Principal Direction Linear Oracle for Gene Expression Ensemble Classification

Leif E. Peterson, Matthew A. Coleman

***Abstract*— A principal direction linear oracle (PDLO) ensemble classifier for DNA microarray gene expression data is proposed. The common fusion-selection ensemble based on weighted trust for a specifier classifier was replaced with pairs of subclassifiers of the same type using PDLO to perform a linear hyperplane split of training and testing samples. The hyperplane split forming the oracle was based on rotations of principal components extracted from sets of filtered features in order to maximize the separation of samples between the pair of miniclassifiers. Eleven classifiers were evaluated for performance with and without PDLO implementation, which included k nearest neighbor (kNN), naïve Bayes classifier (NBC), linear discriminant analysis (LDA), learning vector quantization (LVQ1), polytomous logistic regression (PLOG), artificial neural networks (ANN), constricted particle swarm optimization (CPSO), kernel regression (KREG), radial basis function networks (RBFN), gradient descent support vector machines (SVMGD), and least squares support vector machines (SVMLS). PLOG resulted in the best performance when used as a base classifier for PDLO. The greatest performance for PLOG implemented with PDLO occurred for tenfold CV and 100 rotations of PC scores with fixed angles for hyperplane splits. Random rotation angles for hyperplane splits resulted in reduced performance when compared to rotations with fixed angles.**

I. Introduction

Ensemble learning has proven to result in performance levels which exceed average classifier performance[1-2]. The history of improved ensemble learning performance is founded on several premises. First, complexities inherent in data can result in complex decision boundaries that are too difficult for a single classifier to handle. The application of a given classifier is commonly hinged to a variety of assumptions surrounding a particular set of data and pattern recognition functions, each of which effect scale, robustness, and computational efficiency. Examples of classifier fusion techniques include majority voting, mixture of experts, bagging, boosting, and bootstrapping. Majority voting exploits a variety of addition, product, and weighting rules for adjusting classifier outcome to achieve better performance [3]. The mixture of experts approach determines the particular area of the feature space where each expert performs optimally, and assigns future samples to the expert that is most capable of providing a correct solution in the specific space[4-12]. Bagging ensembles randomly select independent bootstrap samples of data and build classifiers from the various sets of samples[13,14]. Ensemble learning through boosting repeatedly runs a weak classifier with sequentially derived weighted mixtures of the training data to form a composite classifier[15-18].

A requirement for ensemble classifiers is that the individual classifiers have diversity and are different from one another, otherwise there will be no improvement in results when compared with the individual classifiers. Hashem [19] has reported varying degrees of diversity such as “good” and “poor”, whose results directly translate into decreased and increased performance based on the combination of classifiers considered. Measurements made among classifiers employed during bagging indicate a decrease in diversity with an increase in the number of training instances, while for boosting the diversity increases with increasing training sample sizes [20]. An alternative approach involves overproduce-and-select, in which a pool of classifiers are spawned and then optimally selected on-the-fly by monitoring accuracy and diversity parameters such as the double-fault measure [21], measure of difficulty [22], Kohavi-Wolpert variance [23], kappa [24], and generalized diversity [25]. Despite previous efforts to enhance and refine ensemble methods, the majority of studies on ensemble construction based on diversity have yielded unsatisfactory results, since a universal theory for optimized ensemble construction does not exist[26-28].

This investigation focuses on increasing ensemble diversity through use of a principal direction linear oracle (PDLO). Instead of using different classifiers in the ensemble, a single classifier is replaced with a miniensemble of two subclassifiers to which training and testing samples are assigned after performing a linear hyperplane split on the principal directions from principal component analysis (PCA). Empirical gene expression data are used for determining whether each classifier considered resulted in better performance by itself or when applied to PDLO. Other ensemble methods such as majority voting, boosting, etc., were not employed since the goal of this study was to determine which classifier resulted in the greatest performance when used for the pair of subclassifiers in miniensembles. The effect of the number of iterations and the number of folds used in cross validation (CV) on PDLO performance was also evaluated.

II. Methods

A. DNA Microarray Data Sets Used

Data used for classification analysis were available in C4.5 format from the Kent Ridge Biomedical Data Set Repository (http://sdmc.i2r.a-star.edu.sg/rp), see Table I. The 2-class adult brain cancer data were comprised of 60 arrays (21 censored, 39 failures) with expression for 7,129 genes [29]. The 2-class adult prostate cancer data set consisted of 102 training

L.E. Peterson is with the Division of Biostatistics and Epidemiology, Dept. of Public Health, The Methodist Hospital, 6565 Fannin Street, Suite MGJ6-031, Houston, Texas 77030, USA. E-mail: peterson.leif@ieee.org.

M.A. Coleman is with the Biology and Biotechnology Research Program, Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, California 94550, USA. E-mail: coleman16@llnl.gov.

TABLE I

Data sets used for classification analysis.

Cancer	Classes-Genes-Samples	Selected*
Brain [29]	2-7129-60 (21 censored, 39 failures)	16
Prostate [30]	2-12600-102 (52 tumor, 50 normal)	11
Breast [31]	2-3170-15 (8 BRCA1, 7 BRCA2)	6
Breast [32]	2-24481-78 (34 relapse, 44 non-relapse)	17
Colon [33]	2-2000-62 (40 negative, 22 positive)	5
Lung [34]	2-12533-32 (16 MPM, 16 ADCA)	29
Leukemia [35]	2-7129-38 (27 ALL, 11 AML)	9
Leukemia [36]	3-12582-57 (20 ALL, 17 MLL, 20 AML)	13
SRBCT [37]	4-2308-63 (23 EWS, 8 BL, 12 NB, 20 RMS)	20

* Genes selected using greedy PTA.

samples (52 tumor, and 50 normal) with 12,600 features. The original report for the prostate data supplement was published by Singh et al [30]. Two breast cancer data sets were used. The first had 2 classes and consisted of 15 arrays for 8 BRCA1 positive women and 7 BRCA2 positive women with expression profiles of 3,170 genes [31], and the second was also a 2-class set including 78 patient samples and 24,481 features (genes) comprised of 34 cases with distant metastases who relapsed ("relapse") within 5 years after initial diagnosis and 44 disease-free ("non-relapse") for more than 5 years after diagnosis [32]. Two-class expression data for adult colon cancer were based on the paper published by Alon et al [33]. The data set contains 62 samples based on expression of 2000 genes in 40 tumor biopsies ("negative") and 22 normal ("positive") biopsies from non-diseased colon biopsies from the same patients. An adult 2-class lung cancer set including 32 samples (16 malignant pleural mesothelioma (MPM) and 16 adenocarcinoma (ADCA)) of the lung with expression values for 12,533 genes [34] was also considered. Two leukemia data sets were evaluated: one 2-class data set with 38 arrays (27 ALL, 11 AML) containing expression for 7,129 genes [35], and the other consisting of 3 classes for 57 pediatric samples for lymphoblastic and myelogenous leukemia (20 ALL, 17 MLL and 20 AML) with expression values for 12,582 genes [36]. The Khan et al [37] data set on pediatric small round blue-cell tumors (SRBCT) had expression profiles for 2,308 genes and 63 arrays comprising 4 classes (23 arrays for EWS-Ewing Sarcoma, 8 arrays for BL-Burkitt lymphoma, 12 arrays for NB-neuroblastoma, and 20 arrays for RMS-rhabdomyosarcoma).

B. Gene Filtering and Selection

For each data set, input genes were ranked by the F-ratio test statistic, and the top 150 were then used for gene selection. Gene selection was based on a stepwise greedy plus-take-away (PTA) method [38]. We developed a novel plus 1 take away 1 stepwise gene selection algorithm which combines Mahalanobis distance and F-to-enter and F-to-remove statistics. Gene-specific expression on each array was standardized using the mean and standard deviation over the 150 genes identified by filtering. Forward stepping was carried out to add(delete) the most(least) important genes for class separability based on squared Mahalanobis distance and the F-to-enter and F-remove statistics. Genes were entered into the model if their standardized expression resulted in the greatest Mahalanobis distance between the two closest classes and their F-to-enter statistic exceeded the F-to-enter criterion. At any step, a gene was removed if its F-to-enter statistic (F=3.84) was less than the F-to-remove criterion (F=2.71). Table I lists the number of genes selected using greedy PTA.

C. Principal Direction Linear Oracle

The Principal Direction Linear Oracle (PDLO) ensemble classifier was used to invoke a linear hyperplane split of training and testing samples into two miniclassifiers. Let $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{ip})$ be the set of feature values for sample $\mathbf{x}_i$, and $\mathbf{z}_i = (z_{i1}, z_{i2}, \dots, z_{ip})$ be the set of standardized feature values for sample $\mathbf{x}_i$. Let $\mathbf{R}$ be the $p \times p$ ("gene by gene") correlation matrix based on n training samples. By the principal axis theorem, there exists a rotation matrix $\mathbf{E}$ and diagonal matrix $\mathbf{\Lambda}$ such that $\mathbf{ERE}' = \mathbf{\Lambda}$. Pre-multiplying both sides by $\mathbf{E}$, and post-multiplying by $\mathbf{E}'$, yields the principal form (or spectral decomposition) of $\mathbf{R}$ given as

$$\underset{p \times p}{\mathbf{R}} = \underset{p \times p}{\mathbf{E \Lambda E}'} \tag{1}$$

where columns of $\mathbf{E}$ and $\mathbf{E}'$ are the eigenvectors and diagonal entries of $\mathbf{\Lambda}$ are the eigenvalues.

The concept of principal directions relies on the eigenvectors derived from PCA. Let $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m$ represent the eigenvectors associated with the m greatest eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ extracted from the correlation matrix $\mathbf{R}$. A unique characteristic of eigenvectors, i.e., principal components, derived from PCA is they are all orthogonal (uncorrelated) with one another. For each lth extracted principal component $(l = 1, 2, \dots, m)$ there exists a p-vector $(j = 1, 2, \dots, p)$ of principal component score coefficients determined with the relationship $\beta_{jl} = e_{jl}/\sqrt{\lambda_l}$. For each ith sample, the lth principal component score ("PC score") is calculated as

$$y_{il} = \beta_{1l} z_{i1} + \beta_{2l} z_{i2} + \cdots + \beta_{pl} z_{ip}. \tag{2}$$

The vector $\mathbf{y}_l$ is distributed $N(0, 1)$ and serves as a new feature representing each sample in score space. In addition, each sample can be represented in 3D space (X, Y, Z) by assuming $X_i = y_{i1}$, $Y_i = y_{i2}$, and $Z_i = y_{i3}$. The motivation for PDLO is that if a standardized data set primarily consists of two largely separated clusters, then by theory the first eigenvector $\mathbf{e}_1$ associated with the largest eigenvalue λ_1 will form a straight line connecting the centers of the two clusters, since the two clusters will define the greatest amount of variation in the data. A reliable linear hyperplane $h(\mathbf{y})$ split of the data can then be made where $y_{i2} = 0$. Samples having $y_{i2} > 0$ lie above $h(\mathbf{y})$ and are assigned to data set $\mathcal{D}_1$, whereas samples with $y_{i2} \leq 0$ lie on or below $h(\mathbf{y})$ and are assigned to $\mathcal{D}_2$. The first miniensemble is used for training and testing with $\mathcal{D}_1$ and the second miniensemble used for training and testing with $\mathcal{D}_2$. The predicted class membership of test samples in $\mathcal{D}_1$ and $\mathcal{D}_2$ are then used during construction of the confusion matrix used in performance evaluation.

An iterative scheme was employed in which PC scores for the first 3 PCs $\mathbf{y}_1$, $\mathbf{y}_2$, and $\mathbf{y}_3$ for each ith sample were rotated around the axis of the first PC, i.e., $\mathbf{y}_1$ as follows

$$\begin{pmatrix} y'_{i1} \\ y'_{i2} \\ y'_{i3} \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{i1} \\ y_{i2} \\ y_{i3} \\ 1 \end{pmatrix}. \tag{3}$$

PDLO performance was evaluated for two methods of rotation, one in which an increasing angle of rotation was used where $\theta = \text{iteration}\frac{2\pi}{\#\text{iterations}}$ and another involving randomly selected rotation angles with $\theta = U(0,1)2\pi$. Algorithm 1 lists the computational steps for employing the principal direction linear oracle to invoke a hyperplane to split of samples into a set $\mathcal{D}_1$ above the hyperplane $h(\mathbf{y})$ and a set $\mathcal{D}_2$ below.

Algorithm 1: Principal Direction Linear Oracle (PDLO)

```
Data: Eigenvector $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ from $\mathbf{E}$ associated with
      3 greatest eigenvalues for a set of $p$ training
      (testing) genes selected with greedy PTA.
Result: A set of samples, $\mathcal{D}_1$, above hyperplane $h(\mathbf{y})$ and
        set $\mathcal{D}_2$ below hyperplane $h(\mathbf{y})$
foreach iteration $j$ do
    If fixed rotation angle: $\theta = j\frac{2\pi}{\#\text{iterations}}$
    If random rotation angle: $\theta = U(0,1)2\pi$
    for sample $i \leftarrow 1$ to $n$ do
        Rotate scores $y_{i1}$, $y_{i2}$, $y_{i3}$ around axis $\mathbf{y}_1$ using
        angle $\theta$
        if $y_{i2} > 0$ then
            Sample is above hyperplane $h(\mathbf{y}) \rightsquigarrow \mathcal{D}_1$
        endif
        if $y_{i2} \leq$ then
            Sample is below hyperplane $h(\mathbf{y}) \rightsquigarrow \mathcal{D}_2$
        endif
    endfor
endfch
```

Eleven classifiers were employed for training and testing [39], [40]. These included k nearest neighbor (kNN), naïve Bayes classifier (NBC), linear discriminant analysis (LDA), learning vector quantization (LVQ1), polytomous logistic regression (PLOG), artificial neural networks (ANN), constricted particle swarm optimization (CPSO), kernel regression (KREG), radial basis function networks (RBFN), gradient descent support vector machines (SVMGD), and least squares support vector machines (SVMLS). KREG employed kernel tricks in a least squares fashion to determine coefficients which reliably predict class membership when multiplied against kernels for test samples. All 2-class and 3-class problems were solved using all possible 2-class problems. First, k-means cluster analysis was performed on all of the training samples to determine centers. Coefficients for kernel regression were determined using the least squares model

$$\boldsymbol{\alpha} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}, \tag{4}$$

where $\mathbf{H}$ is a sample × gene matrix with a linear kernel in element $h_{ij} = K(\mathbf{x}_i, \mathbf{c}_j) = \mathbf{x}_i^T\mathbf{c}_j$, where $\mathbf{c}_j$ is a center vector, i.e., mean vector, from k-means cluster analysis, and $\mathbf{y}$ is sample vector with y_i set to +1 for training samples in the first class and -1 for samples in the second class being compared in the 2-class problem. A positive value of y_i denotes membership in the first class and a negative value reflects membership in the second class. The RBFN employed the same matrix algebra as kernel regression, but was based on the RBF kernel $K(\mathbf{x}_i, \mathbf{c}_j) = \exp(-||\mathbf{x}_i - \mathbf{c}_j||)$. Note that this is not a Gaussian RBF kernel which uses $\exp(-||\mathbf{x}_i - \mathbf{c}_j||/\sigma)$ as the kernel. For SVMs, we used an L1 soft norm gradient descent-based [41] and L2 soft norm least squares approach to SVM [42]. A weighted exponentiated RBF kernel was employed to map samples in the original space into the dot-product space, given as $K(\mathbf{x}, \mathbf{x}^T) = \exp(-\frac{\gamma}{m}||\mathbf{x} - \mathbf{x}^T||)$, where m=#features. Such kernels are likely to yield the greatest class prediction accuracy providing that a suitable choice of γ is used. To determine an optimum value of γ for use with RBF kernels, a grid search was done using incremental values of γ from 2^{-15}, 2^{-13},..., 2^3 in order to evaluate accuracy for all training samples. We also used a grid search in the range of 10^{-2}, 10^{-1},..., 10^4 for the SVM margin parameter C. The optimal choice of C was based on the grid search for which classification accuracy is the greatest, resulting in the optimal value for the separating hyperplane and minimum norm $||\xi||$ of the slack variable vector. SVM tuning was performed by taking the median of parameters during grid search iterations when the test sample misclassification rate was zero.

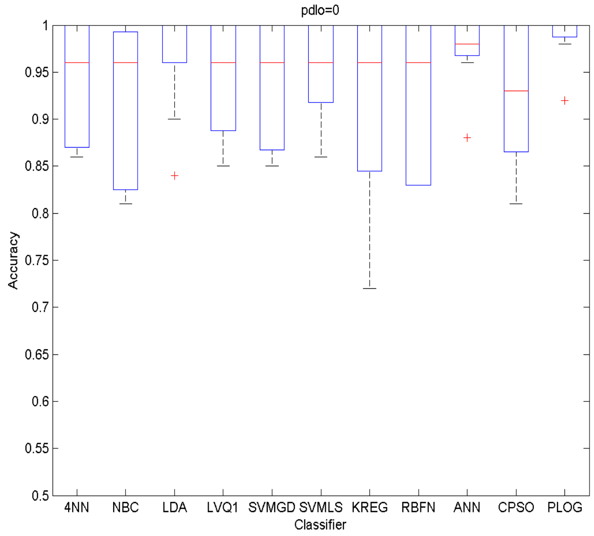

Fig. 1. Boxplot of classifier accuracy without PDLO for all data sets.

Classifier performance with and without PDLO was assessed using 10 tenfold CV [43] with 10 PDLO iterations per fold and fixed rotation angles for hyperplane splits. PLOG yielded the greatest performance when implemented with PDLO, and therefore was evaluated using CV folds of 2, 5, and 10, and 10 to 100 iterations with fixed or random rotation angles for hyperplane splits of samples. A majority voting scheme was used in which the assigned class was based on the most frequent class assignment during the iterations. Mixtures of different classifiers were not used because the focus was to establish performance of various classifiers with and without

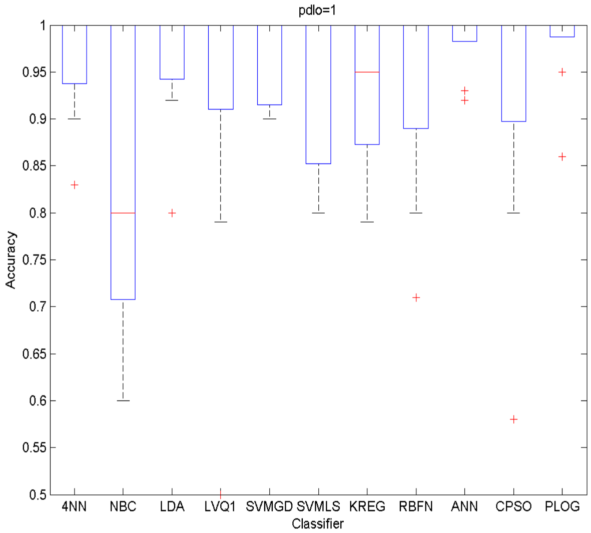

Fig. 2. Boxplot of classifier accuracy with PDLO for all data sets.

PDLO, and investigate the effects of the number of CV folds and rotation iterations on performance of PDLO.

III. Results

Figure 1 and Figure 2 show boxplots of classifier accuracy without and with PDLO for 10 tenfold CV. In the absence of PDLO (Figure 1), PLOG showed the greatest 25th percentile of accuracy for all data sets, followed by ANN and LDA. When PDLO was applied to the base classifiers, that is, use of an oracle with two miniensembles within the classifier, the same pattern emerged wherein PLOG had the greatest 25th percentile followed again by LDA and ANN (Figure 2).

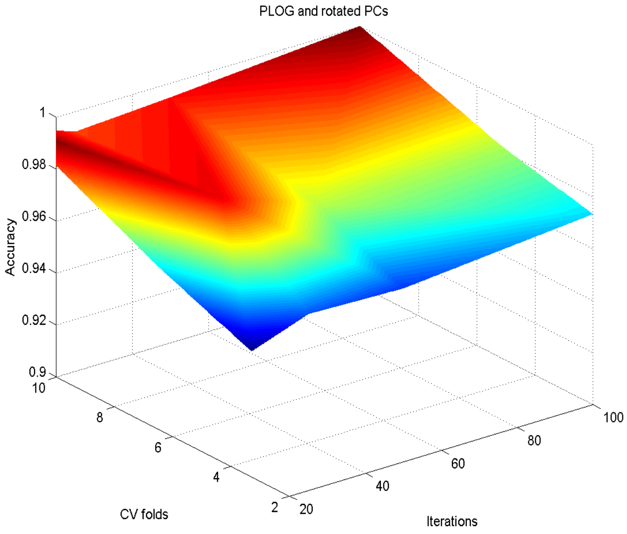

Fig. 3. PDLO accuracy as a function of CV folds and fixed rotation angles θ used during rotation of PC scores. PLOG used as base classifier for PDLO.

Figure 3 illustrates PLOG performance with PDLO as a function of CV folds and rotation iterations when fixed rotation angles were used for rotating the PC scores prior to sample hyperplane splits. Figure 4 shows that reduced performance was obtained for PLOG implemented with PDLO when random angles were employed for PC score rotations before hyperplane splits of samples.

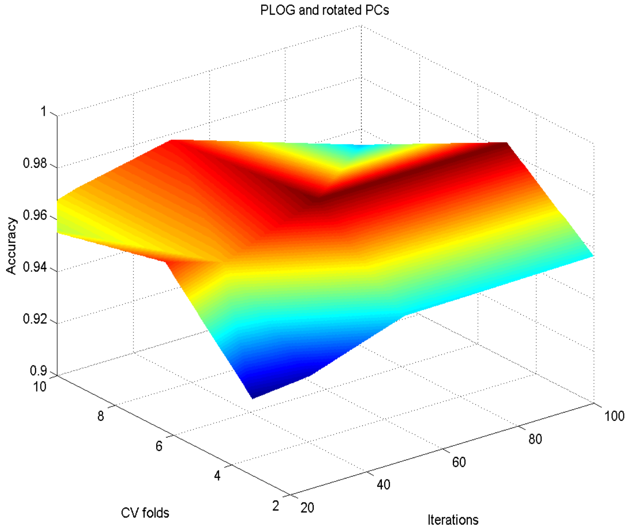

Fig. 4. PDLO accuracy as a function of CV folds and random rotation angles θ used during rotation of PC scores. PLOG used as base classifier for PDLO.

IV. Discussion and Conclusion

Linear oracles for classification are not a new concept. Their use primarily originated in the development of decision tree classifiers. Hyperplane splits first appeared in oblique decision trees in the form of axis-parallel splits used in CART-LC [44] and later OC1 [45]. Random linear oracles were recently applied by Kuncheva and Rodriguez to 35 UCI data sets using a variety of classification methods including Adaboost, bagging, multiboost, random subspace, and random forests [46]. In their study of classifier ensembles, random hyperplane splits were used in which 2 points were randomly selected and the perpendicular vector at the midpoint between the 2 points was used as a reference for the hyperplane. Superior results were obtained for the random linear oracle when compared with the routine uses of various bagging and boosting forms of decision tree methods.

In the present study, our focus was to evaluate the effect of PDLO on performance for 11 base classifiers, since to date this has eluded systematic investigation. Principal directions were used for the purpose of developing linear hyperplanes from orthogonal eigenvectors describing the majority of variance in the data. Thus far, we have observed that the greatest performance of PDLO occurred when implemented with PLOG, ANN, and LDA. Using PLOG as the base classifier, we observed that tenfold CV and 100 rotations using fixed rotation angles for hyperplane splits resulted in the greatest performance. We are currently evaluating differences in diversity among multiple classifiers used in ensembles vs. PDLO.

In conclusion, PLOG resulted in the best performance when used as a base classifier for PDLO. The greatest performance for PLOG when implemented with PDLO occurred for tenfold CV and 100 rotations of PC scores with fixed angles for hyperplane sample splits.

References

[1] J. Kittler, M. Hatef, R.P.W. Duin, J. Matas. On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions, 20(3), 226-239. 1998

[2] L.I. Kuncheva. Combining Pattern Classifiers: Methods and Algorithms. New York(NY): John Wiley, 2004.

[3] M. van Erp, L. Vuupijl, L. Shomaker. An overview and comparison of voting methods for pattern recognition. 1-6. 2002. Hoboken(NJ), IEEE. Proceedings of the 8th International Workshop on Frontiers in Handwriting Recognition (WFHR02).

[4] M.I. Jordan, R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. Neural Computation, 6, 181-214, 1994.

[5] G.J. McLachlan, D. Peel. Finite Mixture Models. John Wiley, New York(NY), 2000.

[6] S. Gutta, J.R.J. Huang, P. Jonathon, H. Wechsler. Mixture of experts for classification of gender, ethnic origin, and pose of human faces. IEEE Transactions on Neural Networks. 11(4), 948-960, 2000.

[7] M. Szummer, C.M. Bishop. Discriminative writer adaptation. In 10th International Workshop on Frontiers in Handwriting Recognition (IWFHR), 2006.

[8] K. Rose, Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. Proceedings of IEEE, 86(11), 2210-2239, 1998.

[9] K. Chen, L. Xu, H. Chi. Improved learning algorithms for mixture of experts in multiclass classification. Neural Networks. 12, 1229-1252 (1999).

[10] S-K. Ng, G.J. McLachlan. Using the EM algorithm to train neural networks: misconceptions and a new algorithm for multiclass classification.

[11] A. Rao, D. Miller, K. Rose, and A. Gersho. Mixture of experts regression modeling by deterministic annealing. IEEE Transactions on Signal Processing. 45(11), 2811-2820, 1997.

[12] C. Bishop, M. Tipping. A Hierarchical Latent Variable Model for Data Visualisation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 20(3), 281-293, 1998.

[13] L. Breiman. Bagging predictors. Machine Learning. 24(2), 123-140, 1996.

[14] B. Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. J. American Stat. Assoc. 1983. 78:316-331.

[15] Y. Freund, R.E. Schapire. Experiments with a New Boosting Algorithm, in Proceedings of the Thirteenth International Conference on Machine Learning, 148-156. San Francisco, Morgan Kaufmann, 1996.

[16] Y. Freund, R.E. Schapire. A short introduction to boosting. Journal of Japanese Society for Artificial Intelligence, 14(5):771-780, 1999.

[17] Y. Qu, B.H. Adam, Y. Yasuo, M.D. Ward, L.H. Cazres, P.F. Schellhammer, Z. Feng, O.J. Semmes, G.L. Wright. Boosted decision tree analysis of surface-enhanced laser desorption/ionization mass spectral serum profiles discriminates prostate cancer from noncancer patients. Clinical Chemistry 2002. 48(10): 1835-1843.

[18] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos. Boosting linear discriminant analysis for face recognition. ICIP (1), 657-660, 2003.

[19] S. Hashem. Treating harmful collinearity in neural network ensembles. In: Combining Neural Networks (ed: A.J.C. Sharkey), pp. 101-125. London, Springer-Verlag, 1999.

[20] M. Skurichina, L.I. Kuncheva, R.P.W. Duin. Bagging and boosting for the nearest mean classifier: effects of sample size on diversity and accuracy. Lecture Notes in Computer Science 2002;2364:62-71.

[21] G. Giacinto, F. Roli. Design of effective neural network ensembles for image classification processes. Image Vision and Computing. 19(9-10), 699-707, 2001.

[22] L.K. Hansen, P. Solamon. Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence. 12(10), 993-1001, 1990.

[23] R. Kohavi, D.H. Wolpert. Bias plus variance decomposition for zero-one loss functions. Proc. 13th Int. Conf. on Machine Learning, pp. 275-283. San Francisco, Margan Kaufman, 1996.

[24] D. Margineantu, T. Dietterich, T. Pruning adaptive boosting. In Proceedings of the Fourteenth International Conference on Machine Learning, Morgan Kaufmann, San Francisco, 1997.

[25] W.J. Krzanowski, D. Partridge. Software Diversity: practical statistics for its measurement and exploitation, Res. Report 324. Exeter(UK), Univ. of Exeter, 1995

[26] L.I. Kuncheva, C.J. Whitaker. Measures of diversity in classifier ensembles. Machine Learning 2003;51:181-207.

[27] L.I. Kuncheva. That Elusive Diversity in Classifier Ensembles. Lecture Notes in Computer Science 2003;2652:1126-38.

[28] C.A. Shipp, L.I. Kuncheva. Relationships between combination methods and measures of diversity in combining classifiers. Information Fusion 2002;3:135-48.

[29] S.L. Pomeroy, P. Tamayo, M. Gaasenbeek, L.M. Sturla, M. Angelo, M.E. McLaughlin, J-Y.H. Kim, L.C. Goumnerovak, P. M. Blackk, C. Lau, J.C. Allen, D. ZagzagI, J.M. Olson, T. Curran, C. Wetmore, J.A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califanokk, G. Stolovitzkykk, D.N. Louis, J.P. Mesirov, E.S. Lander, T.R. Golub, Prediction of central nervous system embryonal tumour outcome based on gene expression, Nature. 415(6870) (2002) 436-442.

[30] D. Singh, P.G. Febbo, K. Ross, D.G. Jackson, J. Manola, C. Ladd, P. Tamayo, A.A. Renshaw , A.V. D'Amico, J.P. Richie, E.S. Lander, M. Loda, P.W. Kantoff, T.R. Golub, W.R¿ Sellers, Gene expression correlates of clinical prostate cancer behavior, Cancer Cell. 1(2) (2002) 203-209.

[31] I. Hedenfalk, D. Duggan, Y. Chen et al, Gene-expression profiles in hereditary breast cancer, N. Engl. J. Med. 344 (2001) 539-548.

[32] L.J. van 't Veer, H. Dai, M.J. van de Vijver, Y.D. He, A.A. Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, G.J. Schreiber, R.M. Kerkhoven, C. Roberts, P.S. Linsley, R. Bernards, S.H. Friend, Gene expression profiling predicts clinical outcome of breast cancer, Nature. 415 (2002) 530-536.

[33] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine, Broad patterns of gene expression revealed by clustering of tumor and normal colon tissues probed by oligonucleotide arrays, Proc. Natl. Acad. Sci. USA. 96(12) (1999) 6745-6750.

[34] G.J. Gordon, R.V. Jensen, L.L. Hsiao, S.R. Gullans, J.E. Blumenstock, S. Ramaswamy, W.G. Richards, D.J. Sugarbaker, R. Bueno, Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma, Cancer Res. 62(17) (2002) 4963-5967.

[35] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, E.S. Lander, Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression, Science. 286 (1999) 531-537.

[36] S.A. Armstrong, J.E. Staunton, L.B. Silverman, R. Pieters, M.L. den Boer, M.D. Minden, S.E. Sallan, E.S. Lander, T.R. Golub, S.J. Korsmeyer, MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia, Nature Genetics. 30(1) (2001) 41-47.

[37] J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu CR, C. Peterson, R.S., Meltzer, Classification and diagnostic prediction

of cancers using gene expression profiling and artificial neural networks, Nature Med., 7 (2001) 673-679.

[38] P. Somol, P. Pudil, J. Nonovicova, J. Paclik. Adaptive floating search methods in feature selection. Pattern Recognition Letters 1999;20:1157-63.

[39] L.E. Peterson, M.A. Coleman. Machine learning-based receiver operating characteristic (ROC) curves for crisp and fuzzy classification of DNA microarrays in cancer research. Int. J. Approx. Reasoning, (in press).

[40] L.E. Peterson, R.C. Hoogeveen, H.J. Pownall, J.D. Morrisett. Classification Analysis of Surface-enhanced Laser Desorption/Ionization Mass Spectral Serum Profiles for Prostate Cancer. Proceedings of the 2006 IEEE World Congress on Computational Intelligence (WCCI 2006).

[41] N. Christianini, J. Shawe-Taylor. 2000. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge Univ. Press, Cambridge.

[42] S. Abe. Support Vector Machines for Pattern Classification. Advances in Pattern Recognition Series. Springer, Berlin, 2005.

[43] R. Kohavi. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. International Joint Conference on Artificial Intelligence (IJCAI) 1995;1137-45.

[44] L. Breiman, J. Friedman, R. Olshen, C. Stone. Classification and Regression Trees. Boca Raton(FL), Chapman & Hall/CRC, 1984.

[45] S.K. Murthy, S. Kasif, and S. Salzberg. A system for induction of oblique decision trees. Journal of Artificial Intelligence Research. 2, 1-33, 1994.

[46] L.I. Kuncheva, J.J. Rodriguez. Classifier Ensembles with a Random Linear Oracle. IEEE Transactions on Knowledge and Data Engineering. 19(4) 500-508, 2007.