THE EFFECT OF NOISY BOOTSTRAPPING ON THE ROBUSTNESS OF SUPERVISED CLASSIFICATION OF GENE EXPRESSION DATA

Niv Efron and Nathan Intrator
School of Computer Science, Tel-Aviv University, Ramat-Aviv 69978, Israel
{nefron,nin}@post.tau.ac.il

Abstract. This paper discusses the role of noisy bootstrapping in the analysis of microarray data. We apply linear discriminant analysis, according to Fisher's method, to perform feature selection and classification, creating a linear model which enables clinicians easier interpretation of the results. We present the effects of bootstrapping in improvement of the results, and specifically robustifying classification with an increased number of genes.

The performance of our method is demonstrated on publicly available datasets, and a comparison with state of the art published results is included. In particular, we show the effect of the number of features (genes) on the result, as well as the effect of bootstrapping. The results show that our classifier is accurate and quite competitive to other classifiers, although it is simpler, and enables considering a larger set of genes in the classification.

INTRODUCTION

Biotechnologies used for profiling gene expression are advancing rapidly. High density oligonucleotide chips and cDNA microarrays provide us the capability to monitor expression levels of many thousands of genes simultaneously, thus creating large data sets to investigate. Such data exploration can grant insight and understanding of cellular processes and gene functionality. One example is the ability to examine gene variations among tumors. Distinguishing between tumor classes using gene expression values can advance research of cancer classification and assist applying appropriate treatment successfully. This presents a challenging task in supervised learning - extremely high-dimensional data, with only few observations available.

Large dimensionality of the data makes the learning task more complex. It is especially difficult when the number of training examples is very small. This is usually the case when dealing with cDNA microarrays, usually containing

0-7803-8608-6/04/$20.00

several thousands of values in each observation (gene expression values), and only a few dozens of observations (different tissues). This indicates the use of basic supervised learning algorithms, since complex learning machines tend to over-fit such training data.

In this paper, we present a classification mechanism, based on two simple techniques - noisy bootstrapping and linear discriminant analysis. Our classifier is robust, does not require any complicated tuning and performs well on the data sets we use for validation. We show the use of noisy bootstrap for creating robust multi-gene models, that perform well on test data.

Our model was applied to several highly studied gene expression data sets, that are investigated by many groups. This grants further validation to the competitive results shown by our method.

The data sets used in this paper were introduced in [1, 2, 11], and were later investigated in numerous publications, dealing with classification or clustering of gene array data and describing various mechanisms attempting to tackle the difficulties of such data - graph algorithms [12], support vector machines [10], genetic and evolutionary algorithms. Most of the publications also use a gene selection procedure, such as principle component analysis, partial least squares, or other methods to sort the genes based on the standard deviation of their expression levels. Several groups applied predictor aggregation techniques on the gene data (AdaBoost [4], boosting with decision trees [5] and also bagging for improved clustering [6]).

As will be detailed later, our classification method is based on a simpler mechanism than most of the above citations - linear discriminant analysis. This enables us to present a linear model, whereas most of the other publications produce non-linear models. In particular, we show that a linear model can be used to achieve results that are comparable to the above mentioned non-linear models, as long as the classifier's robustness is maintained (through bootstrapping). Building linear models has the advantage of enabling simple interpretation of the resulting model, and this is one of the reasons biologists and clinicians prefer linear models with no variable transformations.

In contrast to other cases of classification of high dimensional data, interpretation of gene array data requires the use of many genes, some of them surrogate to others. Minimizing the features set may result in concentrating on a small group of very influential genes. Potentially, these are genes that have strong correlation to many biological phenotypes, and thus, they are not very useful when developing pharmaceutic treatments (as they may lead to many side effects). It is most desirable to find genes that have a local and focused influence, relevant to the specific cancer type at question. This is the motivation behind our attempt to find ways to produce good classification results while using relatively large amounts of genes (instead of looking for the smallest set of features sufficient for successful classification). This increases the amount of interesting genes for clinicians to study for drug development with potentially minimal side effects.

Using large amounts of features on small training data sets leads to the problem known as "curse of dimensionality". Particularly, in a linear model,

this leads to a singular within-class scatter matrix (S_W). The standard solution to this problem is regularizing and using $(S_W+\lambda*I)^{-1}$ (ridge-regression). This has the flaw of using the same λ for all variables, and thus, this solution is sub-optimal. The current prevalent remedy is to normalize the data, so that all variables have the same standard deviation. This may lead to a greater problem, since small variability may mean irrelevance and the normalization might actually emphasize the noise. We address the problem by using a variation of the noisy bootstrap [8], which extends the classical bootstrap by adding parametric noise and increasing the number of training patterns. This reduces the variance fluctuations caused by a small number of samples, and enables extracting many parameter models, which perform well on unseen test data.

We demonstrate the novel modelling technique on several heavily studied cancer related microarray experiments, and show that state of the art performance (which was previously achieved by a different method for each data set) can be achieved by our classifier.

METHODS

The general classification framework described in this paper deals with 3 stages in the classification process - noisy bootstrapping, feature selection and training the classifer.

The input to the process contains two groups of samples - training samples (including their classifications) and test samples (which also include classifications - in order to evaluate the performance of the classifier). First, we estimate the noise in the data and extend the training set by re-sampling with simulated noise, thus creating new perturbed samples. This enables us to increase the amount of training samples (the effects of this will be shown later). Next, we perform feature selection. This is done on the extended training set (the classifier's construction must not rely on the test samples which are used to validate it). The final step is to train a classifier on the extended data set, and to test its performance on the test samples.

The next sections detail the feature selection and classification techniques, based on Fisher's linear discriminant analysis (FLDA). Then the bootstrapping method used to extend the training set is described. Results on several data sets are presented in a following section, also containing a comparison with previous results.

The notation used in this paper is as follows: gene expression data on p genes for n observations will be represented in an $n \times p$ matrix $X = (x_{i,j})$, where $x_{i,j}$ represents the expression level of gene j in observation i. The expression levels of a single observation will be noted as x, a $p \times 1$ vector. The target values (class memberships) will be noted as y_i, where $y_i \in 1, 2, ..., K$, where K is the number of classes. The target classes can represent different tumor classes, survival indicators or malignant/normal classifications.

In this paper, the only preprocessing done on the data was standardiza-

tion, i.e., for each gene, the gene expression values were normalized so that their mean across all observations is 0, and the standard deviation is 1.

Feature Selection

The probability of error in a given classification problem is influenced by the input dimensionality (number of features per sample), and the number of training examples. If the distribution of the input space (class-conditional densities) is known, then additional features reduce the error probability. However, in most cases there are no such assumptions on the structure of the input space, and the performance of a classifier starts to deteriorate as the dimensionality grows over a certain point. This *"curse of dimensionality"* is due to the inherent sparseness of a high dimensional input space. This becomes the main factor affecting the classification performance.

In the case of gene arrays, the dimensionality of the data is very large and the number of samples is usually small. It is necessary to choose a small subset of variables that have the largest influence on the target function - the class membership. Many of the genes show little change in the expression levels throughout the different observations. We will look for genes that behave differently in observations from different classes, thus implying that they are among the genes governing the target values.

Fisher's Criterion. In 1936, R.A Fisher proposed a criterion to measure the separation between two groups in a direction w, as the distance between the means of the two groups, divided by the sum of their scatters [9]. Formally, the criterion is defined as $J(w) = \frac{|\tilde{\mu_1} - \tilde{\mu_2}|^2}{(\tilde{s_1}^2 + \tilde{s_2}^2)}$, where $\tilde{\mu_k} = w^t \mu_k$ (μ_i is the mean of group k), and $\tilde{s_k}^2 = \sum_{x \in C_k} (w^t x - \tilde{\mu_k})^2$ is the scatter of group k in the direction w. Larger values of $J(w)$ represent stronger separations.

In order to employ Fisher's criterion for the task of gene choosing, we score each of the genes according to the separation of the classes considering only that gene (note that this is a 1-dimensional case, so w has no effect). That is, if C_k ($k = 1, 2$) represent the two classes, then for each gene j, we calculate $\tilde{\mu_k} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{i,j}$, $\tilde{s_k}^2 = \sum_{i \in C_k} (x_{i,j} - \mu_k)^2$ and $Score(j) = \frac{|\tilde{\mu_1} - \tilde{\mu_2}|^2}{(\tilde{s_1}^2 + \tilde{s_2}^2)}$.

After scoring the genes, we sort them by descending scores and consider (for the classification problem) only the first d ($1 \leq d \leq p$). d can be determined according to testing results (cross-validation, etc.).

This scoring method is suitable for 2-class classification problems. A similar scoring method can be applied on multi-class problems. Each gene will be scored according to the ratio of its *Between-Classes-Sum-of-Squares* to its *Within-Classes-Sum-of-Squares* [7]. For each gene j, we calculate $Score(j)$ as $\frac{BSS(j)}{WSS(j)} = \frac{\sum_{k \in K} \sum_{i \in C_k} (\bar{x}_{k,j} - \bar{x}_{\cdot,j})^2}{\sum_{k \in K} \sum_{i \in C_k} (x_{i,j} - \bar{x}_{k,j})^2}$. Again, the genes are sorted according to their scores, and the d genes with the highest scores are considered in the classification process.

Classification - Fisher's Linear Discriminant Analysis

After the dimensionality is reduced, a training algorithm may be applied on the data. Fisher's criterion implies an optimization problem - finding w such that $J(w)$ is maximal. This means finding a linear combination of the gene expression values $(w \cdot x)$, so that the separation is stronger. Note that this is a multi-dimensional scenario, differently from the 1-dimensional BSS-WSS score, which considered each gene separately. If we examine the definition of Fisher's Criterion - $J(w) = \frac{|\tilde{\mu_1} - \tilde{\mu_2}|^2}{(\tilde{s_1}^2 + \tilde{s_2}^2)}$, then we can obtain $J(w)$ as an explicit function of w. Let us define the scatter matrix for class k as $S_k = \sum_{x \in C_k}(x - \mu_k)(x - \mu_k)^t$, the *Within-Class* scatter matrix as $S_W = \sum_{k=1}^{2} S_k$ and the *Between-Class* scatter matrix as $S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^t$. Now we can express $\tilde{s_k}^2 = w^t S_k w$, $\tilde{s_1}^2 + \tilde{s_2}^2 = w^t S_W w$ and $(\tilde{\mu_1} - \tilde{\mu_2})^2 = w^t S_B w$. Thus, the Fisher criterion can be expressed as $J(w) = \frac{w^t S_B w}{w^t S_W w}$, and the w that maximizes $J(w)$ is $w^\star = S_W^{-1}(\mu_1 - \mu_2)$.

In order to use the Fisher criterion for 2-class classification problems we need to calculate $w^\star$ from the data in the training set. Having $w^\star$, we calculate $y = w^\star \cdot x$ for each new test sample x. According to the sign of y, we determine if x belongs to class 1 or class 2.

An extension of Fisher's Linear Discriminant enables us to perform classifications in problems where the data is divided between $K \geq 2$ classes (this elaboration is taken from [7]). Let us extend the definitions of $S_W = \sum_{k=1}^{K} S_k$ and $S_B = \sum_{k=1}^{K} n_k(\tilde{\mu_k} - \tilde{\mu})(\tilde{\mu_k} - \tilde{\mu})^t$, where n_k is the number of observations from class C_k, $\tilde{\mu_k}$ is the mean of class C_k, and $\tilde{\mu}$ is the mean of all the data.

The extreme values of $\frac{w^t S_B w}{w^t S_W w}$ occur at the w which are the eigenvectors of $S_W^{-1} S_B$. There are at most $s \leq \min(K-1, p)$ non-zero eigenvalues $\lambda_1, \lambda_2, ..., \lambda_s$, with corresponding linearly independent eigenvectors $v_1, v_2, ...v_s$. After the eigenvectors are calculated, a classification of an observation x can be made. Let $d_k(x) = \sum_{l=1}^{s}((x - \bar{x_k}) \cdot v_l)^2$ denote the (squared) Euclidean distance (in terms of the discriminant variables) of x from $\bar{x_k}$ (the mean vector of the k-th class in the training set). The predicted classification of x will be $C(x) = \arg\min_k d_k(x)$, that is the class whose mean vector is closest to x in the space of the discriminant variables.

In order to deal with cases where the inversion of the matrix S_W is inaccurate, we can regularize it, by adding a certain percentile of its eigenvalues to its diagonal. Formally, if $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_s$ are the eigenvalues of S_W , then we regularize by using $S_W^{reg} = S_W + prctile(< \lambda_1, ..., \lambda_s >, \alpha) \cdot I$, $(0 \leq \alpha \leq 1)$. We will refer to this regularized version of FLDA as λ-FLDA.

Bootstrapping

The standard bootstrapping process [8], is based on re-sampling of the training data (with replacement), thus creating several training sets (usually the same size as the original training set). This enables performing the training process on these pseudo-sets and creating an ensemble of predictors, whose

results are then aggregated. This is common when dealing with data sets with a small amount of observations, since the ensemble of classifiers creates a more robust classification model. Gene array data sets usually contain a small number of high dimensional observations. This implies that the use of re-sampling techniques might prove useful in supervised and unsupervised learning problems with such data (examples can be seen in [4, 5, 6]).

Noisy Bootstrap. Noisy bootstrapping is based on re-sampling the data, and adding noise to the samples. Noise is simulated according to an estimated parametric model built from the training data. From these perturbed samples, bootstrap training sets are created (again, same size as the original training set) and several classifiers are built and aggregated in order to complete the training process.

In this paper, we present a variation on the noisy bootstrap. Instead of creating several training sets (with size equal to the original) and aggregating several classifiers trained on them, we create one large sample set on which a single classifier is trained. This seems more appropriate in our case, since the small original sample size can prevent a robust solution of the LDA process (due to the singularity of the scatter matrix).

As a parametric model of the noise in each class in the data, we assume a multivariate Gaussian distribution with mean 0, and a diagonal covariance matrix. For each class k, we calculate the standard deviations of the genes, observed in samples with $y_i = k$, and use them as the diagonal of the covariance matrix of the assumed distribution for class k. We chose a diagonal covariance matrix since the small amount of samples prevents proper estimation of the correlation between different genes. Note that only the noise is estimated by a parametric model, while the samples are taken from the actual training data. After the model is built, new observations are generated according to it and a larger data set (which includes the original training set) is created. We refer to the ratio between the amount of bootstrap observations and the number of observations in the original training set as the **bootstrap ratio**.

RESULTS

We examined our classifier on three data sets, which are described in Table 1. The results shown here were achieved using a *Leave-One-Out cross validation* process (LOOCV). Figure 1 shows the results.

An important observation is clear from the results. As we increase the bootstrap ratio, it enables us to consider more genes in the training process and still maintain good classification success ratios. The extension of the data set makes the training process more robust, enabling it to consider more and more genes. This "stairs effect" is visually apparent in the FLDA results plotted in Figure 1.

Data Set	Lymphoma (DLBL) [1]	Colon (I2000) [2]	Leukemia (ALL-AML) [11]
# of genes	4026	2000	7129
# of samples	62	62	72
classes	3 cancer types (11,9,42)	2 classes (40 tumor, 22 normal)	2 cancer types (47,25)

TABLE 1: DATA SETS DESCRIPTION - SHOWS THE NUMBER OF FEATURES (GENES), THE NUMBER OF SAMPLES AND THEIR DIVISION INTO DIFFERENT CLASSES IN EACH OF THE DATA SETS.

Validation of the Results

The datasets used in this paper were previously examined in numerous studies. In order to test the performance of our classifier, we compare our results to several published LOOCV results on these datasets. Table 2 presents the comparisons (we only noted the optimal results for each data set). Our classifier, using FLDA and noisy bootstrap, performed well with respect to the others. On the Lymphoma data, our classifier correctly classified all the samples. On the colon data, our results were 56 correct classifications out of 62, equal to the results of [12] and [10]. Others report lower success ratios. On the Leukemia data, we misclassified only 1 sample out of 72, which is equal to the results of [12], and a little better than the others.

The Leukemia data set [11] was originally divided to a training set (38 obs.) and a test set (34 obs.). This enables another comparison besides the LOOCV results. We repeated the experiment with our classifier 10 times,

Ref.	Method	Percent		
		correct	incorrect	unclassified
Lymphoma (DLBL)				
-	FLDA, Noisy Bootstrap	**100.0**	0.0	0.0
[5]	LogitBoost (all genes)	92.0	8.0	0.0
	LogitBoost (50 genes)	98.4	1.6	0.0
Colon (I2000)				
-	FLDA, Noisy Bootstrap	**90.3**	9.7	0.0
[4]	Clustering (CAST)	88.7	11.3	0.0
	Nearest-Neighbor	80.6	19.4	0.0
	SVM, linear kernel	77.4	12.9	9.7
	SVM, quad. kernel	74.2	14.5	11.3
	AdaBoost, 100 iter.	72.6	17.7	9.7
[12]	CLICK (all genes)	85.5	9.7	4.8
	CLICK (50 genes)	**90.3**	9.7	0.0
[5]	LogitBoost (all genes)	87.1	12.9	0.0
	LogitBoost (50 genes)	83.9	16.1	0.0
[10]	SVM	**90.3**	9.7	0.0
[3]	MAVE (50,100,200 genes)	83.9	16.1	0.0
Leukemia (ALL-AML)				
-	FLDA, Noisy Bootstrap	**98.6**	1.4	0.0
[4]	Nearest-Neighbor	91.6	8.4	0.0
	SVM, linear kernel	93.0	1.4	5.6
	SVM, quad. kernel	94.4	1.4	4.4
	AdaBoost, 100 iter.	95.8	2.8	1.4
[12]	CLICK (all genes)	90.3	4.2	5.5
	CLICK (50 genes)	**98.6**	1.4	0.0
[5]	LogitBoost (all genes)	97.2	2.8	0.0
	LogitBoost (50 genes)	95.8	4.2	0.0

TABLE 2: OTHER LOOCV RESULTS ON THE DATA SETS, TAKEN FROM SEVERAL PUBLICATIONS.

Method	Correct Classifications
FLDA + Noisy bootstrap	**33.6 ± 0.5 / 34**
[11]	29/34
[10]	30-32/34
[13]	32/34
[5]	33/34
[3]	33/34

TABLE 3: TEST SET ERROR RATES ON THE ALL-AML DATA SET. EACH ROW RECORDS THE AMOUNT OF TEST SAMPLES CORRECTLY CLASSIFIED BY THE CORRESPONDING METHOD (OUT OF THE 34 TEST SAMPLES).

using several values for bootstrap ratios and for d. In 6 of the 10 experiments, perfect classification was achieved (34 from 34), and in the other 4 experiments, 33 out of the 34 test samples were classified correctly. Table 3 presents the test set error rates from different published experiments compared with the results of our method. It can be seen that our classifier achieved good results on this test set as well, and that no other method managed to classify all 34 test samples correctly.

CONCLUSIONS

Interpretation of microarray data is becoming essential in the search for new drugs. Supervised learning algorithms can speed up such interpretation and assist in focusing on specific genes. In this work we presented a model that relies on a large number of genes for prediction, yet is still interpretable for clinicians. The motivation behind that is to enable clinical interpretation of the results, so that a clinician would have a larger collection of potential genes to choose from (for drug discovery, etc.). Thus, more target-specific genes that have the desired therapeutic effect can be chosen and the amount of side effects will be minimized.

Robust classification has to deal with the "curse of dimensionality", which results from prediction of high-dimensional data with few observations. We tackled this problem by using a linear model (this was also essential for practical interpretability of the results) and by using the noisy bootstrap, which addressed the singularity of the scatter matrix in a form that is more local to each gene (rather than only adding a single constant to the diagonal, as is done for example in ridge regression). The noisy bootstrap enables extending the number of chosen genes while maintaining, and in effect improving, robust classification performance.

Our classification process represents a basic learning machine with reasonable capacity. This prevents over-fitting to the training set, which is important when dealing with high-dimensional problems with a small number of samples. Our results show high success ratios, and the robustness of the process is demonstrated by their stability with respect to the number of participating genes. We have demonstrated state of the art performance on the DLBL, I2000 and ALL-AML data sets ([1, 2, 11] respectively). The three competing models (SVM [10], CLICK [12] and LogitBoost [5]) showed

stronger sensitivity to the number of features used for classification.

Due to its simplicity, we expect our classifier to perform well on other microarray data sets, and to demonstrate high generalization capabilities.

REFERENCES

[1] A. Alizadeh, M. Eisen, R. Davis, C. Ma, I. Lossos, A. Rosenwald, J. Boldrick, H. Sabet, T. Tran, X. Yu, J. Powell, L. Yang, G. Marti, T. Moore, J. H. Jr., L. Lu, D. Lewis, R. Tibshirani, G. Sherlock, W. Chan, T. Greiner, D. Weisenburger, J. Armitage, R. Warnke, R. Levy, W. Wilson, M. Grever, J. Byrd, D. Botstein, P. Brown and L. Staudt, "Different types of diffuse large b-bell lymphoma identified by gene expression profiling," **Nature**, vol. 403, pp. 503–511, 2000.

[2] U. Alon, N. Barkai, D. Notterman, K. Gish, s. Ybarra, D. Mack and A. Levine, "Broad patterns of gene expressions revealed by clustering analysis of tumor and normal colon tissus probed by oligonucleotide arrays," **Proc. Natl. Acad. Sci.**, vol. 96, pp. 6745–6750, 1999.

[3] A. Antoniadis, S. Lambert-Lacroix and F. Leblanc, "Effective dimension reduction methods for tumor classification using gene expression data," **Bioinformatics**, vol. 19, pp. 563–570, 2003.

[4] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer and Z. Yakhini, "Tissue classification with gene expression profiles," **Journal of Computational Biology**, vol. 7, pp. 559–584, 2000.

[5] M. Dettling and P. Bühlmann, "Boosting for tumor classification with gene expression data," **Bioinformatics**, vol. 19, pp. 1061–1069, 2003.

[6] S. Dudoit and J. Fridlyand, "Bagging to improve the accuracy of a clustering procedure," **Bioinformatics**, vol. 19, pp. 1090–1099, 2003.

[7] S. Dudoit, J. Fridlyand and T. Speed, "Comparison of discrimination methods for the classification of tumor using gene expression data," Techn. Report 576, **Department of Statistics, University of California, Berkley**, 2000.

[8] B. Efron and R. Tibshirani, **An introduction to the bootstrap**, New York: Chapman and Hall, 1993.

[9] R. Fisher, "The use of multiple measurements in taxonomic problems," **Annals of Eugenics**, vol. 7, pp. 179–188, 1936.

[10] T. Furey, N. Cristianini, N. Duffy, D. 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.

[11] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield and E. Lander, "Molecular classification of cancer: class dicovery and class prediction by gene expression monitoring," **Science**, vol. 286, pp. 531–537, 1999.

[12] R. Sharan, R. Elkon and R. Shamir, "Cluster analysis and its applications to gene expression data," in H. Mewes, H. Seidel and B. Weiss (eds.), **Bioinformatics and Genome Analysis**, Springer, pp. 83–108, 2002.

[13] R. Tibshirani, T. Hastie, B. Narasimhan and G. Chu, "Diagnosis of multiple cancer types by shrunken centroids of gene expression," **PNAS**, vol. 99, no. 10, pp. 6567–6572, 2002.

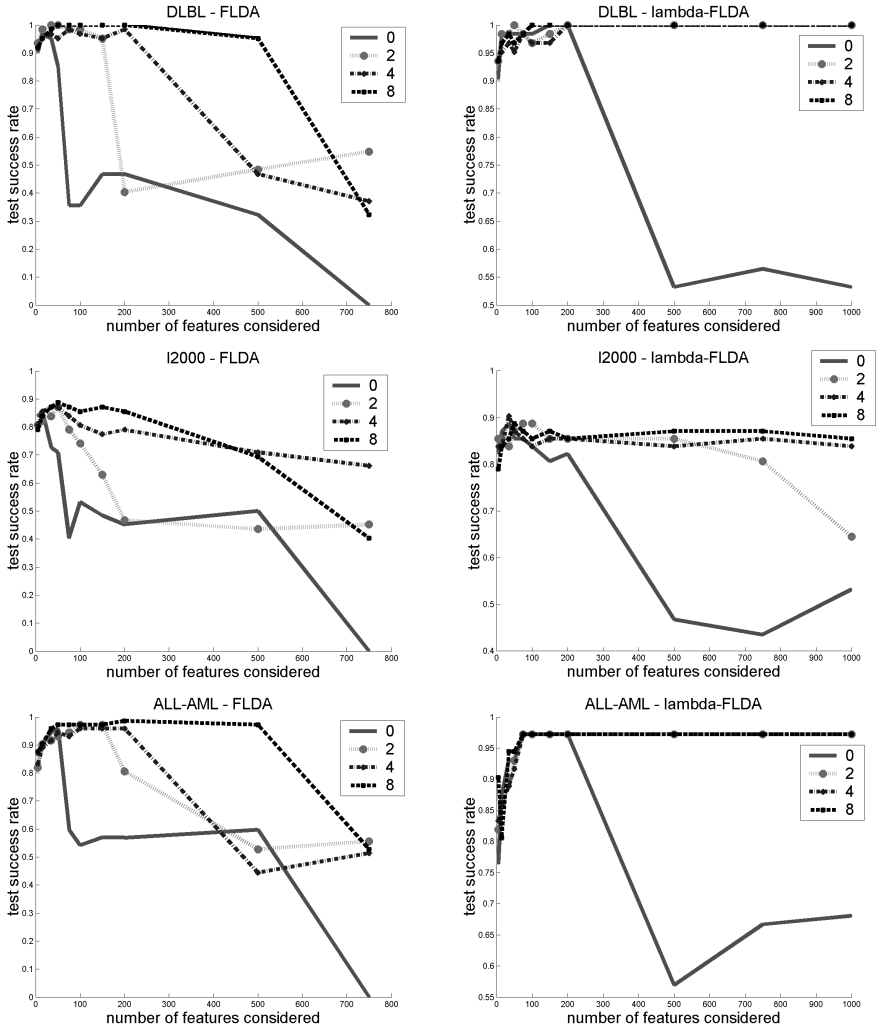

Figure 1: results - LOOCV success ratios vs. number of genes considered, using **FLDA** and λ**-FLDA**. The different lines represent different bootstrap ratios. Each row includes results on a different data set - Lymphoma (DLBL), Colon (I2000) and Leukemia (ALL-AML). For the λ**-FLDA**, a value of $\alpha = 0.75$ was taken.