Informative Gene Set Selection Via Distance Sensitive Rival Penalized Competitive Learning and Redundancy Analysis

Liangliang Wang and Jinwen Ma*

Department of Information Science, School of Mathematical Sciences and LMAM, Peking University, Beijing, 100871, China
jwma@math.pku.edu.cn

Abstract. This paper presents an informative gene set selection approach to tumor diagnosis based on the Distance Sensitive Rival Penalized Competitive Learning (DSRPCL) algorithm and redundancy analysis. Since the DSRPCL algorithm can allocate an appropriate number of clusters for an input dataset automatically, we can utilize it to classify the genes (expressed by the gene expression levels of all the samples) into certain basic clusters. Then, we apply the post-filtering algorithm to each basic gene cluster to get the typical and independent informative genes. In this way we can obtain a compact set of informative genes. To test the effectiveness of the selected informative gene set, we utilize the support vector machine (SVM) to construct a tumor diagnosis system based on the express profiles of its genes. It is shown by the experiments that the proposed method can achieve a higher diagnosis accuracy with a smaller number of informative genes and less computational complexity in comparison with the previous ones.

1 Introduction

Microarray data or gene expression profiles have been widely used in many applications, especially on tumor diagnosis (e.g., [1], [2]). Given a set of samples labelled "tumorous" or "normal", the task of tumor diagnosis is to build a binary classifier as a diagnosis system to predict the unlabelled samples. Mathematically, a microarray dataset of N genes and d samples can be represented by a matrix $(x_i^\mu)_{N\times d}$, where the element x_i^μ represents the expression level of the μ-th gene at the i-th sample. Generally, there are thousands of genes in a microarray chip and so high dimensional data would cause a series of problems, such as high computing complexity, low prediction accuracy and unexplainable biological meanings [3]. Moreover, in comparison with the number of genes, we can only collect a small number of samples at present because of the high expense. In fact, there are only a small number of genes which are related or informative to a tumor. Therefore, informative gene selection to a tumor is often used as a preprocessing technique in the tumor diagnosis or classification.

* Corresponding author.

D. Liu et al. (Eds.): ISNN 2007, Part I, LNCS 4491, pp. 1227–1236, 2007.

Actually, informative gene selection (to a tumor), i.e., finding the genes that are discriminative between the normal and tumorous phenotypes, has been studied extensively in the past several years. Typically, informative genes are selected by ranking genes according to a kind of criterion, such as t, F, rank sum and χ^2 test statistics [4],[5],[6], [7]. Generally, these traditional methods just select the top k genes (k is a fixed positive integer). In this way, there again appears a serious problem that informative genes selected through individual gene evaluations are often highly correlated, which also leads to a low prediction accuracy of the diagnosis system.

To maintain a high prediction accuracy of the diagnosis system, we should find a set of uncorrelated or independent but still highly informative genes. In order to do so, many researchers have made the redundancy analysis on genes for selecting the independent informative gene set for tumor diagnosis [8],[9],[3]. However, many of these approaches are too sensitive to the order of genes according to their individual ranks such that too many genes are eliminated and thus some useful information may be lost. On the other hand, some valuable information can also be discovered by evaluating the classification capability of combinations of genes [10]. To this end, we established a post-filtering gene selection algorithm to select informative genes of a tumor with a microarray dataset based on redundancy and multi-gene analysis [11].

To further improve the efficiency of informative gene set selection, we can consider the structure of genes expressed by the rows of the matrix of the microarray dataset. That is, these genes consists of some different functional clusters to the tumor. If we can get these clusters, we can implement the post-filtering gene selection algorithm on each cluster to select the typical and independent genes. In this way, we can get a compact set of informative genes which is efficient for the tumor diagnosis.

Based on the above idea, this paper further proposes a new approach to the informative gene set selection. Since the Distance Sensitive Rival Penalized Competitive Learning (DSRPCL) algorithm [12], as a generalization of the original rival penalized competitive learning algorithm [13], can allocate an appropriate number of clusters for an input dataset automatically, we utilize it to classify the genes into a number of functional clusters. Then, we use the post-filtering gene selection algorithm on each cluster to select the typical and independent informative genes. Finally, we get the compact set of informative genes for tumor diagnosis.

In the sequel, we introduce the DSRPCL algorithm for discovering the functional gene clusters in Section 2. Section 3 describes the post-filtering gene selection algorithm on each gene cluster for typical and independent informative gene selection. The experiment results of the proposed informative gene set selection method as well as their comparisons are presented in Section 4. Finally, we conclude the paper in Section 5.

2 The DSRPCL Algorithm for Functional Gene Clusters

Given a microarray dataset $(x_i^\mu)_{N\times d}$ of N genes and d samples, we let it be $S = \{X^\mu\}_{\mu=1}^N$, where $X^\mu = [x_1^\mu, x_2^\mu, \cdots, x_d^\mu]^T$ represents the μ-th gene through its expression levels over all the d samples. Suppose that X^μ is just an input to a simple competitive learning network, which only has one layer of units. Initially there are n units with the weight vector $W_i = [w_{i1}, w_{i2}, \cdots, w_{id}]^T$ respectively for the i-th unit. All the weight vectors can be represented by a big vector $W = \text{vec}[W_1, W_2, \cdots, W_n]$. For each input X^μ, the basic idea of the DSRPCL algorithm is that not only the weight vector of the winner unit is modified to adapt to the input, but also the weight vectors of the rivals or losers are punished to keep away from the input. As a weight vector diverges to infinity, the corresponding cluster becomes empty and can be canceled. As a result, we can automatically obtain the number of gene clusters as well as their centers, or "representative genes", of these clusters assuming n is larger than the true number of the actual gene clusters. As a result, the genes are automatically divided into several functional clusters by classifying each gene into the cluster whose center is closest to it.

Table 1. The DSRPCL algorithm and its variants

1 Randomly initialize the vector $W_1^{(0)}, \cdots, W_n^{(0)}$, and let $T = 0$.

2 Update W_i with a learning rate η $(0 \le \eta \le 1)$:

1) Batch DSRPCL:

$$\Delta W_i = -\eta \frac{\partial E(W)}{\partial W_i} = \begin{cases} \eta \sum_\mu (X^\mu - W_i), & \text{if} \quad i = c(\mu), \\ -\eta \sum_\mu \|X^\mu - W_i\|^{-P-2}(X^\mu - W_i), & \text{otherwise.} \end{cases}$$

2) DSRPCL1:

$$\Delta W_i = \begin{cases} \eta(X^\mu - W_i), & \text{if} \quad i = c(\mu), \\ -\eta\|X^\mu - W_i\|^{-P-2}(X^\mu - W_i), & \text{otherwise.} \end{cases}$$

3) DSRPCL2:

$$\Delta W_i = \begin{cases} \eta(X^\mu - W_i), & \text{if} \quad i = c(\mu), \\ -\eta\|X^\mu - W_i\|^{-P-2}(X^\mu - W_i), & \text{if} \quad i = r(\mu), \\ 0, & \text{otherwise.} \end{cases}$$

4) SARPCL:

a) Let $\lambda = e^{(-k_1 T - k_0)}$, $\eta = \eta_0/(c_1\ T + c_0)$ and $t = 0$.

b) Randomly select X^μ from $S = \{X^1, \ldots, X^N\}$, and take $\xi \sim$Uniform$[0, 1]$.

c) $$\Delta W_i = \begin{cases} \eta(X^\mu - W_i), & \text{if} \quad i = c(\mu), \\ -\eta\|X^\mu - W_i\|^{-P-2}(X^\mu - W_i), & \text{otherwise.} \end{cases}$$

If $\xi \le \lambda$, let $\Delta W_i = -\Delta W_i$.

d) If $t < M$, let $t = t + 1$ and return to STEP b).

e) If $\lambda < \varepsilon$, stop.

3 If $|E(W)^{(T+1)} - E(W)^{(T)}| > \varepsilon_1$, let $T = T + 1$, and return to STEP 2; otherwise, stop.

Mathematically, the DSRPCL algorithm tries to minimize the following cost function:

$$\begin{aligned} E(W) &= E_1(W) + E_2(W) \\ &= \frac{1}{2}\sum_{\mu} \|X^{\mu} - W_{c(\mu)}\|^2 + \frac{1}{P}\sum_{\mu, i \neq c(\mu)} \|X^{\mu} - W_i\|^{-P}, \end{aligned} \tag{1}$$

where $c(\mu)$ is the index of the winner unit for the μ-th gene. That is, $W_{c(\mu)}$ is the nearest weight vector for X^{μ}. P is a positive constant. Ma and Wang[12] obtained the derivatives of $E(W)$ with respect to w_{ij} as follows:

$$\frac{\partial E(W)}{\partial w_{ij}} = -\sum_{\mu} \delta_{i,c(\mu)}(x_j^{\mu} - w_{ij}) + \sum_{\mu}(1 - \delta_{i,c(\mu)})\|X^{\mu} - W_i\|^{-P-2}(x_j^{\mu} - w_{ij}) \tag{2}$$

where $\delta_{i,j}$ is the Kronecker function. The DSRPCL algorithm is just a gradient descent algorithm based on the above derivatives of the cost function $E(W)$. Table 1 gives the details of the DSRPCL algorithm and its variants, where we denote it as the batch DSRPCL algorithm, the DSRPCL1 algorithm is the adaptive DSRPCL algorithm, and the DSRPCL2 algorithm modifies only the rival weight vector (i.e., the second winner) so that $E_2(W)$ is only affected by the largest term with $r(\mu)$. The other variant of the DSRPCL algorithm is the simulated annealing rival penalized competitive learning (SARPCL) by applying the simulated annealing mechanism to the DSRPCL1 algorithm. The stopping threshold value ε is a pre-fixed small positive number. k_0, k_1, c_0 and c_1 are positive constant numbers which can be selected by experience.

According to the properties of the DSRPCL algorithm shown in [12], when n is selected to be large enough, the DSRPCL algorithm can lead to a number of functional gene clusters from which we can detect the typical and independent informative genes more efficiently.

3 The Post-filtering Gene Selection Algorithm for Discovering the Independent Informative Genes

In order to attain a compact set of independent and informative genes to a tumor, we can remove the redundant informative genes through the post-filtering gene selection algorithm proposed in [11]. Initially, genes are ranked by the goodness of classification or diagnosis with the individual gene expression profile on the samples, which is measured through a statistical test under the null hypothesis that no difference exists between the gene expression profiles of tumorous and normal samples. The first gene will certainly be selected, but two types of redundant genes are identified and dealt with differently from the second gene to the last one. Let A be the expression profile of a selected gene and B be that of another candidate gene. Firstly, B will be abandoned if it is highly correlated with A because no much valuable classification information is added. Secondly, B is possibly redundant if the p-value of the test on the combination of genes B and A is greater than that of A alone. Otherwise, B will be chosen as a new informative gene. Furthermore,

Table 2. The post-filtering gene selection algorithm

0. Let iteration number $i = 0$, and the output gene set $O = \emptyset$,.
Order the gene sets:
1. If $i == 0$, let $S^i = \{X^{\mu_1}\}_{\mu_1=1}^{N_1}$, N_1 is the initial number of genes; otherwise, $S^i = \{S^i_{\mu_i}\}_{\mu_i=1}^{N_i}$, where $S^i_{\mu_i} = \{R^{i-1}_l, R^{i-1}_j\}$, $i > 1$, and $l, j = 1, \cdots, \text{length}(R^{i-1})$, satisfying $S^i_l \cap S^i_j = \emptyset$ and $T(S^i_l) \leq T(S^i_j)$ for $i \geq 1$, $\forall\ l < j$, and $l,\ j = 1, \cdots, N_i$.
Redundancy analysis:
2. Let $r = 0,\ \mu_i = 1$.
3. If $T(S^i_{\mu_i}) \geq \alpha$, let $r = r + 1,\ R^i_r = S^i_{\mu_i}$; else
1) let $j = \mu_i + 1$.
2) if $\text{corr}(S^i_{\mu_i}, S^i_j) > \beta$, let $S^i = S^i - S^i_j$; elseif $T(S^i_{\mu_i}) < T(S^i_{\mu_i}, S^i_j)$, let $r = r + 1,\ R^i_r = S^i_j$
3) if $j < \text{length}(S^i)$, let $j = j + 1$, go to STEP 2); otherwise, stop.
4. If $\mu_i < \text{length}(S^i)$, let $\mu_i = \mu_i + 1$, go to STEP 3.
5. $O = O \cup S^i$.
6. If $i < M$, let $i = i + 1$, got to STEP 1.

the possibly redundant genes are combined into subsets and evaluated again from the view of multi-genes in a similar way with the individual genes.

Based on the above ideas, the post-filtering gene selection algorithm can be descried in Table 2. The i^{th} iteration of the algorithm evaluates the classification capability of 2^{i-1} genes and removes the redundant ones. S^i represents an ordered list of subsets of 2^{i-1} genes according to p-values of a kind of statistical test denoted by $T(\cdot)$. In the redundancy analysis, an element in S^i will be removed if its correlation coefficient with other selected gene subsets are greater than β. Furthermore, an element in S^i will be moved to R^i if it attains a p-value less than α or that of the combination with some already selected gene subsets. The $(i+1)^{th}$ iteration will evaluate larger gene subsets by combining two smaller ones from R^i. If α is relatively large, we tend to select genes with good classification by individual evaluations. On the other hand, if α is too small, we tend to select genes with good classification by multi-gene analysis. We can stop the algorithm when no more informative genes can be found in an iteration. Alternatively, for simplicity, the algorithm stops when it reaches a user-specified maximum number of iterations M.

As we implement the post-filtering gene selection algorithm on each of the functional gene clusters obtained via the DSRPCL algorithm, we can get a set of typical independent informative genes from different aspects with a high speed since the number of genes in each cluster is generally much decreased.

4 Experimental Results and Comparisons

We tested the effectiveness of our proposed gene set selection method on the colon cancer dataset[1] through the support vector machine. This dataset contains the expression profiles of 2000 genes in 22 normal tissues and 40 tumor tissues.

[1] Retrieved from `http://microarray.princeton.edu/oncology/database.html`

Before our experiments, we normalized the gene expression profiles with zero mean and unit variance in order to eliminate the possible noise. For a tumor diagnosis system, i.e., a binary classifier, we took its prediction accuracy as the evaluation metric of this method. We used the radial basis functions (RBF's) as the kernels to build SVMs with a MATLAB toolbox called $OSUSVM3.0$ [2]. There were two parameters γ and C to be selected. We took a grid search procedure from 16×16 pairs of γ and C ($\gamma,\ C = 2^{-7}, 2^{-6}, \cdots, 2^{8}$), and chose the values optimizing the performance of SVMs on the training dataset [14].

Table 3. Results of clustering on the colon dataset

clustering method	n	K	the number of genes in each cluster
Batch DSRPCL	12	6	560, 413, 186, 36, 317, 488
DSRPCL1	10	6	178, 412,484, 351, 366,209
DSRPCL2	10	6	353, 409,40, 518, 193, 487
SARPCL	10	6	153, 55, 421, 248, 470,653

The first step was to cluster gene profiles using the DSRPCL algorithm and its variants, which stopped when the difference of the cost functions $E(W)$ between two successive steps was less than a threshold value 1.e-3. In addition, the selection of parameters was important in the DSRPCL algorithm and its variants. If P was too small, the power of de-learning might become so strong that the result of clustering was wrong. P was usually selected around 0.15 [12].

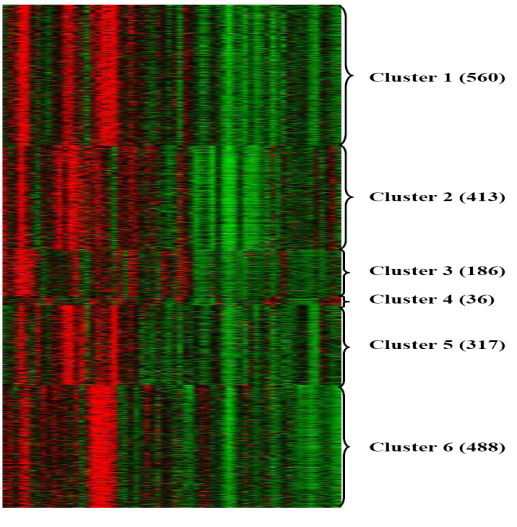

Fig. 1. The clustering result of Batch DSRPCL on the colon dataset

[2] Available on http://eewww.eng.ohio-state.edu/ ~ maj /osu_svm

With the initial number of clusters n, Table 3 showed the final number of clusters K and the number of genes in each cluster after clustering by DSRPCL and its variants on the colon dataset. For the SARPCL algorithm, we set $t = 500$, $k_1 = 0.005$, $k_0 = 1.200$, $c_1 = 0.015$ based on experiments on a small simulated dataset. Both DSRPCL and it variants automatically clustered genes into 6 clusters. Figure 1 illustrated the result of Batch DSRPCL clustering through the program *TreeView* [3], in which every row represented a gene profile and every column a sample. In this experiment, we applied the hierarchical clustering method to samples before clustering genes for the convenience of visual investigation. The positive expression was shown in red, and the negative expression was shown in green. Visually, 6 clusters distinguished themselves with others well, which might imply some biological significance.

Table 4. The LOOCV results of several methods of informative gene selection (Numbersu in brackets showed the number of selected informative genes)

clustering method	α	0.001	0.005	0.01	0.05
Traditional method	i=0	90.3% (60)	91.9% (137)	91.9% (188)	88.7% (389)
Post-filtering	i=1	91.9% (7)	91.9% (10)	91.9% (10)	91.9% (13)
	i=2	93.5% (9)	95.2% (19)	95.2% (22)	95.2% (34)
	i=3	93.5% (10)	96.8% (23)	98.4% (28)	95.2% (49)
Batch DSRPCL	i=1	93.5% (11)	100% (13)	100% (14)	96.8% (19)
	i=2	95.2% (65)	93.5% (59)	95.2% (56)	95.2% (67)
	i=3	96.8% (145)	96.8% (147)	95.2% (148)	96.8% (135)
DSRPCL1	i=1	95.2% (15)	98.4% (17)	96.8% (17)	93.5% (23)
	i=2	93.5% (57)	95.2% (59)	96.8% (59)	95.2% (67)
	i=3	96.8% (137)	96.8% (119)	96.8% (123)	96.8% (127)
DSRPCL2	i=1	93.5% (13)	95.2% (16)	96.8% (17)	96.8% (26)
	i=2	93.5% (65)	96.8% (58)	96.8% (57)	93.5% (70)
	i=3	96.8% (145)	96.8% (122)	96.8% (121)	96.8% (130)
SARPCL	i=1	96.8% (14)	98.4% (13)	98.4% (16)	95.2% (20)
	i=2	95.2% (58)	95.2% (59)	95.2% (64)	93.5% (72)
	i=3	95.2% (134)	95.2% (147)	95.2% (132)	93.5% (140)

The second step was to apply the post-filtering informative gene selection algorithm to each gene cluster. For each cluster, we conducted the leave-one-out cross-validation (LOOCV) experiments [15]. The classifier was successively learned on $d - 1$ samples and tested on the remaining one. We applied the gene set selection in each cross-validation trial on the training samples of that trial. The construction of a classifier was restricted to the selected informative genes using the training data. Finally, we computed the average prediction accuracy

[3] Which can be downloaded from `http://rana.lbl.gov`

and the number of informative genes for the d results as our evaluation result. Here, we used 2-sample rank sum test for individual genes and 2-sample Hotelling T^2 test as the multivariate statistical test on multiple genes, and chose $\beta = 0.6$ [11]. The prediction accuracies of several methods were presented in Table 4. The traditional method selected genes only through rank sum test [6] without further filtering ($i = 0$). Post-filtering chose genes of more informative by applying the post-filtering algorithm to those selected by the traditional method. From Table 4, while the best prediction accuracy achieved by the post-filtering algorithm was 98.4% using 28 genes, the prediction accuracy could achieve 100% by the Batch DSRPCL algorithm using only 13 genes. Moreover, we noticed the post-filtering method often needed several iterations to reach a higher prediction accuracy, but using the DSRPCL algorithm and its variants usually achieved the highest prediction accuracy in the first iteration because the clustering process provided some useful information for finding the informative genes.To compare with the traditional clustering methods, we used the profile clustering program *Cluster*[4] written by Eisen to cluster genes on the colon dataset by using these methods. Here, we set $\alpha = 0.005$, $\beta = 0.6$, and used rank sum test. By letting the number of cluster be 6, K-means clustering method completed the clustering after 35 iterations. The numbers of genes in every cluster were 75, 1042, 444, 28, 356 and 55, respectively. By using the hierarchical clustering method, when the genes were divided into 6 clusters the numbers of genes in every cluster were 4, 293, 341, 471, 5 and 886, respectively. From the results in Table 5, it can be found that the DSRPCL algorithm and its variants as the unsupervised clustering methods were better than the traditional clustering methods. Particularly, the DSRPCL methods not only needed not specify the number of clusters, but also led to better prediction accuracies.

Table 5. The LOOCV results of post-filtering algorithm using different clustering methods (Numbers in brackets showed the number of selected informative genes)

	hierarchical	K-means	Batch DSRPCL	DSRPCL1	DSRPCL2	SARPCL
i=1	87.1% (16)	90.3% (19)	100% (13)	98.4% (17)	95.2% (16)	98.4% (13)
i=2	96.8% (82)	95.2% (73)	93.5% (59)	95.2% (59)	96.8% (58)	95.2% (59)
i=3	95.2% (166)	96.8% (153)	96.8% (147)	96.8% (119)	96.8% (122)	95.2% (147)

5 Conclusions

Traditional informative gene set selection methods are most simple and fast, but selected genes may be highly correlated and redundant, which strongly influences the prediction or diagnosis accuracy of a tumor. The post-filtering gene selection algorithm tries to overcome this redundancy problem. However, it becomes very slow if the number of genes is very large. Moreover, it is also difficult to explain

[4] Which can be downloaded from `http://rana.lbl.gov`

these selected informative genes biologically. By utilizing the DSRPCL algorithm and its variants to cluster genes automatically, we can divide genes into several functional clusters without specifying the number of clusters in advance. In such a way, the post-filtering gene selection algorithm performs more effectively on each cluster due to the low computing complexity. Of course, the clustering process itself is time-consuming, but it can result in some relatively uncorrelated gene clusters which not only benefit the explainable biological meaning but also aid selecting most unique and uncorrelated genes. All in all, the proposed informative gene set selection based on DSRPCL and redundancy analysis can achieve higher diagnosis accuracy with a smaller number of informative genes and might reduce the computing complexity and benefit the biological explanation.

Acknowledgements

This work was supported by the Natural Science Foundation of China for Project 60471054.

References

1. Dudoit D.S., Fridyand J., Speed T. P.: Comparison of Discrimination Methods for the Classification of Tumor Using Gene Expression Data. Journal of American Statistical Association **97** (2002) 77-87
2. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene Selection for Cancer Classification Using Support Vector Machine. Machine Learning **46** (2002) 389-422
3. Yu, L., Liu, H.: Redundancy Based Feature Selection for Microarray Data. Proceedings of the Tenth ACM Conference on Knowledge Discovery and Data Mining (SIGKDD'04) (2004) 737-742
4. Golub T.R., Slonim D.K., Tamayo P., et al.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science **286** (1999) 531-537
5. Ding C.: Analysis of Gene Expression Profiles: Class Discovery and Leaf Ordering. Proceedings of the 6th Annual International Conference on Computational Molecular Biology (RECOMB'02) (2002) 127-136
6. Deng L., Ma J., Pei J.: Rank Sum Method for Related Gene Selection and Its Application to Tumor Diagnosis. Chinese Science Bulletin **49** (2004) 1652-1657
7. Luo, J., Ma, J.: A Multi-Population X-2 Test Approach to Informative Gene Selection. Lecture Notes in Computer Science **3578** (2005) 406-413
8. Koller, D., Sahami, M.: Toward Optimal Feature Selection. Proceedings of the 13th International Conference on Machine Learning (ICML'96) (1996) 284-292
9. Xing, E.P., Jordan, M.I., Karp R.M.: Feature Selection for High-Dimensional Genomic Microarray Data. Proceedings of the 18th International Conference of Machine Learning (ICML'01) (2001) 601-608
10. Bo, T., Jonassen, I.: New Feature Subset Selection Procedures for Classification of Expression Profiles. Genome Biology **3** (2002) RESEARCH0017.1–0017.11
11. Wang, L., Ma, J.: A Post-Filtering Gene Selection Algorithm Based on Redundancy and Multi-Gene Analysis. International Journal of Information Technology **11** (2005) 36-44

12. Ma, J., Wang, T.: A cost-Function Approach to Rival Penalized Competitive Learning (RPCL). IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics **36** (2006) 722-737
13. Xu, L., Krzyzak, A., Oja, E.: Rival Penalized Competitive Learning for Clustering Analysis, RBF Net, and Curve Detection. IEEE Transactions on Neural Networks **4** (1993) 636-649
14. Hsu, C., Chang, C., Lin, C.: A Practical Guide to Support Vector Classification. National Taiwan University. Department of Computer Science and Information Engineering, given in the web: www.csie.ntu.edu.tw/∼cjlin/papers/guide/guide.pdf.
15. Ben-Dor, A., Friedman, N., Yakhini, Z.: Scoring Genes for Relevance. Agilent Technical Report no.AGL-2000-13 (2000)