# Improvement of Performance in Multiclass Problems by Using Biclassification Based on Error-Correcting Output Code

Young Bun Kim, Jung Hun Oh and Jean Gao *

*Abstract*—**Error-correcting output coding (ECOC) is a widely used multicategory classification algorithm that decomposes multiclass problems into a set of binary classification problems. In this paper, we propose a new method based on a bi-classification strategy, consisting of one-vs-one and ECOC classification. Also we introduce methods to improve a standard ECOC. The proposed method is compared to other algorithms by performing experiments with gene expression datasets.**

***Keywords: ECOC, multiclass, bi-classification, one-vs-one***

## 1 Introduction

In data communication, error bits caused by noise during transmission from transmitter to receiver can be detected and corrected by using error-correcting code. In multiclass problems, a similar idea was borrowed from the error-correcting concept. That is called error-correcting output coding (ECOC) where misclassifications wrongly guessed by several classifiers can be corrected. ECOC is a classification method that breaks down $k$-class problems into two-class problems. In ECOC framework, each class is assigned a codeword that is a unique string of length $n$ made by {-1,1}, entirely forming a $k \times n$ coding matrix $M$. In each column of the matrix $M$, training samples of class labels that have the same value are combined, eventually forming a binary class label. Overall, it yields the $n$ number of classifiers called dichotomies. After training with matrix $M$, a test sample is put into the trained classifiers, which produces the $n$ number of code bits consisting of {-1,1}. The test sample is assigned to a class with the closet codeword to the generated code using some distance measure, decoding function. To make a good matrix in ECOC is a challenging issue where rows as well as columns should be well separated one another.

There are two main strategies to tackle the multiclass problems [1], [2], [3]. The first method at once predicts a class label for a test sample by using all class samples. In the second method, the multiclass problems are broken down into a set of binary classification problems which is more computationally tractable [4]. Those methods include one-against-the-rest, one-against-one, and error-correcting output coding (ECOC). In this study, we employ one-vs-one and ECOC method. That is, the two methods form our proposed bi-classification while being performed independently.

Many methods to design codes have been studied [5], [6], [7], [8]. Dietterich and Bakiri provided methods for constructing good error-correcting codes in which different coding methods were employed according to the number of classes included in the problem [5]. Also they showed the robustness of the ECOC in several attributes such as the small sample size and the assignment of codewords. Pujol *et al.* introduced the discriminant ECOC which dealt successfully with the problem of the design of application dependent discrete ECOC matrices [7]. Ie *et al.* proposed a multicategory classification method based on ECOC for a classification problem to assign a sequence of amino acids to one of the known protein structures [3]. Decoding rule is also a very important component in ECOC [9], [10]. The decoding rule presented by Passerini *et al.* combines the margins through an estimate of their class conditional probabilities, which recalibrates the outputs of the classifiers and improves the overall multiclass classification accuracy [10]. Escalera *et al.* introduced a variant of ECOC, called ECOC-ONE, which generates a matrix with an initial optimal tree, forming a network by using dichotomies as nodes [11]. Kuncheva proposed to use diversity measures rather than the standard minimum Hamming distance to evaluate the quality of an error-correcting code and suggested an evolutionary algorithm to construct the code [12]. Methods combining boosting and the ECOC have been studied, which have the performance advantages of boosting [13], [14].

In this paper, we propose a new multicategory classification strategy based on ECOC in which a bi-classification method is used. The bi-classification consists of one-vs-one and ECOC classification. For test samples, two class labels generated from two classifications are compared. If the two class labels are different, a retraining is performed with only the two-class samples. In order to vali-

*Y.B. Kim, J.H. Oh and J. Gao are with the Computer Science and Engineering Department, University of Texas, Arlington, Texas 76019, USA. (Emails:{ybkim, jung.oh, gao}@uta.edu).

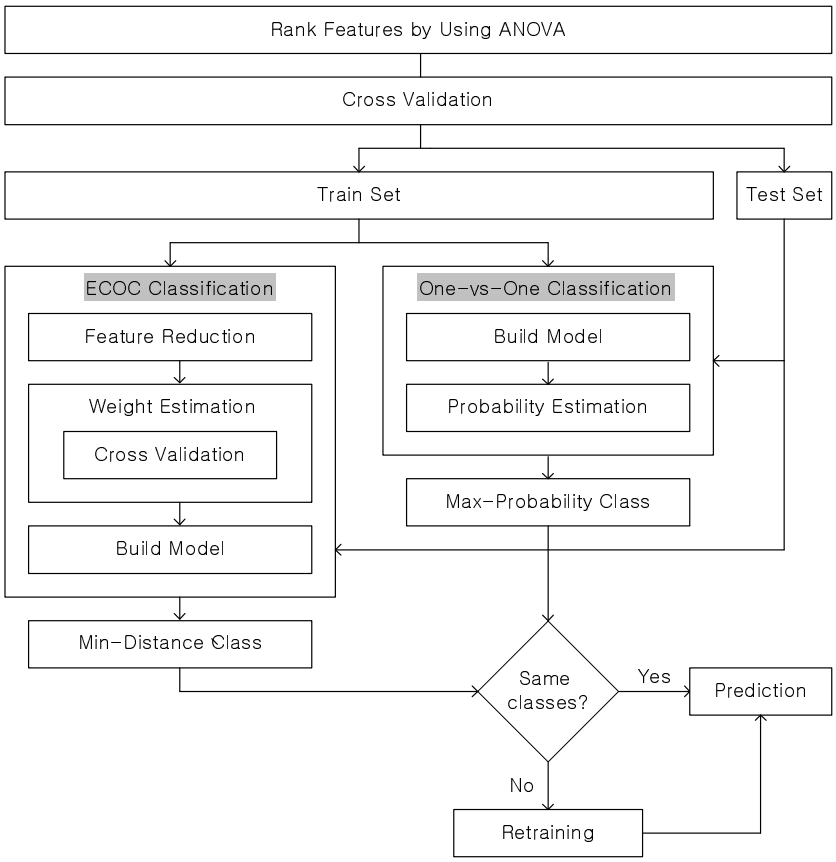



Figure 1: Algorithm for the proposed method.

date our proposed method, we perform experiments with well-known gene expression datasets such as ALL-AML-3 and breast cancer and compare the performance with other classification algorithms.

The rest of this paper is organized as follows. First, we describe our bi-classification strategy where several issues such as weighting, feature reduction, SVM (Support Vector Machine) probability and decoding function are dealt with in detail. Then, we show experimental results of our method which are compared with other algorithms and conclude the paper.

## 2 Bi-classification strategy

We propose a strategy to enhance a standard ECOC. The method we use is in basis of a bi-classification, one-vs-one classification and ECOC classification. Two classifications are performed independently yielding two prediction results (class labels).

### 2.1 One-vs-one classification

The most commonly used methods to decompose multi-class problems into a set of binary classification problems are one-vs-the-rest and one-vs-one. Either one results in as many class labels as the number of used binary classifiers. In general, voting is used to determine a final class label. In voting, two class labels produced in each binary classifier have an equal weight. Instead, if there is a method to represent the two class labels by a certain measure, it would be better. Platt proposed a way to obtain a posterior probability by using a parametric sigmoid model based on SVM [15]. Now, a remaining issue is how to combine all probabilistic values (each binary classifier yields two.) so that each class label has a probability. For this, many studies in one-vs-one strategy have been done. In this study, we employ the one-vs-one as the binary classifier. To decide a final probability of a class $c$, we use the following simple equation:

$$P_c = \frac{1}{k-1} \sum P(y = c|x). \quad (1)$$

A test sample is assigned to a class which has the maximum probability.

$$\omega = \mathbf{max}_{1 \le c \le k} P_c \quad (2)$$

### 2.2 ECOC classification

Weight values of all dichotomies in the standard ECOC are equal. However, prediction ability of dichotomies is different depending on matrix $M$. To calculate the weight value for dichotomies, we employ a weighting function

Table 1: The mean and standard deviation (in parenthesis) of accuracies in ALL-AML-3 dataset.

| Methods / No. of features | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|
| Proposed Method | **97.92(0.73)** | 96.39(0.97) | 97.08(0.79) | 96.11(0.59) | 96.11(0.59) |
| ECOC | 95.56(1.28) | 94.17(2.25) | 95.42(1.86) | 94.86(1.97) | 93.06(2.27) |
| Random Forest | 95.83(1.73) | 95.14(1.50) | 95.28(2.09) | 94.72(1.83) | 94.17(2.34) |
| Naive Bayes | 96.25(1.32) | 97.64(0.94) | 96.81(0.67) | 97.22(0.65) | 97.22(0.65) |
| J48 | 88.47(1.97) | 89.86(3.34) | 90.42(1.66) | 89.72(2.47) | 93.47(1.61) |

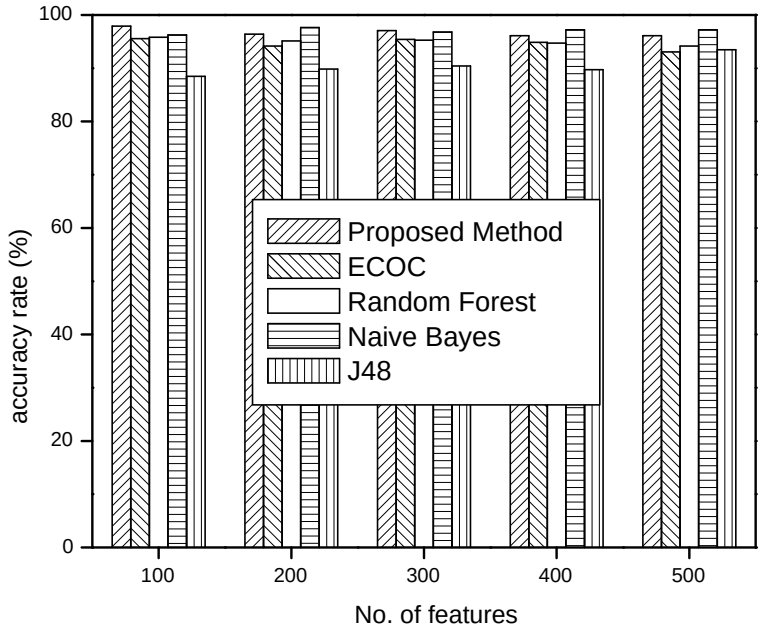



Figure 2: Histograms of classification accuracies on ALL-AML-3 dataset.

which is similar to that used in boosting algorithms as shown in Eq. (3). The weight value of each dichotomy is computed by using the error rate generated by the dichotomy with validation dataset. Each weight value represents how well the corresponding dichotomy predicts a code bit. We use linear SVMs in weighting function and dichotomies. In Eq. (3), $w_i$ and $e_i$ are the weight value and the error rate of the $i$-th dichotomy which are obtained with the validation dataset. The concept behind the weighting function is that if the accuracy of each dichotomy is larger than 50%, the weight value becomes positive; otherwise a negative value is returned, which comes to a penalty.

$$w_i = 0.5\mathbf{log}(\frac{1-e_i}{e_i}) \tag{3}$$

There may exist irrelevant features in training with dichotomies. It will degrade prediction ability of dichotomies. Here, we use a feature reduction algorithm which reduces computational cost caused by the $n$ number of dichotomies. We use the information gain as a feature reduction algorithm where features whose gain values are less than 0 are eliminated.

We use a decoding function to see which codeword in matrix $M$ is closest to an output code generated by dichotomies. The weight value above is used in the following decoding function

$$d_j = \sum_{i=1}^{n} \mathbf{exp}(-w_i x_i y_i^j) \tag{4}$$

where $d_j$ is a distance in the $j$-th class, $x_i$ is the $i$-th bit of the output code generated with a test sample and $y_i^j$ is the $i$-th bit of the codeword of the $j$-th class in $M$ matrix. A test sample is assigned to a class which has the minimum distance.

$$\varphi = \mathbf{min}_{1\leq j\leq k} d_j \tag{5}$$

### 2.3 Retraining

After independent running of one-vs-one and ECOC classification, we compare the two results to predict a final class label for a test sample. If the two predictions are the same, the identical label will be selected for the test sample; otherwise, a retraining is carried out because we can not predict a final class label between two different ones. That is, the retraining is performed, if $\omega$ is different from $\varphi$. Since only two-class samples predicted

Table 2: The mean and standard deviation (in parenthesis) of accuracies in breast cancer dataset.

| Methods / No. of features | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|
| Proposed Method | 72.77(2.47) | **74.36(2.43)** | 72.98(4.29) | 71.81(4.11) | 67.34(2.70) |
| ECOC | 65.86(3.11) | 65.11(3.16) | 61.49(4.75) | 58.83(4.17) | 54.26(3.44) |
| Random Forest | 66.38(3.45) | 63.40(3.14) | 63.30(2.80) | 60.53(3.87) | 58.40(3.38) |
| Naive Bayes | 62.98(0.84) | 64.57(0.88) | 65.11(1.10) | 64.89(1.33) | 63.94(0.78) |
| J48 | 57.87(3.79) | 58.19(3.85) | 55.96(2.85) | 54.68(2.20) | 54.04(3.12) |

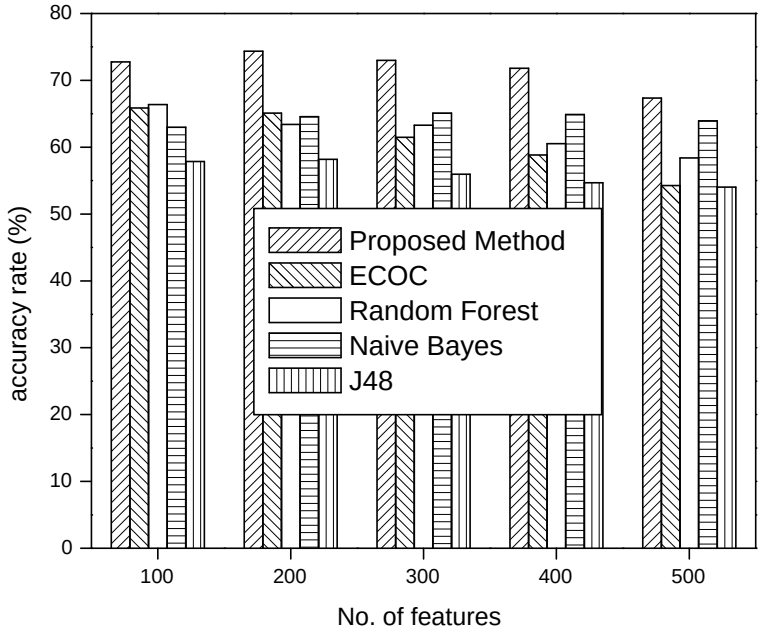



Figure 3: Histograms of classification accuracies on breast cancer dataset.

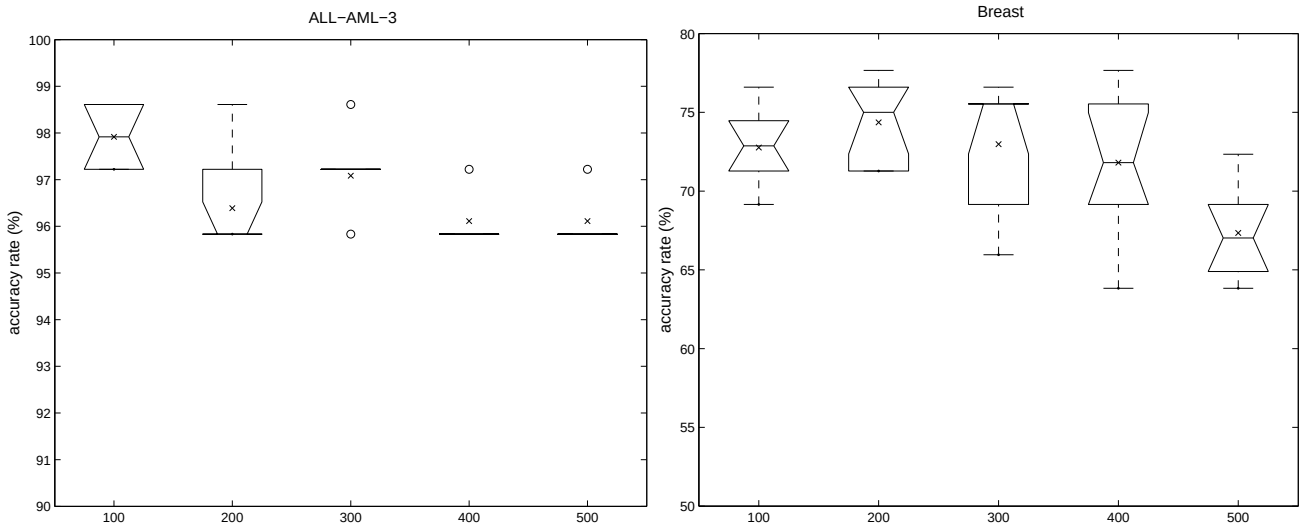
from one-vs-one and ECOC classification participate in the retraining, it will form a binary classification. Fig. 1 illustrates the proposed algorithm.

## 3 Experiments

To test the proposed method, we made use of gene expression data sets: ALL-AML-3 and breast cancer, both having 3 classes [16], [17]. ALL-AML-3 dataset consists of 72 samples and 7129 genes. Golub *et al.* studied this dataset in a binary classification problem between AML and ALL [18]. It is possible to separate them into a three class dataset such as B-cell, T-cell, and AML because of the bipartition of each component. The dataset is available at [http://www-genome.wi.mit.edu]. Breast cancer dataset is composed of 96 samples and 4869 genes [19]. The original breast cancer dataset can be downloaded from [http:// www.rii.com/publications/2002/vantveer.htm]. The dataset has 34 patients that developed distant metastases within 5 years, 44 that remained disease-free for over 5 years, and 18 with BRCA1 germline mutations. Two samples corresponding to BRCA2 mutations were excluded in this study.

We implemented the proposed algorithm based on LIBSVM [20]. Linear SVM was used in retraining and both one-vs-one and ECOC classification. For ECOC classification, random coding strategy was used in which values {-1, 1} are selected uniformly at random to make matrix $M$. SVM is a kernel based learning algorithm to solve two-class classification problems [21], [22], [23], [24], [25]. An optimal hyperplane is sought to separate a given set of binary labeled training data by maximizing the margin between the two classes. To do so, SVM maps the training data into a higher dimensional space via a mapping function and constructs a decision function.

Prior to experiments, normalization was carried out so that each gene expression has mean equal to 0 and variance equal to 1. With ANOVA (analysis of variance) statistical method, we ranked the genes (features) and performed the experiments with the top 100, then the top 200 and so forth up to the top 500 features. Our method further removed the features in dichotomies using the information gain method. The performance of our method is compared with other algorithms, standard ECOC, Random Forest, Naive Bayes, and J48. All comparison algorithms were experimented in WEKA tools [26].




(a) ALL-AML-3 (b) Breast

Figure 4: Boxplots of classification accuracies of the proposed method on (a) ALL-AML-3 and (b) breast cancer dataset.

In all experiments, 10-CV (Cross Validation) was applied where at each CV, 90 percent of samples are used for training and the rest for testing. In order to obtain the weight value of each dichotomy, the 90 percent samples for training were further split into 10 folds. Again, with the 10 folds, 10-CV was carried out 20 times and an averaged error was put into Eq. (3) to obtain the weight value. This task was separately performed in each dichotomy. The whole procedure was iterated 30 times.

Table 1 and Fig. 2 represent the experimental results on ALL-AML-3 dataset. The proposed method achieved the best accuracy 97.92% with 100 features. Overall, the results of Naive Bayes are comparable to our method. On breast cancer dataset, the results of the proposed method outperformed others for all cases as shown in Table 2 and Fig. 3. Again, our method obtained the best accuracy 74.36% with 200 features. Díaz-Uriarte and Alvarez de Andrés achieved the accuracy of 65.4% using the 0.632+ bootstrap method with 200 bootstrap samples [19]. In both experiments, the performance of J48 is worst. The standard ECOC also does not show a good performance. As a result, we are motivated to make a variant of the standard ECOC. Fig. 4-(a) and Fig. 4-(b) illustrate boxplots of the proposed method on ALL-AML-3 and breast cancer dataset, respectively. For ALL-AML-3 dataset as shown in Fig. 4-(a), our method shows a very stable performance.

## 4 Conclusion

We proposed a new bi-classification strategy based on ECOC, where both results of one-vs-one and ECOC classification are considered. To reduce cost caused by using all features in dichotomies, ECOC classification used a feature reduction algorithm. Each dichotomy was given its own weight value. Also, a new decoding function was presented. Through experiments, we showed that our method performed better than other algorithms. Also, we showed the standard ECOC does not provide a good performance against other methods. It motivated us to extend the standard ECOC. The bi-classification method can be a way to enhance the standard ECOC in multi-class problems.

We have designed a feature selection method in multi-class problems [27]. In future work, we will apply the method to our proposed bi-classification strategy to find an optimal feature subset. It will help us find biomarkers which are directly associated with diseases.

## Acknowledgments

This work was supported in part by NSF under grants IIS-0612152 and IIS-0612214.

## References

[1] C.W. Hsu and C.J. Lin, "A Comparison of Methods for Multi-class Support Vector Machines," *IEEE Trans. Neural Networks*, vol. 13, pp. 415-425, 2002.

[2] B. Fei and J. Liu, "Binary Tree of SVM: A New Fast Multiclass Training and Classification Algorithm," *IEEE Trans. Neural Networks*, vol. 17, no. 3, 2006.

[3] E. Ie, J. Weston, W.S. Noble, and C. Leslie, "Multi-class Protein Fold Recognition Using Adaptive Codes," *ICML 2005*, pp. 329-336, 2005.

[4] E.L. Allwein, R.E. Schapire, and Y. Singer, "Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers," *Journal of Machine Learning Research*, vol. 1, pp. 113-141, 2002.

[5] T.G. Dietterich and G. Bakiri, "Solving Multiclass Learning Problems via Error-Correcting Output Codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263-286, 1995.

[6] K. Crammer and Y. Singer, "On the Learnability and Design of Output Codes for Multiclass Problems," *Machine Learning*, vol. 47, no. 2-3, pp. 201-233, 2002.

[7] O. Pujol, P. Radeva, and J. Vitria, "Discriminant ECOC: A Heuristic Method for Application Dependent Design of Error Correcting Output Codes," *IEEE Trans. pattern analysis and machine intelligence*, vol. 28, no. 6, 2006.

[8] T. Windeatt and R. Ghaderi, "Coding and Decoding for Multiclass Learning Problems," *Information Fusion*, vol. 4, no. 1, pp. 11-21, 2003.

[9] R.S. Smith and T. Windeatt, "Decoding Rules for Error Correcting Output Code Ensembles," *MCS 2005*, LNCS 3511, pp. 53-63, 2005.

[10] A. Passerini, M. Pontil, and P. Frasconi, "New Results on Error Correcting Output Codes of Kernel Machines," *IEEE Trans. Neural Networks*, vol. 15, no. 1, 2004.

[11] S. Escalera and O. Pujol, "ECOC-ONE: A Novel Coding and Decoding Strategy," *ICPR 2006*, pp. 578-581, 2006.

[12] L.I. Kuncheva and C.J. Whitaker, "Measures of Diversity in Classifier Ensembles," *Mach. Learn.*, vol. 51, pp. 181-207, 2003.

[13] V. Guruswami and A. Sahai, "Multiclass Learning, Boosting, and Error-Correcting Codes," *COLT. 99*, pp. 145-155, 1999.

[14] R.E. Schapire, "Using Output Codes to Boost Multiclass Learning Problems," *Proc. 14th Intl. Conf. on Machine Learning*, pp. 313-321, 1997.

[15] J. Platt, "Probabilistic Outputs for SVMs and Comparisons to Regularized Likelihood Methods," *Advances in Large Margin Classifiers*. MIT Press, 1999.

[16] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene Selection for Cancer Classification Using Support Vector Machines," *Machine Learning*, vol. 46, pp. 389-422, 2002.

[17] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, and D. Haussler, "Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data," *Bioinformatics*, vol. 16, pp. 906-914, 2000.

[18] T.R. Golub, D.K. Slonim, P. Tamayo, *et al.*, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, vol. 286, pp. 531-537, 1999.

[19] R. Díaz-Uriarte and S. Alvarez de Andrés, "Gene Selection and Classification of Microarray Data Using Random Forest," *BMC Bioinformatics*, vol. 7, No. 3, 2006.

[20] C.-C. Chang and C.-J. Lin, LIBSVM: a library for support vector machines, Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm

[21] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[22] C.J.C Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, 1998.

[23] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing Kernel Parameters for Support Vector Machines", *Machine Learning*, vol. 46, no. 1-3, pp. 131-159, 2002.

[24] B. Schoelkopf and A.J. Smola, *Learning With Kernels*. Cambridge, MA: MIT Press, 2002.

[25] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[26] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques, 2nd ed*. San Francisco: Morgan Kaufmann, 2005.

[27] J.H. Oh, A. Nandi, P. Gurnani, P. Bryant-Greenwood, K.P. Rosenblatt, and J. Gao, "Prediction of Labor for Pregnant Women Using High-Resolution Mass Spectrometry Data," *BIBE 2006*, pp. 332-339, 2006.