A Hybrid Random Subspace Classifier Fusion Approach for Protein Mass Spectra Classification

Amin Assareh[1], Mohammad Hassan Moradi[1], and L. Gwenn Volkert[2]

[1] Department of Biomedical Engineering, Amirkabir University of Technology, Tehran, Iran
asserah83@googlemail.com, mhmoradi@aut.ac.ir
[2] Department of Computer Science, Kent State University, USA
volkert@cs.kent.edu

Abstract. Classifier fusion strategies have shown great potential to enhance the performance of pattern recognition systems. There is an agreement among researchers in classifier combination that the major factor for producing better accuracy is the diversity in the classifier team. Re-sampling based approaches like bagging, boosting and random subspace generate multiple models by training a single learning algorithm on multiple random replicates or sub-samples, in either feature space or the sample domain. In the present study we proposed a hybrid random subspace fusion scheme that simultaneously utilizes both the feature space and the sample domain to improve the diversity of the classifier ensemble. Experimental results using two protein mass spectra datasets of ovarian cancer demonstrate the usefulness of this approach for six learning algorithms (LDA, 1-NN, Decision Tree, Logistic Regression, Linear SVMs and MLP). The results also show that the proposed strategy outperforms three conventional re-sampling based ensemble algorithms on these datasets.

1 Introduction

Rapid advances in mass spectrometry have led to its use as a prime tool for diagnosis and biomarker discovery [1]. The high-dimensionality-small-sample (HDSS) problem of cancer proteomic datasets is the main issue that plagues and propels current research on protein mass spectra classification [2].

The complexity and subtlety of mass spectra patterns between cancer and normal samples may increase the chances of misclassification when a single classifier is used because a single classifier tends to cover patterns originating from only part of the sample space. Therefore, it would be beneficial if multiple classifiers could be trained in such a way that each of the classifiers covers a different part of the sample space and their classification results were integrated to produce the final classification. Resampling based algorithms such as bagging, boosting, or random forests improve the classification performance by associating multiple base classifiers to work as a "committee" or "ensemble" for decision-making. Any supervised learning algorithm can be used as a base classifier. Ensemble algorithms have been shown to not only increase classification accuracy, but also reduce the chances of overtraining since the committee avoids a biased decision by integrating the different predictions from the individual base classifiers [3]. In recent years a variety of approaches to classifier combination have been applied in the domain of protein mass spectra classification [3-8].

E. Marchiori and J.H. Moore (Eds.): EvoBIO 2008, LNCS 4973, pp. 1–11, 2008.

2 Background

Efforts to improve the performance of classifier combination strategies continue to be an active area of research, especially within the field of bioinformatics as the number of available datasets continues to rapidly increase. It has been empirically shown that the decision made by a set (pool/committee/ensemble/team) of classifiers is generally more accurate than any of the individual classifiers. Both theoretical and empirical research has demonstrated that a good team is one where the individual classifiers in the team are both accurate and make their errors on different parts of the input space. In the other words, one of major factors responsible for improving the performance of a classifier combination strategy is the diversity in the classifier team. There is a consensus among researchers in classifier combination that this diversity issue supersedes the importance of the aggregation method [9]. However, the choice of an appropriate aggregation method can further improve the performance of an ensemble of diverse classifiers.

From the architecture prospective, various schemes for combining multiple classifiers can be grouped into three main categories: 1) parallel, 2) cascading (or serial combination), and 3) hierarchical (tree-like). In the parallel architecture, all the individual classifiers are invoked independently, and their results are then combined by a suitable strategy. Most combination schemes in the literature belong to this category. In the gated parallel variant, the outputs of individual classifiers arc selected or weighted by a gating device before they are combined. In the cascading architecture, individual classifiers are invoked in a linear sequence. The number of possible classes for a given pattern is gradually reduced as more classifiers in the sequence have been invoked. For the sake of efficiency, inaccurate but cheap classifiers (low computational and measurement demands) are considered first, followed by more accurate and expensive classifiers. In the hierarchical architecture, individual classifiers are combined into a structure, which is similar to that of a decision tree classifier. The tree nodes, however, may now be associated with complex classifiers demanding a large number of features. The advantage of this architecture is the high efficiency and flexibility in exploiting the discriminant power of different types of features. Using these three basic architectures, even more complicated classifier combination systems can be constructed [9].

Different combiners expect different types of output from individual classifiers. Lei Xu et al. [10] grouped these expectations into three levels: 1) measurement (or confidence), 2) rank, and 3) abstract. At the confidence level, a classifier outputs a numerical value for each class indicating the belief or probability that the given input pattern belongs to that class. At the rank level, a classifier assigns a rank to each class with the highest rank being the first choice. Rank value cannot be used in isolation because the highest rank does not necessarily mean a high confidence in the classification. At the abstract level, a classifier only outputs a unique class label or several class labels (in which case, the classes are equally good). The confidence level conveys the richest information, while the abstract level contains the least amount of information about the decision being made.

Roughly speaking, building an ensemble based classifier system includes selecting an ensemble of individual classification algorithms, and choosing a decision function for combining the classifier outputs. Therefore, the design of an ensemble classifier system involves two main phases: the design of the classifier ensemble itself and the design of the combination function. Although this formulation of the design problem leads one to think that effective design should address both phases, until recently most design methods described in the literature have only focused on one phase [11].

2.1 Classifier Ensemble Design

So far, two main strategies are discussed in the literature on classifier combination: classifier *selection* and classifier *fusion*. The presumption in classifier selection is that each classifier has expertise in some local area of the feature space. When a feature vector *x* is submitted for classification, the classifier responsible for the vicinity of *x* is given the highest authority to label *x*. Classifier fusion, on the other hand, assumes that all classifiers are equally "experienced" in the whole feature space and the decisions of all of them are taken into account for any *x*.

Classifier fusion approaches are further divided into resampling-based methods and heterogenous methods. The resampling methods generate multiple models by training a single learning algorithm on multiple random replicates or sub-samples of a given dataset whereas the heterogeneous ensemble methods (also called multi-strategy methods) train several different learning algorithms on the same dataset. The approach we describe in this paper is clearly a resampling-based method but differs from the standard resampling-based methods of bagging, boosting, and random forest. In general, resampling-based methods take two perspectives: training a learning algorithm utilizing the same subset of features but different subsets of training data (i.e. Bagging [12] or Boosting [13, 14] or alternatively utilizing the same subset of training data but different subsets of the feature set (i.e. Random Forest or Random Subspace algorithms [15, 16]. Our hybrid approach combines these two perspectives by randomly selecting different subsets of training data *and* randomly selecting different features from a feature set.

2.2 Decision Function Design

In this work we investigate four decision functions to allow evaluation of the impact of different functions on our hybrid approach. The decision functions we investigate are the Majority function, the Weighted Majority function, the Mean function and the Decision Template approach. The 2001 paper by Kuncheva et al. [17] provides an excellent reference on the use of Decision Templates for multiple classifier fusion, including a detailed description of the construction of a soft decision profile for use in ensemble systems.

3 Methods

We have applied our approach to two serum protein mass spectra datasets of ovarian cancer, publicly available from the clinical proteomics program of the national cancer institute website (http://home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp). The first dataset is "Ovarian 8-7-02" which was produced using the WCX2 protein chip. An upgraded PBSII SELDI-TOF mass spectrometer was employed to generate the spectra, which includes 91 controls and 162 ovarian cancer samples. The second dataset is “Ovarian 4-3-02” prepared by the same chip, but the samples were processed by hand and the baseline was subtracted resulting in the negative intensities seen for some values. The spectra contain 100 control, 100 ovarian cancer and 16 benign samples. Each spectrum of these two datasets includes peak amplitude measurements at 15,154 points defined by corresponding *m/z* values in the range 0–20,000 Da. Figure 1 illustrates the mean spectrums of each dataset.

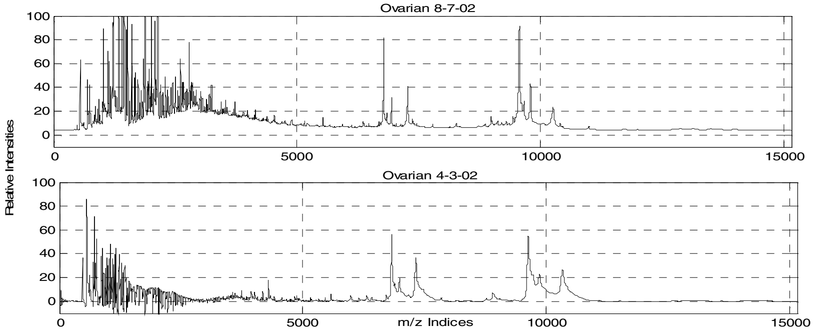

Fig. 1. The mean spectra of the applied datasets: Ovarian 8-7-02 (upper panel) and Ovarian 4-3-02 (lower panel)

Generally, a mass spectrum consists of signals, baseline, and noise. The signals are produced by the peptides, proteins, and contaminants present in the sample; the baseline is the slowly varying trend under the spectrum; and the noise consists of chemical background, electronic noise, signal intensity fluctuations, statistical noise, warping of the signal shapes (due to overcharging in ion traps), and statistical noise in the isotopic clusters (see below). Signals, baseline, and noise can never be totally separated; the baseline, for example, can depend on the presence of large and intense signals as well as on abundant low-intensity noise. Noise can be quite intense and is sometimes impossible to distinguish from real signals. [1]. The goal of preprocessing stage is to ‘‘clean up’’ the data such that machine learning algorithms will be able to extract key information and correctly classify new samples based on a limited set of examples [2]. In analyzing mass spectra of blood samples, the preprocessing stage roughly includes three main tasks: baseline correction, smoothing and normalization.

Mass spectra exhibit a monotonically decreasing baseline, which can be regarded as low frequency noise because the baseline lies over a fairly long mass-to-charge ratio range. In this study, we utilized local average within a moving window as a local estimator of the baseline and the overall baseline is estimated by sliding the window over the mass spectrum. The size of the applied window was 200 M/Z. In addition shape preserving piecewise cubic interpolation has been applied to regress the window estimated points to a soft curve. Mass spectra of blood samples also exhibit an additive high frequency noise component. The presence of this noise influences both data mining algorithms and human observers in finding meaningful patterns in mass spectra. The heuristic high frequency noise reduction approaches employed most commonly in studies to date are smoothing filters, the wavelet transform (WT), or the deconvolution filter [2]. We employed a locally weighted linear regression method with a span of 10 M/Z to smooth the spectra. Figure 2 illustrates the smoothing effect on a section of a typical spectrum.

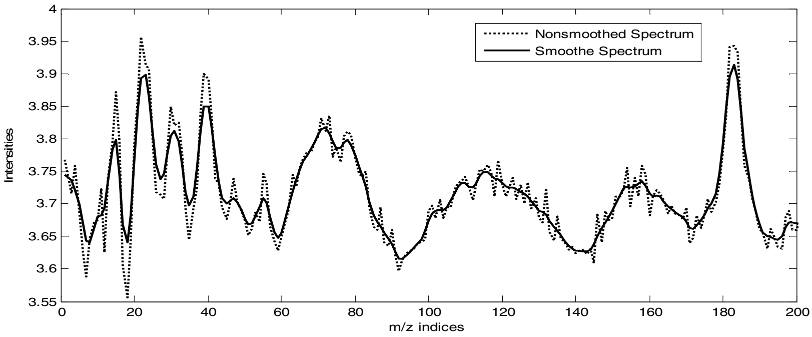

Fig. 2. The effect of the smoothing process on a part of a typical spectrum

A point in a mass spectrum indicates the relative abundance of a protein, peptide or fragment; therefore, the magnitudes of mass spectra cannot be directly compared with each other. Normalization methods scale the intensities of mass spectra to make mass spectra comparable. We normalized the group of mass spectra using total ion current (TIC) method. Figure 3 demonstrates the effect of the preprocessing stages we have applied on a typical mass spectrum from the “Ovarian 8-7-02” dataset.

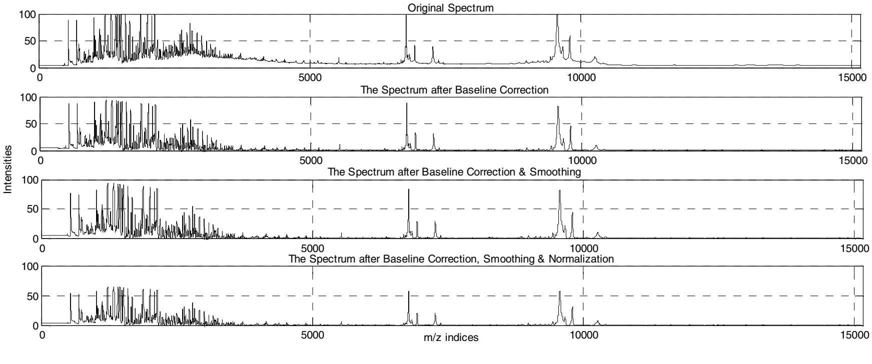

Fig. 3. The effect of preprocessing stages on a typical mass spectrum: The original spectrum (first panel), the spectrum after baseline correction (second panel), the spectrum after baseline correction and smoothing (third panel) and the spectrum after baseline correction, smoothing and normalization (last panel)

3.1 Feature Extraction and Selection

In the present study we use all m/z points as initial features and select the final features set using a t-test with correlation elimination approach. The t-test algorithm with correlation elimination can be succinctly described by the following two steps:

1) Select the first feature based on t-test score as given in equation (1).

$$t = \frac{\left(\overline{x_1} - \overline{x_2}\right)}{\sigma_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \tag{1}$$

where

$$\sigma_p{}^2 = \frac{(n_1 - 1)\sigma_1{}^2 + (n_2 - 1)\sigma_2{}^2}{n_1 + n_2 - 2} \tag{2}$$

is the pooled standard variance, and $\overline{x_i}$, for i = 1 or 2 is the mean of the putative variable in class i, and n_i, for i = 1 or 2 is the size of class i.

2) For each of the rest of the potential features, calculate the correlation and local information, w_1 and w_2 respectively, between the applied variable and all previously selected features.

$$\begin{gathered} w_1 = 1 - R \\ w_2 = 1 - e^{-\left(d/10\right)^2} \end{gathered} \tag{3}$$

where R is the Pearson correlation given in Equation (4),

$$R(x, y) = \frac{Cov(x, y)}{\sqrt{Var(x) \cdot Var(y)}} \tag{4}$$

and d is the distance between the candidate feature and all previously selected features.

From these two steps the score for each feature, designated as FS, is then calculated as the product of the t-test and the correlation scores as illustrated in Equation (5).

$$FS = t \times w_1 \times w_2 \tag{5}$$

3.2 Base Learning Algorithms

We test our approach using six well-known base classification algorithms. The following classification algorithms represent a variety of approaches and therefore allow us to assess the robustness of our approach across a variety of classification algorithms. The following learning algorithms have each been applied to the two mass-spectrum data-sets described above as stand alone classifiers using the top 50 features and as base classifiers in our hybrid random subspace fusion ensemble approach.

- Decision Trees
- Linear Discriminant Analysis (LDA)
- 1-Nearest Neighbor (1-NN)

- Logistic Regression
- Linear Support Vector Machines (with a linear kernel)
- Multi Layer Perceptron (MLP) with two hidden layers and 10 neurons in each layer, all the nonlinear functions are tangent-sigmoid and weights were randomly initialized to values in [-1, 1]. The learning function is gradient descent with momentum and back-propagation training was pursued until a limit of 100 epochs or an error of 0 was attained).

3.3 The Proposed Hybrid Random Subspace Classifier Fusion Ensemble Strategy

The heart of our hybrid approach is to randomly choose a subset of training samples and a subset of top features for each of the classifiers that will participate in the ensemble. This approach is hypothesized to maximize the diversity of the ensemble, which has been shown to be an essential feature of effective ensemble approaches. The following steps summarize the proposed strategy for the two-class cases (the strategy can be extended to more cases, but we leave to another paper):

1. Randomly select m samples from the training set (we set m = 60% of training set size)
2. Randomly select n features from n_{max} top-ranked features (we set n =10 and n_{max} = 50)
3. Train a classification algorithm with above selected samples and features
4. Classify the testing samples with the constructed classifier and calculate the corresponding support degree by assigning the *Certainty Factor* (*CF*) to the winner class and (1-*Certinaty Factor*) to the loser class.
5. Iterate above steps for i=1 to I_{max} (we set and I_{max} =100), saving the *CF* for each iteration.
6. Construct a soft decision profile (I_{max}×2) for each test sample using the saved support degrees
7. Inferring the final class from the decision profile using an appropriate decision function. We report in this paper on our experience with Majority, Weighted Majority, Mean, and Decision Template combiners.

4 Results

We compare the performance of our ensemble to each of the base learning algorithms to establish the need for an ensemble in the first place. We then compare the performance of our hybrid random subspace fusion approach to three other well-known resampling based ensemble approaches. For each of the six base classifiers, we selected the 50 top-ranked feature determined by the t-test with correlation elimination as described above. We compared the performance of these base-classifiers to the performance of our proposed hybrid random subspace method on each of the six base learning algorithms for four different decision functions, Majority (MAJ), Weighted Majority (WMAJ), Mean and Decision Template. As described earlier, in each of 100 iterations we randomly select 10 features from the 50 top-ranked features and also randomly selected 60% of the training set. We inferred a *Certainty Factor* for each classifier by testing it over the entire training set and then applied the classifier to the testing samples. After the 100 iterations, we built a soft decision profile for each test using the inferred certainty factor of each classifier. The final classification decision in then determined using one of the four decision templates. This process was repeated 10 times for each choice of base-classifier and decision template in a full 10-fold-cross-validation framework (i.e. 100 total runs for each configuration).

For comparing the classification performance of these different configurations, we used the average of sensitivity and specificity as the performance measure. Although accuracy is the best known measure of classification performance (the number of correctly classified examples over the total number of examples in a given dataset), when class distribution is imbalanced, accuracy can be misleading because it is dominated by performance on the majority class. In two-class problems, accuracy can be replaced by sensitivity and/or specificity. Sensitivity or 'true positive rate' is the number of correctly predicted positive instances over all positive instances. It is the criterion of choice when false negatives incur high penalty, as in most medical diagnosis. Specificity or 'true negative rate' is the number of correctly predicted negative instances over all negative instances. It is used when false alarms are costly [1]. Presentation of the results using this combined measure of sensitivity and specificity allows us to present the results for a large number of different experiments in a relatively small amount of space. Given that overall performance of our approach using this measure is always above 98% we feel this condensed measure is appropriate for this short paper.

Table 1. Performance results obtained on the Ovarian 8-7-02 dataset, for each of six learning algorithms operating either as individual classifiers (utilizing 10 or 50 top features) or as part of the proposed Hybrid Random Subspace strategy utilizing one of four decision functions

Learning Algorithm	Individual Classifier Performance		Hybrid Random Subspace Fusion Ensemble Performance			
	10 Top Features	50 Top Features	Majority	Weighted Majority	Mean	Decision Template
LDA	99.76 (0.2)	**100**	99.98 (0.1)	99.98 (0.1)	99.98 (0.1)	99.98 (0.1)
1-NN	98.95 (.07)	99.23 (0.7)	**100**	**100**	**100**	**100**
Decision Tree	98.36 (0.9)	98.18 (0.5)	**99.9 (0.1)**	**99.9 (0.1)**	**99.9 (0.1)**	**99.9 (0.1)**
Logistic Regression	99.77 (0.3)	99.92 (0.2)	**99.98 (0.1)**	**99.98 (0.1)**	**99.98 (0.1)**	**99.98 (0.1)**
Linear SVMs	99.48 (0.4)	99.89 (0.2)	**99.98 (0.1)**	**99.98 (0.1)**	98.32 (0.6)	**99.98 (0.1)**
MLP	98.46 (2.1)	99.31 (1.8)	**100**	**100**	**100**	**100**

The results are presented as the Mean and Standard Dev. over all runs for each of the two datasets, Ovarian 8 and Ovarian 4 in Tables 1 and 2 respectively. The results clearly show that our propose hybrid random subspace strategy outperforms the performance of each of the six base classifiers tested. For all approaches the Ovarian 8-7-02 data is generally easier to classify, with all approaches achieving average performance above 93%. The second data set, Ovarian 4-3-02, is clearly a more difficult dataset for all of these approaches, yet our hybrid random subspace strategy still achieves higher average performance regardless of the combination function utilized. We can note that overall higher performance is achieved when using the decision template combination function.

Table 2. Performance results obtained on the Ovarian 4-03-02 dataset, for each of six learning algorithms operating either as individual classifiers (utilizing 10 or 50 top features) and operating under the proposed Hybrid Random Subspace strategy utilizing one of four decision functions

Learning Algorithm	Individual Classifier Performance		Hybrid Random Subspace Fusion Ensemble Performance			
	10 Top Features	50 Top Features	Majority	Weighted Majority	Mean	Decision Template
LDA	95.86 (1.2)	96.04 (1.8)	98.98 (0.4)	**98.99 (0.5)**	**98.99 (0.5)**	98.97 (0.5)
1-NN	90.25 (2.0	92.82 (1.3)	99.46 (0.4)	99.66 (0.3)	99.5 (0.4)	**99.82 (0.2)**
Decision Tree	90.76 (2.3)	90.69 (1.1)	99.64 (0.4)	99.73 (0.4)	99.73 (0.4)	**99.83 (0.3)**
Logistic Regression	96.64 (1.3)	96.53 (1.5)	98.88 (0.6)	98.86 (0.6)	**98.92 (0.6)**	98.69 (0.5)
Linear SVMs	95.89 (1.1)	95.3 (1.3)	**98.39 (0.5)**	98.37 (0.6)	98.32 (0.6)	97.53 (0.2)
MLP	96.06 (1.3)	95.63 (0.8)	99.14 (0.3)	99.36 (0.4)	99.36 (0.4)	**99.45 (0.5)**

Table 3. Performance reportred as the Mean and Standard Dev. of the hybrid random subspace fusion strategy and other resampling strategies, using four different decision functions

Dataset	Fusion Strategy	Performance for Different Decision Functions			
		Majority	Weighted Majority	Mean	Decision Template
Ovarian 8-7-02	Hybrid	**99.97 ±0.1**	**99.97 ±0.1**	**99.97 ±0.1**	**99.97 ±0.1**
	Bagging	99.15 ±0.5	98.33 ±0.1	99.15 ±0.5	99.12 ±0.5
	Boosting	99.27 ±0.7	98.06 ±0.8	99.10 ±0.6	98.89 ±0.9
	Random Forest	99.55 ±0.5	99.90 ±0.2	99.88 ±0.2	99.85 ±0.3
Ovarian 4-3-02	Hybrid	**99.64 ±0.1**	**99.73 ±0.4**	**99.73 ±0.4**	**99.83 ±0.3**
	Bagging	95.28 ±1.2	95.27 ±1.3	95.20 ±1.2	95.21 ±1.2
	Boosting	96.87 ±1.7	96.76 ±1.9	96.32 ±1.8	96.93 ±1.8
	Random Forest	93.53 ±1.0	95.83 ±0.7	95.70 ±0.9	96.10 ±0.7

Given that our approach is a resampling strategy, we have also compared the performance with that of three other resampling strategies, including bagging, boosting and random forest. In Table 3 we provide the performance results for each of these other resampling strategies as obtained for the same four combination functions as used above together with decision trees as the base classifier strategy. The results from the hybrid random subspace strategy as reported for decision trees above are also included to facilitate comparisons between these resampling strategies. We note that for other choices of base classifiers (e.g. LDA, Logistics Regression, etc) the performance of the other resampling strategies is generally worse and is therefore not reported here.

5 Conclusion

In this paper, we have described a new hybrid approach for combining sample subspace and feature subspaces when constructing an ensemble of classifiers. We demonstrate the usefulness of our approach on two public datasets of serum protein mass spectra from ovarian cancer research. Following appropriate preprocessing and dimensionality reduction stages, six well-known classification algorithms were utilized as the base classifiers. The results showed a clear enhancement in the performance of the base classifiers when applying the proposed method. Furthermore the performance enhancement was apparent regardless of the decision function used. Future work will investigate how robust this approach is by applying it to other datasets and testing the use of other base classifiers and combination functions.

References

[1] Hilario, M., Kalousis, A., Prados, J., Binz, P.-A.: Data mining for mass spectra-based cancer diagnosis and biomarker discovery. Drug Discovery Today: BioSilico (Elsevier Ltd) 2, 214–222 (2004)

[2] Hilario, M., Kalousis, A., Pellegrini, C., Muller, M.: Processing and Classification of Mass Spectra, Mass Spectrometry Reviews, vol. 25, pp. 409– 449 (2006)

[3] Shin, H., Markey, M.K.: A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples. Journal of Biomedical Informatics 39, 227–248 (2006)

[4] Assareh, A., Moradi, M.H.: A Novel Ensemble Strategy for Classification of Prostate Cancer Protein Mass Spectra. In: 29th IEEE EMBS Annual International Conference (2007)

[5] Bhanot, G., Alexe, G., Venkataraghavan, B., Levine, A.J.: A robust meta-classification strategy for cancer detection from MS data. Proteomics 6, 592–604 (2006)

[6] Vlahou, A., Schorge, J.O., Gregory, B.W., Coleman, R.L.: Diagnosis of Ovarian Cancer Using Decision Tree Classification of Mass Spectral Data. Journal of Biomedicine and Biotechnology 5, 308–314 (2003)

[7] Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K., Zhao, H.: Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. Bioinformatics 19, 1636–1643 (2003)

[8] Yasui, Y.: A data-analytic strategy for protein biomarker discovery: profiling of high-dimensional proteomic data for cancer detection. Biostatistics 4, 449–463 (2003)

[9] Opitz, D., Maclin, R.: Ensemble Methods: An Empirical Study. Journal of Artificial Intelligence Research 26, 169–198 (1999)

[10] Xu, A., Krzyzak, Suen, C.Y.: Methods of Combining Multiple Classifiers and their Applications to Handwriting Recognition. IEEE Trans. on Systems, Man, and Cybernetics 22 (May/June, 1992)

[11] Bunke, H., Kandel, A.: Hybrid Methods in Pattern Recognition. In: Series in Machine Perception and Artificial Intelligence, vol. 47, Word Scientific, Singapore (2002)

[12] Breiman, L.: Bagging Predictors. Machine Learning 24, 123–140 (1996)

[13] Freund, Y., Schapire, R.: Experiments with a new boosting algorithm. In: Thirteenth International Conference on Machine Learning. Bari, Italy, pp. 148–156 (1996)

[14] Schapire, R.: The Strength of Weak Learnability. Machine Learning 5, 197–227 (1990)

[15] Ho, T.K.: The random subspace method for constructing decision forests. IEEE Trans. Pattern Analysis and Machine Intelligence 21, 832–844 (1998)

[16] Breiman, L.: Random Forests. Machine Learning 45, 5–32 (2001)

[17] Kuncheva, L., Bezdek, J., Duin, R.: Decision Templates for Multiple Classifier Fusion: An Experimental Comparison. Pattern Recognition 34(2), 299–314 (2001)