. Ichimaru, *Diagnosing the Electric Power Apparatuses using the Decision Tree Method*, IEEE Trans., Dielectrics and Electrical Insulation, 15(5), 1252–1261, 2008.

[11] H. Hirose, F. Zaman, K. Tsuru, T. Tsuboi, and S. Okabe, *Diagnosis Accuracy in Electric Power Apparatuses Conditions using the Classification Methods*, IEICE Technical Report, to appear,2008.

[12] T. Hothorn and B. Lausen, *Double-bagging: combining classifiers by bootstrap aggregation*, Pattern Recognition,36 (6), 1303–1309, 2003.

[13] W. Iba and P. Langley, *Induction of one-level decision trees*, In Proceedings of Ninth International Machine Learning Conference, Aberdeen, Scotland. 1992.

[14] T. Joachims. Making large-scale support vector machine learning practical, Advances in Kernel Methods: Support Vector Machines, MIT Press, Cambridge, MA, 1999.

[15] H. C. Kim, S. Pang, H. M. Je, et al., *Constructing support vector machine ensemble*, Pattern Recognition, 36(12), 2757–2767, 2003.

[16] Y. Li, Y. Cal, R. Yin, X. Xu, *Fault diagnosis based on support vector machine ensemble*, In Proceedings of 2005 International Conference on Machine Learning and Cybernetics, 6, 3309–3314,18-21 Aug. 2005.

[17] K. Li, Y. Dai, W. Zhang, *Ensemble Implementations on Diversified Support Vector Machines*, In the Proceeding of IEEE Fourth International Conference on Natural Computation, 180-184, 2008.

[18] D. Meyer, F. Leisch, and K Hornik, *The support vector machine under test*, Neurocomputing, 55:169–186, September 2003.

[19] J. Platt, *Fast Training of Support Vector Machines Using Sequential Minimal Optimization* In Advances in Kernel Methods - Support Vector Learning, B. Scholkopf, C. J. C. Burges, and A. J. Smola, Eds., MIT Press, Cambridge, Massachusetts, 185–208, 1999.

[20] B. Scholkopf, A. Smola , and K. Muller, *Nonlinear component analysis as a kernel eigenvalue problem*, Neural Computation, 10(5),1299–1319,1998.

[21] B. Scholkopf, C. J. C. Burges, and A. J. Smola, *Advances in Kernel Methods: Support Vector Learning*, MIT Press, Cambridge, Massachusetts, 1999.

[22] B. Sun, D. Huang, *Least squares support vector machine ensemble*, In Proceedings of IEEE International Joint Conference on Neural Networks, 3, 2013–2016, 2004.

[23] V. Vapnik. The Nature of Statistical Learning Theory,Springer, New York, 1999.

[24] J. Weston, C. Watkins, *Support vector machines for multi-class pattern recognition*, Proceedings of the Seventh European Symposium on Artificial Neural Networks, Bruges, Belgium, 1999.