# On Improvement on Generalization Performance of Classifier by Using Empirical Risk

Yukun Chen[1,2], Hai Zhao[1] and Bao-Liang Lu[1]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Department of Engineering Mechanics, Shanghai Jiao Tong University
1954 Hua Shan Rd., Shanghai 200030, China
ykchen@sjtu.edu.cn, {zhaohai,blu}@cs.sjtu.edu.cn

***Abstract*— A combination classification algorithm, ER-SVM, is proposed to improve the generalization performance of support vector machine (SVM) by directly making full use of the Empirical Risk (ER) information of SVM in the paper. SVM classification is the implementation of Structure Risk Minimization (SRM) principle. SVM may achieve SRM from the minimal summation of ER and VC confidence according to the theory of VC dimension. However, the ER is seldom zero for a trained SVM in practice. That is, though the minimal summation of ER and VC confidence can be achieved in theory, it is very time-consuming in parameters selection for a given task to make ER zero. In order to overcome such difficulty, a combination classification algorithm is proposed to improve the performance by utilizing ER information. The SR arising from the existing ER is reduced by using aided nearest neighbor method. In addition, the proposed algorithm is independent of training parameters in SVM. The experimental results verify the effectiveness of the proposed algorithm.**

## I. INTRODUCTION

As well known, support vector machines (SVMs) based classification have been paid most attention by the researchers in the community of machine learning. It is a successful implementation of the SRM principle based on statistical learning theory [1]. SVM classification often brings out better generalization performance than other methods. However, there are still the following potential drawbacks in SVM. a) For a specified SVM, its ER is seldom zero; b) It is often hard to effectively determine a suitable kernel function and the corresponding optimal parameters. It is often difficult to accomplish a classification task well by using only one single classification method. What's more, there are many other classification methods except for SVM. Since the ER of SVM is seldom zero, it is natural for the idea to adopt other possible classification methods which may take advantage of the ER information, then we can combine them together with SVM to solve classification problem [2][3][4].

There have been some existing studies in discussing combination of Nearest Neighbor (NN) method and SVM. Li et al. suggest that the test set are divided into two parts. The samples in the nearer part from the optimal hyperplane are classified by NN classifier, while the samples in the rest part are classified by SVM [5][6]. The method partially avoids the difficulty in selecting the kernel function and its parameters and improve the generalization performance in the cases without ideal parameters. The algorithm proposed in [5][6] is further improved by Tian in [7]. The optimal distance from hyperplane between two parts of test set is given, and a lower generalization error is obtained. Karacali et al. work in a distinct way. They realize the SRM by a k-NN classifier. Such technique also partially overcomes some common difficulties in SVM [8]. Their work don't utilize the statistical learning theory and quadratic programming, therefore their method is not very close to the technologies used in SVM. In one word, all of the existing methods don't directly reduce ER in order to improve the generalization performance, and ER still exists to some extend.

SVM classifier may realize a global optimization based on quadratic programming. SVM is the implementation of the SRM principle instead of the ERM principle in traditional classification methods. The optimal hyperplane based on maximizing the margin between two classes can be determined in SVM. It is known that the SRM can be achieved by the minimal summation of the ER and VC confidence according to VC dimension theory. However, the ER is seldom zero when the SR is minimal. That is, the ER is often larger than zero in practice. Essentially, the property of SVM almost causes that it can't recognize all the training samples correctly even for a consistent and totally correct-labeled training samples set. If we acknowledge that every training sample always hold some information on data distribution, then this means that if the ER is not zero, then SVM can't achieve the best generalization performance that a classifier with all the information in training samples could achieve. The algorithm proposed in this paper will fully use ER loss of SVM to improve the generalization performance of SVM.

The remain part of this paper is organized as follow. The ERM principle and k-Nearest Neighbor method are briefly introduced in Section II. The proposed algorithm is detailed in section III. The experimental results and discussion are given in Section IV and V, respectively. The conclusion and the future research are given in Section VI.

## II. THE BACKGROUND

### A. The Empirical Risk Minimization

Since a two-class problem is essential and multi-class problem may be always decomposed into some two-class ones according to 'standard' decomposition method, such as one-

0-7803-9422-4/05/$20.00 ©2005 IEEE

verse-one and one-verse-rest strategies, this study will only concern with two-class classification problem.

Assumed that both the training samples and the test samples satisfy an unknown joint probability distribution, that is, they are iid (independent and identical distribution). The aim of learning is to determine an adjustable parameter $\alpha$ in a set of possible functions. SVM classifier is to find the parameter $\alpha$ to minimize the expected test error, that is, to realize the SRM.

An upper bound of the SR, named after VC dimension upper bound, is given by Vapnik and Chervonenkis [1],

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h(ln(2h/l)+1) - ln(\eta/4)}{l}} \qquad (1)$$

Where $R(\alpha)$ stands for the SR, $R_{emp}(\alpha)$ stands for the ER, $h$ is VC dimension and $l$ is the number of the training samples. The equation (1) will hold along with the probability $1-\eta$ for any $0 \leq \eta < 1$.

The second item on the right hand side in (1) is often called "VC confidence". One may notice that the smaller $R(\alpha)$ can not be obtained even for a very small $R_{emp}(\alpha)$ when $h/l$ is very larger. That is, SRM based SVM classifier can't be sure to get good generalization ability in such case. Thus, the summation of ER and VC confidence should be minimized by adjusting them two simultaneously for the minimization of the SR. Unfortunately, ER will get larger when $h/l$ is getting smaller. Only if $h$ is an optimal value as illustrated in Fig. 1, the best generalization performance can be obtained. SRM Inductive Principle may be adopted to find the optimal value of $h$ to minimize the summation of ER and VC confidence.

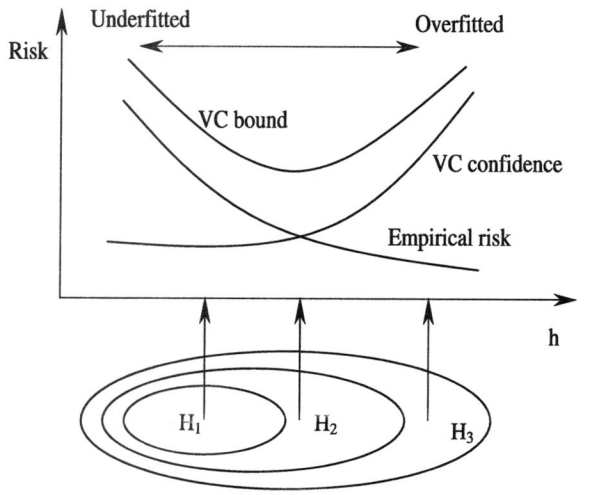


Fig. 1. Illustration of Structure Risk Minimization

We know that ER is seldom minimal, that is, ER is not zero when SR is minimal. ER can be expressed as

$$\begin{aligned} R_{emp}(\alpha) &= \frac{1}{2l}\sum_{i=1}^{l}|y_i - f(x_i,\alpha)| \qquad (2) \\ &= \frac{1}{2l}\sum_{i=1}^{e}|y_i - f(x_i,\alpha)| \\ &= \frac{1}{2l}\sum_{i=1}^{e}2 = \frac{e}{l} \end{aligned}$$

where $e$ is the number of the misclassified training samples.

According to the property of a continuous function, the ER of all possible samples in the neighbor domain of all misclassified samples can't be zero, either. Assume that the radius of neighbor domain is $\delta$ and $\delta$ is small enough, then every sample in it will not be classified correctly as shown in Fig. 2.

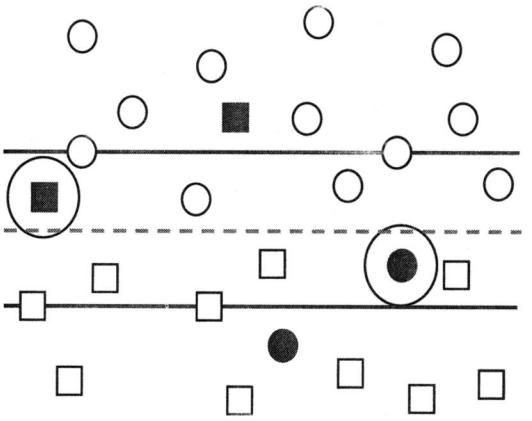

Fig. 2. A Trained SVM. The dashed line is the optimal hyperplane after training. All samples in the two solid lines consist the support vectors. The solid samples are the misclassified training samples because of ER. The big circle around the solid samples is the neighbor domain of a misclassified training sample, and the radius of the circle is $\delta$.

### *B. k-Nearest Neighbors Classifier*

The k-NN classifier is one of the most important methods in non-parameter pattern recognition. The main property of k-NN method is that all samples of the training data set are taken as "the representative point". To accomplish k-NN classification, the distances between the test sample $x$ and all the samples should be calculated and $k$ nearest neighbors of $x$ among them are chosen. Then the class label of $x$ is predicted by the most occurring label among $k$ nearest neighbors. 1NN method is the extreme case of k-NN method when $k = 1$.

## III. The Algorithm

The main idea of the proposed algorithm is to make full use of the ER information in training and adopt the assistant classification method, NN, to reduce SR in test. As seen in Fig. 2, the existence of misclassified training samples leads to the existence of ER and SR in the trained SVM. Even worse, any

sample falling in the small enough neighbor domain of any one misclassified training sample will still be misclassified in test by the trained SVM. The smaller $\delta$ is, the higher probability that such samples are misclassified is. This means that those test samples will be certain to be misclassified in test if they are falling in the specified neighbor domain of misclassified training samples. Therefore, if all the misclassified training samples are marked in training and used to train an aided classifier NN after training, then those to-be-misclassified test samples which are in the small enough neighbor domain of all the misclassified training samples will be correctly classified in test.

Notice once the training for the specified kernel function and its parameters is accomplished, an SVM will finally lost the distribution information of those misclassified training samples. The reason is that SVM with an improper kernel or corresponding parameters can't completely represent the probability distribution of the training set. If the data distribution information is fully considered, then the generalization ability of SVM can be ensured to some extent. The adopted assistant method to help fully making use of data distribution information in this study is just NN method. Something to be said is that, NN is not used directly in all the training and test samples. It is trained just by the misclassified training samples, and just tests the samples that are falling in the appointed radius neighbor domain of all the misclassified training samples.

It is self-evident that the more the information of the training samples is used, the higher generalization performance the designed classifier can achieve (Here, we only consider the case that both the training samples and the test samples are consistent and correctly labelled.). Though the SR can be minimized by SVM in theory, this depends strictly on the proper kernel function and corresponding parameters. If an unideal kernel function or its parameters are specified to train SVM, then SVM will not perform well as before. One may think it worth to search the best kernel and its parameters, however, such search is very time-consuming. The aided classifier, NN, can make up the drawback of SVM with an unideal kernel function or its parameters. Naturally, the corresponding SVM does not need to perform a time-consuming search for better kernel and parameters. Thus, we may obtain the tradeoff in the conflict between a time-consuming search of kernel and parameters and a lower generalization ability in SVM.

The proposed algorithm, named after ER-SVM, can be described in two phases as follows. The specified radius of the neighbor domain of a misclassified training sample is $\delta$.

*1) Training phase:* Train the SVM and self test the training samples with the trained SVM, then mark all the misclassified training samples.

*2) Test phase:* Firstly, the input test sample will be classified by NN classifier whose training set is consist of all the misclassified training samples by the trained SVM. Secondly, the distance between the nearest neighbor and the test sample, $d$, is calculated. If $d < \delta$, then the classification output is just the predication of the NN classifier, otherwise, the test sample will be classified by the trained SVM.

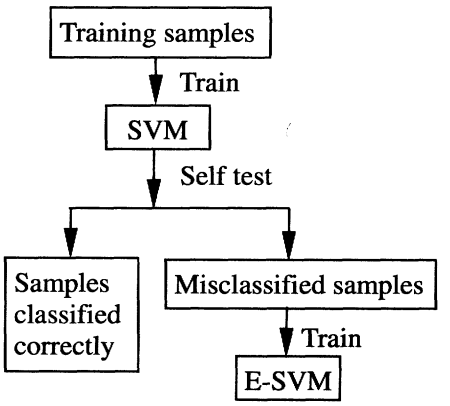


Fig. 3. Illustration of Training Phase of the Proposed Algorithm

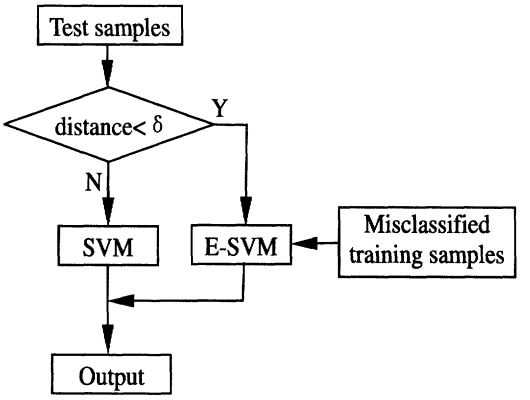


Fig. 4. Illustration of Test Phase of the Proposed Algorithm

The correcting rate is defined as $r = c/e$ to show the correcting capability of the aided NN classifier, where $c$ is the number of samples that is corrected by the NN classifier in test, and $e$ is the number of misclassified training samples by the trained SVM.

## IV. Experiment

One artificial data set generated by the function *normrnd* in MatLab 6.5 and three data sets from STATLOG benchmark repository [9] are chosen for this study. For UCI data set, each data set is consist of 100 groups of training sets and 100 groups of test sets. Each group of training set or test set holds the same number of samples. First several groups of training set or test set of each data set are chosen to combine into one single training set or test set for this study, respectively. The data information is shown in table I, where *Size* is the number of sample in each group training set or test set and *#Group* is the number of groups of training sets or test sets which are chosen in our experiments. The RBF kernel function is chosen in SVM.

TABLE I

DISTRIBUTIONS OF DATA SETS

| Data set | Training set | | Test set | | #Input |
|---|---|---|---|---|---|
| | Size | #Group | Size | #Group | |
| Artificial | 4000 | 1 | 4000 | 1 | 2 |
| Banana | 400 | 10 | 4900 | 5 | 2 |
| Heart | 170 | 2 | 100 | 10 | 13 |
| Waveform | 400 | 10 | 4600 | 2 | 21 |

TABLE II

THE PARAMETERS OF SVMS AND EXPERIMENTAL RESULTS

| Data set | SVM Parameters | | | Training Acc. | Test Acc. |
|---|---|---|---|---|---|
| | #Set | $C$ | $\gamma$ | (%) | (%) |
| Artificial | I | 64 | 0.125 | 87.98 | 87.83 |
| | II | 1024 | 4096 | 99.40 | 90.60 |
| Banana | I | 64 | 4 | 91.23 | 90.44 |
| | II | 2048 | 1024 | 99.83 | 92.38 |
| Heart | I | 4 | 0.25 | 95.59 | 90.40 |
| | II | 1 | 1 | 96.18 | 90.60 |
| Waveform | I | 1 | 0.0625 | 91.55 | 91.07 |
| | II | 16 | 0.125 | 94.43 | 92.61 |

TABLE III

EXPERIMENTAL RESULTS OF ARTIFICIAL DATA SET WITH PARAMETERS I

| $\delta$ | ER-SVM | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.000040 | 338 | 347 | 350 | 335 | -12 | 87.53 |
| 0.000030 | 328 | 263 | 264 | 327 | 64 | 89.43 |
| 0.000020 | 315 | 167 | 167 | 315 | 148 | 91.53 |
| 0.000010 | 303 | 52 | 52 | 303 | 251 | 94.10 |
| 0.000005 | 295 | 10 | 10 | 295 | 285 | 94.95 |
| 0.000001 | 292 | 0 | 0 | 292 | 292 | 95.13 |

TABLE IV

EXPERIMENTAL RESULTS OF ARTIFICIAL DATA SET WITH PARAMETERS II

| $\delta$ | ER-SVM | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.0000100 | 15 | 17 | 17 | 15 | -2 | 90.55 |
| 0.0000050 | 14 | 12 | 12 | 14 | 2 | 90.65 |
| 0.0000010 | 12 | 7 | 7 | 12 | 5 | 90.73 |
| 0.0000005 | 12 | 5 | 5 | 12 | 7 | 90.78 |
| 0.0000001 | 11 | 0 | 0 | 11 | 11 | 90.88 |

Two sets of parameters of SVM are chosen to demonstrate the effectiveness of the proposal algorithm in different cases for every data set. It should be noted these two sets of parameters are not optimal parameters for SVMs with RBF kernel but just randomly chosen. Because the aim of our experiment is to demonstrate that our algorithm utilize the existing ER to improve the performance of SVMs, we just randomly choose two sets of parameters for each data set. The parameters and experimental results are shown in Table II.

### A. Experimental results

A series of decreasing values of the radius, $\delta$, are selected to perform the classification experiments. We start with an experimental value of $\delta$, then reduce the value of $\delta$ continuously and finally stop the experiment in the data set until the aided NN classifier can correct all samples in the neighbor domain which will be misclassified by the original SVM.

All the test samples are divided into two parts. One part consists of the test samples inside the neighbor domain of every misclassified training sample. The other consists of the samples outside the neighbor domain. The samples in the first part is to be actually classified by the aided NN classifier according to our algorithm, the other part of samples are classified by the original SVM classifier. The correctly classified samples in the first part by the NN classifier is countered. As a comparison, the samples in the first part are also taken to be classify by the original trained SVM, and the correctly classified samples are also countered, too. The experimental results of every data sets are shown in Tables V though X, where *#Cor.* and *#Inc.* are the number of samples correctly classified and misclassified by the NN and SVM classifier in the first part, respectively, and *#Sub.Cor.* is the difference of the number of samples correctly classified by the NN and the original SVM classifier.

As the results demonstrate, the aided NN classifier does improve the performance of the classification system as expected when $\delta$ decreases. This means that the ER can be utilized effectively to improve the generalization performance. Someone may ask, if we continue to reduce the value of $\delta$ at this time, what will happen? The answer is less test samples will fall into the neighbor domain of the misclassified training sample and the NN classifier will have little chance to work. If $\delta = 0$, then the aided NN classifier will never have chance to work and the classification system becomes a pure SVM classifier again.

We also give a comparison between our method and KSVM algorithm proposed in [5]. The experimental results are shown in Table XI, where *Thr.* is the parameter that determines whether every test sample is classified by NN or by SVM in algorithm of Li Rong. It can be found that the performance of our algorithm is superior to KSVM in most cases.

### B. Analysis for Experimental Results

As a comparison with other related algorithm, the parameter II of banana is gotten by searching more carefully. We search $C$ or $\gamma$ from $2^{-8}$, $2^{-4}$, ... , to $2^{4}$ combinatorially. Finally, $C = 2048$ and $\gamma = 1024$ is the "best" parameter for SVM. From the table II, V and VI, we can find that, even if the SR of parameter II is better than that of parameter I by almost 2 percent, but the final performance of ER-SVM with parameter

TABLE V

EXPERIMENTAL RESULTS OF BANANA DATA SET WITH PARAMETERS I

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.0005000 | 2000 | 3804 | 3888 | 1916 | -1888 | 82.74 |
| 0.0001000 | 1426 | 1098 | 1123 | 1401 | 303 | 91.68 |
| 0.0000500 | 1331 | 573 | 583 | 1321 | 748 | 93.50 |
| 0.0000100 | 1211 | 98 | 103 | 1206 | 1108 | 94.97 |
| 0.0000050 | 1181 | 59 | 59 | 1181 | 1122 | 95.02 |
| 0.0000010 | 1166 | 5 | 5 | 1166 | 1161 | 95.18 |
| 0.0000005 | 1166 | 0 | 0 | 1166 | 1166 | 95.20 |

TABLE VI

EXPERIMENTAL RESULTS OF BANANA DATA SET WITH PARAMETERS II

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.000050 | 46 | 50 | 55 | -9 | -9 | 92.35 |
| 0.000010 | 31 | 18 | 18 | 13 | 13 | 92.44 |
| 0.000005 | 31 | 13 | 13 | 18 | 18 | 92.46 |
| 0.000001 | 31 | 0 | 0 | 31 | 31 | 92.51 |

TABLE VII

EXPERIMENTAL RESULTS OF HEART DATA SET WITH PARAMETERS I

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.40 | 47 | 63 | 63 | 47 | -16 | 88.80 |
| 0.30 | 47 | 44 | 44 | 47 | 3 | 90.70 |
| 0.20 | 47 | 12 | 12 | 47 | 35 | 93.90 |
| 0.10 | 41 | 5 | 5 | 41 | 36 | 94.00 |
| 0.05 | 41 | 0 | 0 | 41 | 41 | 94.50 |

TABLE VIII

EXPERIMENTAL RESULTS OF HEART DATA SET WITH PARAMETERS II

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.4 | 51 | 80 | 80 | 51 | -29 | 87.40 |
| 0.3 | 51 | 56 | 56 | 51 | -5 | 89.80 |
| 0.2 | 51 | 12 | 12 | 51 | 39 | 94.20 |
| 0.1 | 39 | 5 | 1 | 43 | 38 | 94.10 |
| 0.05 | 39 | 0 | 0 | 39 | 39 | 94.20 |

II (The accuracy is 92.51%.) is not better than than that with parameter I (The accuracy is 95.20%.). Why does this happen? We give the following reasons.

*1) The ER of parameter II is less than that of parameter I:* In fact, we can find that the ER of the parameter II is very little (Only 7 training samples are misclassified in our experiment.) while the ER of the parameter I is larger (350 training samples are misclassified.). Since there are less ER for parameter II to be used by ER-SVM, it have less chance to correct more test samples that fall into the neighbor domain of every misclassified training samples.

*2) SVM with parameter II is more complex than that of parameter I:* Though the declining SR makes SVM have better generalization performance, it also leads to SVM more complex ( The SVs with parameter I is 846, while the SVs with parameter II is 1458.), so the improvement of generalization performance is limited.

The explanation is also for other data sets. For artificial data set, although SVM has higher accuracy than SVM with parameter I in test, it is more complex than SVM with parameter I. For SVM with parameter II, its training accuracy is too high to have left any ER to be used by NN classifier, so the improvement of generalization performance is limited and the final test accuracy of SVM with parameter II (90.88%) is not higher than that (95.13%) of SVM with parameter I.

## V. DISCUSSION

The algorithm can improve the generalization of SVM by using an aided NN classifier with misclassified training samples being the training set. The proposed algorithm may give a distinct improvement on the generalization ability especially for the SVM that is with unideal kernel or parameters. Of course, there is a potential drawback in the NN method aided SVM classification. That is, if some selected ideal kernel functions with its proper parameters minimize the SR, and at this time the ER is also zero, then there will not be any capacity for the aided NN classifier and the proposed algorithm will be invalid at this time. However, such case seldom occurs according to VC dimension theory because SVM with the ER being zero and the minimal SR is often too complex to hold good generalization ability. If we consider that the better the kernel and its parameter are, the more time it would take to search, the proposed algorithm in this paper has given an elastic tradeoff in the conflict between search time of the kernel and its parameters and the generalization ability.

One of the remained problems for our algorithm is how to effectively determine the radius of neighbor domain for the NN classifier. If the radius is too small, then there will not be many samples to be corrected. If the radius is too large that the misclassified training samples neighbor domain will even comprise all the test samples, then the combining classifier will become a pure NN classifier trained by the misclassified training samples, but not the SVM classifier any more. The radius naturally has something with the distribution of sample. The future work can be taken to find an ideal radius to obtain the best generalization ability. Someone may argue that our method only replace the search procedure of the kernel and its parameters by the search procedure of the ideal radius of the neighbor domain. This is surely a partial truth at present. Notice a simple radius determination is much easier to realize than the determination of the kernel and its parameters. What's more, as is known that fast finding the best kernel and its

TABLE IX

EXPERIMENTAL RESULTS OF WAVEFORM DATA SET WITH PARAMETERS I

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.8 | 1309 | 1438 | 2055 | 692 | -746 | 82.96 |
| 0.7 | 984 | 866 | 1256 | 594 | -272 | 88.11 |
| 0.6 | 637 | 418 | 563 | 492 | 74 | 91.87 |
| 0.5 | 460 | 101 | 135 | 426 | 325 | 94.60 |
| 0.4 | 409 | 21 | 22 | 408 | 387 | 95.27 |
| 0.3 | 398 | 2 | 2 | 398 | 396 | 95.37 |
| 0.2 | 398 | 0 | 0 | 398 | 398 | 95.40 |

TABLE X

EXPERIMENTAL RESULTS OF WAVEFORM DATA SET WITH PARAMETERS II

| $\delta$ | NN | | SVM | | #Sub. | #Totoal Acc. |
|---|---|---|---|---|---|---|
| | #Cor. | #Inc. | #Cor. | #Inc. | Cor. | (%) |
| 0.8 | 1016 | 1453 | 1935 | 534 | -919 | 82.62 |
| 0.7 | 692 | 854 | 1092 | 454 | -400 | 88.26 |
| 0.6 | 433 | 376 | 449 | 360 | -16 | 92.43 |
| 0.5 | 315 | 86 | 97 | 304 | 218 | 94.98 |
| 0.4 | 285 | 15 | 16 | 284 | 269 | 95.53 |
| 0.3 | 282 | 2 | 2 | 282 | 280 | 95.65 |
| 0.2 | 282 | 0 | 0 | 282 | 282 | 95.67 |

proper parameters for SVM is an opening difficult problem to be solved. So our algorithm is still very meaningful.

In our algorithm, we just discuss the case that uses NN classifier as an aided classifier. In fact, the NN classifier is just an example to simply accomplish our algorithm, we can also use some other more valid classifiers on the misclassified training samples set. Finding a better aided classifier is also our future work. More effectively utilizing the ER is another important work. Finally, the most important is that, in our algorithm, we only use ER and aided classifier to improve the generalization performance of SVM, in fact, how to use the ER to direct selecting the best kernel and its proper parameter to change or to reduce the support vectors of SVM to improve generalization is even more challenging.

## VI. CONCLUSIONS

A combining classification algorithm is proposed to improve the performance of generalization in the paper by directly making full use of Empirical Risk information of SVM. In fact, the proposed algorithm utilized the distribution information of training samples that SVM can not fully absorb in training.The validity and correctness of the proposed algorithm have been verified by the experimental results.

## ACKNOWLEDGEMENTS

We give thanks to doctor Jing Li for her checking the paper carefully. This research was also partially supported by the National Natural Science Foundation of China via the grants NSFC 60375022 and NSFC 60473040.

TABLE XI

COMPARISON WITH LI RONG'S ALGORITHM AND ORIGINAL SVM

| Data set | The parameters | | | The accuracy (%) | | |
|---|---|---|---|---|---|---|
| | $C$ | $\gamma$ | Thr. | KSVM | SVM | ER-SVM |
| Artificial | 64 | 0.125 | 1 | 91.00 | 87.83 | 95.13 |
| | 1024 | 4096 | 1 | 92.70 | 90.60 | 90.88 |
| Banana | 64 | 4 | 1 | 92.79 | 90.44 | 95.20 |
| | 2048 | 1024 | 1 | 89.12 | 92.38 | 92.51 |
| | 2048 | 1024 | 0.6 | 92.04 | 92.38 | 92.51 |
| Heart | 4 | 0.25 | 1 | 94.30 | 90.40 | 94.50 |
| | 1 | 1 | 1 | 94.70 | 90.60 | 94.20 |
| Waveform | 1 | 0.0625 | 1 | 93.08 | 91.07 | 95.40 |
| | 16 | 0.125 | 1 | 93.42 | 92.61 | 95.67 |

## REFERENCES

[1] V. Vapnik, *" The Nature of Statistical Learning Theory"*, Springer-Verlag, 1995.

[2] Yanhuang Jiang, "Research on stacked classifiers", A Chinese Periodical of *Computer Engineering and Science*, Vol. 26, No. 16 pp.67-70, 2004.

[3] Jianjun Zheng, "A Selective Approach for an Ensemble of Simple Bayesian Classifiers", A Chinese periodical of *Transactions of Beijing Institute of Technology*, Vol. 23, No. 6, pp.724-727, Dec., 2003.

[4] Hongbo Shi, "Boosting Based TAN Combination Classifier", A Chinese periodical of *Journal of Computer Researcn and Development*, Vol. 141, No. 12, pp.340-345, Feb, 2004.

[5] Rong Li, Shiwei Ye, and Zhongzhi Shi, "SVM-kNN Classifier: A New Method of Improving the Accuracy of SVM Classifier", A Chinese periodical of *Acta Electronica Sinica*, Vol. 30, No. 5, pp.745-748, May, 2002.

[6] Rong Li, Shiwei Ye, and Zhongzhi Shi, "An Effective Classified Algorithm of support Vector Machine with Multi-Representative Points Based on Nearest Neighbor Principle", *International Conferences on Info-tech and Info-net (ICII)*, pp.113-119, 2004.

[7] Ming Tian, Yi Zhuang, and Songcan Chen, "Improving Support Vector Machine Classifier by Combining it with k Nearest Neighbor Principle Based on the Best Distance Measurement", *2003 IEEE Intelligent Transportation Systems Proceedings*, Vol. 1: 373-378, 2003.

[8] Bilge Karacali and Hamid Krim, "Fast Minimization of Structural Risk by Nearest Neighbor rule", *IEEE Transaction on Neural Networks*, Vol. 14, No. 1, pp.127-137, Jan., 2002.

[9] G. Ratsch, T. Onoda and K. Muller, "Soft margins for AdaBoost", *Machine Learning*, pp.1-35, [http://ida.first.fhg.de/projects/bench/], 2000.