

MARGIN-BASED FEATURE SELECTION FILTERS FOR MICROARRAY GENE EXPRESSION DATA.

Włodzisław DUCH¹, Jacek BIESIADA²

May 24, 2006

¹ *Department of Informatics, Nicolaus Copernicus University, Toruń, Poland
School of Computer Engineering, Nanyang Technological University, Singapore
E-mail: wduch@is.umk.pl, or Google: Duch*

³ *Division of Comp. Methods, Dept. of Electrotechnology, The Silesian University of
Technology, Katowice, Poland
E-mail: Jacek.Biesiada@polsl.pl*

Abstract

Information selection filters use various relevancy criteria, such as Bayesian consistency, correlation coefficient or mutual information, to determine usefulness of features. Several new ranking indices are introduced. Instead of using all vectors to calculate ranking index margins excluding vectors from strongly overlapping regions are used, sacrificing training accuracy for generalization in ranking of features. This technique is especially useful for microarray gene expression data, where the number of features is very large and the number of samples is very small. Feature selection for three such datasets shows that a relatively small number of genes give the best performance.

Keywords: feature selection, information filters, bioinformatics, gene expression data

1 Introduction.

Biological and medical experiments are frequently very costly and therefore the number of data samples available for analysis is very small. On the other hand information gathered about each case may be very rich. Typical examples analyzed in this paper have less than 100 cases with thousands or tens of thousands features. These features represent intensities of microarray cells interacting with biological tissue, measuring the activity (expression) of particular genes. With such small samples and huge features spaces there is an infinitely many ways to fit the data correctly, and exhaustive searching for good data models may discover many models that by pure chance also fit all known data correctly. Therefore reference models based on the simplest possible description of the data are needed, using only the most robust features and rules (decisions) with significant support. In the recent study of many medical and technical data [1] small sets of logical rules proved to be more accurate than all sophisticated classifiers. Logical descriptions are also highly informative, using only a few important features and providing understandable description of the data.

It is doubtful that analysis of very small datasets using sophisticated methods will have much value. Before sophisticated neural or statistical models are applied to bioinformatics data simplest rule based methods should be tried first. The rule-based description makes overfitting easy to control and will be especially important for bioinformatics data. Bayesian rules defined for each feature may be a good measure of their relevancy. With a large number of features some conditional probability distributions $p(X_k|C)$ of feature values may by chance be separable, while others may accidentally have unusual concentration of samples in the tail of their distribution. In the SVM linear discrimination classification margins are used to increase generalization. Margins may be used in many ways in feature selection. In the next section theoretical framework is presented, while section three contains results of experiments performed on several bioinformatics databases.

2 Theoretical framework.

An information filter [2] is defined by the relevancy coefficient $J(f)$ which gives a measure of dependency between features (f) and classes (C), and is computed for each feature $f \in \mathcal{F}$ individually. Pearson's linear correlation coefficient is probably the simplest such index and therefore should always be used as a reference. For feature X with values x and classes C with values c , where X, C are treated as random variables, correlation coefficient is defined as [3]:

$$\varrho(X, C) = \frac{E(XC) - E(X)E(C)}{\sqrt{\sigma^2(X)\sigma^2(C)}} \quad (1)$$

$\varrho(X, C)$ is equal to ± 1 if X and C are linearly dependent and zero if they are completely uncorrelated. The simplest test estimating significance of the differences in $\varrho(X, C)$ values is based on the probability that two variables are correlated [3]:

$$\mathcal{P}(X \sim C) = \text{erf}\left(|\varrho(X, C)|\sqrt{n/2}\right), \quad (2)$$

where erf is the error function. The feature list ordered by decreasing values of the $\mathcal{P}(X \sim C)$ provides feature ranking.

Linear correlation does not work correctly if the relation between class labels and feature values is not monotonic. A very simple ranking index I_s that works well also for non-monotonic relations is defined as follows. Order the feature values in non-decreasing sequence $x_i \leq x_{i+1}$, add +1 for each $C_i = C_{i+1}$ case, and subtract -1 if $C_i \neq C_{i+1}$; if there is a mixed group with k_1 class C_1 and $k_2 \geq k_1$ class C_2 cases with identical feature values $x_i = x_{i+k}$, $k = k_1 + k_2$, treat it as the worst case, subtracting $3k_1 - k_2 + 1$. This index is very easy to compute and may be modified in various ways, for example adding distance-dependent contributions, but here it will be used in its simplest form.

Information theory is frequently used to define relevance indices. The joint Shannon information is:

$$H(X, C) = - \sum_{i,j} \mathcal{P}(x_i, c_j) \log \mathcal{P}(x_i, c_j) \quad (3)$$

Mutual Information (MI) is the basic quantity used for information filtering:

$$MI(X, C) = H(X) + H(C) - H(X, C) \quad (4)$$

Symmetrical Uncertainty Coefficient (SU) has similar properties to mutual information:

$$SU(X, C) = 2MI(X, C) / (H(X) + H(C)) \quad (5)$$

Estimation of probabilities for small number of data samples is non-trivial. The Parzen window density estimate of a continuous feature X can be used to approximate the probability density $p(x)$ of a distribution [4], where x is a value of feature X . It involves a superposition of normalized window function centered on a training samples. Given a set of n feature values $X = \{x_1, x_2, \dots, x_n\}$, the pdf estimate using Parzen window is given by:

$$\hat{p}(x) = \frac{1}{n} \sum_{i=1}^n \phi(x - x_i, h) \quad (6)$$

where $\phi(\cdot)$ is the window function and h is the window width parameter. Using Gaussian window functions $\phi(x; \sigma)$ the $\hat{p}(\cdot)$ estimate converges to the true density [4]; here σ is the standard deviation. Several values of $\sigma^2 = 0.3, 0.6$, and 0.9 were used in the tests reported below, but because results were not too sensitive to this value only $\sigma = \sqrt{0.3}$ results are reported below.

Decision trees offer another approach to calculate useful relevancy indices. A continuous feature X_i is split using a test $X_i < t$, in effect changing it into a logical variable $z = \text{True}(X_i < t)$, with $z = \text{True}$ or $z = \text{False}$ values. This is equivalent to a one-dimensional, single condition logical rule, predicting class C_1 if z is true, otherwise the class is C_2 .

Given a two class problem and a single feature optimal decisions should be based on the Bayesian Classifier (BC) using the maximum *a posteriori* probability: if $x = x_0$ then for $\mathcal{P}(C_1, x_0) > \mathcal{P}(C_2, x_0)$ class C_1 should always be selected, giving a larger fraction $\mathcal{P}(C_1, x_0)$ of correct predictions, and smaller fraction $\mathcal{P}(C_2, x_0)$ of errors. Bayes error is given by the average accuracy of the Bayesian Classifier (BC) using MAP, or ‘‘informed majority classifier’’ using a single feature X is:

$$BC(X, C) = \sum_i \max_j \mathcal{P}(x_i, c_j) = \sum_i \max_j \mathcal{P}(x_i | c_j) \mathcal{P}(c_j). \quad (7)$$

This index has better justification than the information theoretic indices, but it is unfortunately sensitive to accuracy of probability estimation and does not converge so quickly to the correct values as non-linear indices [2].

The *a priori* probabilities $\mathcal{P}(C)$ are fixed, but $\mathcal{P}(z|t)$ are a function of the threshold, and the joint probabilities $\mathcal{P}(C, z|t)$ also depend on the threshold. These probabilities may be used to calculate mutual information, SU coefficients or just maximize accuracy in BC . Small statistical sample effects make the values of the thresholds t inaccurate, thus contributing to the large uncertainty of the relevance indices and uncertainty of the final ranking [2]. Suppose that 10.000 features are generated, sampling from two partially overlapping Gaussians with $\mathcal{P}(C_1)N$

and $\mathcal{P}(C_2)N$ points, with $N = 70$ and $\mathcal{P}(C_1) = 2/3, \mathcal{P}(C_2) = 1/3$. In this case ranking indices for all features should not significantly differ (and redundancy should be high), with the largest contribution to variance coming from the overlapping region.

A margin around $X_i \approx t$ helps to select only those vectors that belong to class C_k with high confidence, $X_i < t - s \wedge X_i > t + s$, where $2s$ is the margin size. It plays similar role as the margin in SVM method, where a user-define parameter is also introduced. Reliability of the relevance indices selected using vectors that are outside of the margin region should be higher, and thus rankings should be more stable when crossvalidation is used to estimate classification accuracy.

There may be many ways to introduce margins in feature selection; perhaps the simplest is through discretization. Calculation of Bayesian and information theoretical ranking indices requires estimation of probabilities. For the SU index the data has been standardized and Parzen windows technique may be used to calculate reliable estimate of the index value. However, as pointed out in [10] discretization always gave better results with indices based on information theory. The simplest unsupervised discretization introduces three states corresponding to the over-expressions, baseline, and under-expression of genes, removing some noise from the data. For each variable representing gene expression the mean μ and standard deviation σ is calculated for all data (pooled classes), and any value larger than $\mu + \frac{\sigma}{2}$ is replaced by 1, any value smaller than $\mu - \frac{\sigma}{2}$ by -1 and the remaining data are replaced by 0.

Although this discretization is very simple it is quite effective [10]. Indices based on this 3-bin discretization are called BC_3 and SU_3 , and if the interval of the size σ is removed around the mean the index is called BC_2 (calculation of SU_2 has not been done). BC index may also be calculated directly from binary discretization, finding the best treshold $t = (x_{i+1} + x_i)/2$ that maximizes BC index.

A new index that implements a “soft margin” idea has also been added:

$$I_\sigma = \int \min(\mathcal{P}_1(x, \sigma), \mathcal{P}_2(x, \sigma)) dx \quad (8)$$

where $\mathcal{P}_1(x, \sigma), \mathcal{P}_2(x, \sigma)$ are probability distributions estimated by Gaussian Parzen windows technique for class 1 and 2 data, respectively. I_σ measures the area under the maximum of the two class-conditional probability distributions. For almost separated probability distributions there is no penalty from the $\mathcal{P}_1(t) = \mathcal{P}_2(t)$ margin region and the integral may reach 1, while for strong overlaps it may decrease to 0.5 for two identical distributions. Unreliable vectors in the margin region have thus a lower contribution than vectors outside of this region. A hard margin may be introduced excluding from the integration $t \pm s$ region. Instead of I_σ an integral over the product of two distributions is an obvious alternative choice.

3 Numerical experiments

Three DNA microarray gene expression datasets are analyzed below (Table 1). The acute leukemia dataset [5] contains bone marrow samples obtained from adult patients before chemotherapy, with 72 acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) cases, originally divided into the training set (27 ALL and 11 AML), and the test set (20 ALL and 14

AML), with 7129 features. Before normalization thresholding (commonly done by replacing values x by $\max(x, 20)$) and log-transformation has been performed on this data. The colon cancer dataset [6] contains 62 samples, with 40 biopsies from tumor and 22 from healthy parts of the colons of the same patients. Expression levels of 2000 genes with highest minimal intensity are provided. Originally the data has been divided into 40 training and 22 test cases. The lymphoma dataset [7] (DLBCL) has expressions from 4026 genes of two types of diffuse large B-cells. There are 47 samples, 24 of them are from “germinal centre B-like” group while 23 are from the “activated B-like” group.

With such small datasets and very large number of features small sample effects are impossible to avoid. Some features may by chance seem to be very important, perhaps even separating the classes (as is the case with gene 4847 on the leukemia training dataset). Using small number of genes is very risky and a larger “profile” of important genes should be used to increase confidence. It makes more sense to use the leave-one-out or crossvalidation evaluation procedure rather than the *ad hoc* division into training and test sets provided in the original papers [5, 6]. Leave-one-out procedure has the advantage of using almost all data for training and has no variance due to data subsampling, while crossvalidation has the advantage of providing average accuracy and standard deviation, giving a better idea about expected accuracy.

Dataset	Leukemia	Colon Cancer	Lymphoma
Source	Golub et al. (1999) [5]	Alon et al. (1999) [6]	Alizadeh et al. (2000) [7]
Total Samples	72	62	47
Class distribution	47 ALL/25 AML	40 Tumor/22 Normal	24 GBCL/23 ABL
#Genes	7129	2000	4026

Table 1. Summary of the DNA microarray gene expression datasets.

Three relevance indices are used here: linear correlation coefficient ρ , which should provide the base rate for other methods [8], the SU coefficient that tends to work better than mutual information, and the I_σ index for several values of sigma. The original gene expression data contains continuous values. These values are used directly to calculate correlation coefficients. Margin filter values I_σ were calculated from the standardized data with $\sigma = \sqrt{0.30}, \sqrt{0.60}, \sqrt{0.90}$, but because differences were rather small (see Tab. 2) results for only the first value are reported below.

3.1 Leukemia

The “neighborhood analysis” method developed in the original paper [5] finds 1100 genes that are correlated with ALL-AML class distinction. Prediction is based on a rather complex method that assigns weights to the most useful 50 genes and then calculates “prediction strengths” (PS) index as a sum of votes with threshold 0.3. Training was done on 38 samples (27 ALL and 11 AML), using the leave-one-out method to set parameters, and testing was done on 34 samples (20 ALL and 14 AML). As a result 36 samples were correctly predicted and for two samples PS was below the critical 0.3 threshold. 29 of 34 test samples had large correct PS (median 0.77). Results do not differ significantly if the number of predictive genes is changed in the range 10-100.

For this data gene X_{4847} (zyxin) with the threshold $t = -0.087$ on standardized data perfectly separates both classes on the training data, but the margin between them is quite small, and 3 errors are made on the test data. Parzen window density estimations lead to substantial overlap. Expression values of two other genes, X_{1926} and X_{2020} , give only a single error, and threshold rules for 14 other genes make only 2 errors.

Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ρ	4847	4196	1834	2288	6041	3252	1882	1745	1829	2121	2020	2111	3320	4366	6919
SU_3	3252	4847	6855	1834	2121	2288	2354	1882	1779	6041	1685	4328	1745	1144	2642
$I_{\sqrt{0.3}}$	4847	3252	2288	1834	4196	3320	1882	6041	2121	1829	1745	2020	1674	6919	2111
$I_{\sqrt{0.6}}$	4847	3252	2288	4196	1882	1834	3320	1745	2121	1829	6041	1674	6919	2111	2020
$I_{\sqrt{0.9}}$	4847	2288	3252	1882	4196	1834	1745	3320	1829	2121	6041	1674	6919	2111	2020
I_s	1834	4847	1882	2354	2288	760	6855	3252	6376	6041	5501	1685	4377	4366	5772
BC	4847	1882	1834	6855	3252	2288	760	6376	6041	1685	4373	2354	1144	4377	2402
BC_2	3252	4847	4196	6201	2335	2288	1882	758	6225	4082	2642	6041	2020	1834	1829
BC_3	3252	6855	2354	2288	6281	4847	4328	4196	2020	1685	1144	804	1928	5833	3320

Table 2. Top 15 features obtained from various rankings for Leukemia.

Zyxin was selected by ρ and all I_σ indices as the most important, with SU_3 and BC_2 ranking it as second, and BC_3 ranking it at position 6 (Tab. 2). This is an important gene and it may seem that BC_3 has ranked it somehow too low, but surprisingly classification results are very similar to BC_2 and BC (Tab. 3). The X_{1926} gene was never selected among the top 15, and X_{2020} has not been selected by SU_3 , and has not made it to the top 10 genes. The reason is that these genes lead to about 10 errors on the test set, thus making the training/test division (and the original results reported in [5]) rather useless. The values on the training set are unfortunately not correlated with the test set results, confirming our conviction that the training/test division for such as small data has little sense. A better evaluation will be provided by crossvalidation or the leave-one-out procedure. It also shows the usefulness of margins that may decrease rankings of such features (many vectors concentrated around the threshold are not counted).

Four popular classifiers have been used with growing subsets of features to evaluate these rankings. The number of leave-one-out errors obtained is given in Table 3. C4.5 (and other decision trees not reported here) do not handle Leukemia data too well, although for 1 or 2 genes results are quite good. Such small number of genes is not sufficient for reliable classification, our goal should rather be to reach stable number of errors with 20-25 genes.

The one Nearest Neighbor classifier is particularly sensitive to feature selection, showing strong oscillations of accuracy with growing number of features. The SVM and Naive Bayes classifiers give results of similar quality, reaching 1-3 errors for many subsets of genes. Linear correlation coefficient does not work well with 1NN, but with NB and SVM classifiers reaches also 3 errors for 20 or more genes. The SU_3 ranking has 1-3 errors, reaching rather stable plateaux of two errors for SVM with 18-25 genes. The I_s works surprisingly well, giving two errors for a wide range of feature subsets. I_σ and different versions of BC show similar performance and it is not possible to say which one is better. For this particular data all these ranking indices lead to similar results.

Unfortunately classification results do not stabilize for larger (20-100) number of features, oscillating between 2-5 errors, as is evident from Figs. 1-4. These oscillations are well within the

Classifier	Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	
C4.5	ρ	4	4	6	6	6	7	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10	11	11
	SU_3	6	4	6	9	9	8	10	12	12	12	12	12	12	12	13	13	13	13	13	13	13	13	13	13	13	13
	I_s	7	6	8	8	8	8	8	10	12	12	12	12	12	12	13	13	13	13	13	13	13	13	13	13	13	13
	I_σ	4	4	4	7	7	7	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	11	11	11	11
	BC	4	8	8	9	12	10	10	12	12	12	12	12	12	13	13	13	13	13	13	13	13	13	13	13	13	13
	BC_2	6	4	4	4	4	5	7	7	7	7	7	7	7	7	9	9	10	10	10	10	10	10	10	10	10	10
	BC_3	6	3	3	5	5	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	10	12	12	12	12
1NN	ρ	7	8	4	3	5	8	7	9	7	6	7	6	6	6	7	7	8	8	8	8	5	5	6	4	5	4
	SU_3	11	8	8	5	8	9	6	6	5	4	4	5	3	2	1	1	1	2	4	4	3	3	3	3	3	3
	I_s	4	4	3	2	2	3	3	6	8	6	6	4	4	3	2	2	2	2	2	2	2	2	2	2	2	2
	I_σ	7	8	8	8	8	5	5	5	5	6	7	6	6	7	7	5	5	6	7	7	6	6	6	6	6	5
	BC	7	5	3	2	5	7	6	7	6	6	6	4	3	3	4	4	5	5	5	4	4	4	4	5	5	5
	BC_2	11	8	10	6	6	6	8	9	9	10	8	9	8	7	7	8	8	9	9	7	7	7	5	5	5	5
	BC_3	11	5	3	4	6	6	5	10	7	8	6	4	4	4	3	3	4	3	3	4	4	5	5	6	7	7
SVM	ρ	11	6	5	5	5	5	5	5	5	4	5	4	4	4	4	5	5	5	3	3	3	3	3	3	3	3
	SU_3	8	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	2	2
	I_s	15	5	6	5	4	4	4	5	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	2
	I_σ	11	5	6	5	6	4	5	5	4	4	4	4	4	5	4	4	4	4	3	3	3	3	2	2	2	3
	BC	11	10	5	5	4	5	5	5	5	5	4	4	4	4	4	4	4	4	4	2	2	2	2	2	3	3
	BC_2	8	5	5	5	5	5	5	5	5	5	5	4	4	5	4	5	3	3	4	4	4	4	4	4	4	4
	BC_3	8	7	5	5	4	4	4	4	2	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2	2
NBC	ρ	5	7	4	3	4	4	3	3	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	SU_3	8	4	2	2	3	2	2	2	2	3	3	3	3	3	2	2	3	3	3	3	3	3	3	3	3	2
	I_s	6	4	2	1	2	3	3	3	2	2	2	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3
	I_σ	5	4	5	4	4	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	BC	5	2	2	1	2	4	4	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2	3	3	3
	BC_2	8	4	7	6	5	5	5	4	4	4	3	4	4	3	3	3	3	3	3	3	3	3	3	3	3	3
	BC_3	8	4	3	4	4	2	2	2	2	3	3	3	2	3	2	1	2	2	2	2	2	2	2	2	2	2

Table 3. Leave-one errors for the Leukemia dataset with up to 25 top-ranked features; $\sigma = \sqrt{0.3}$.

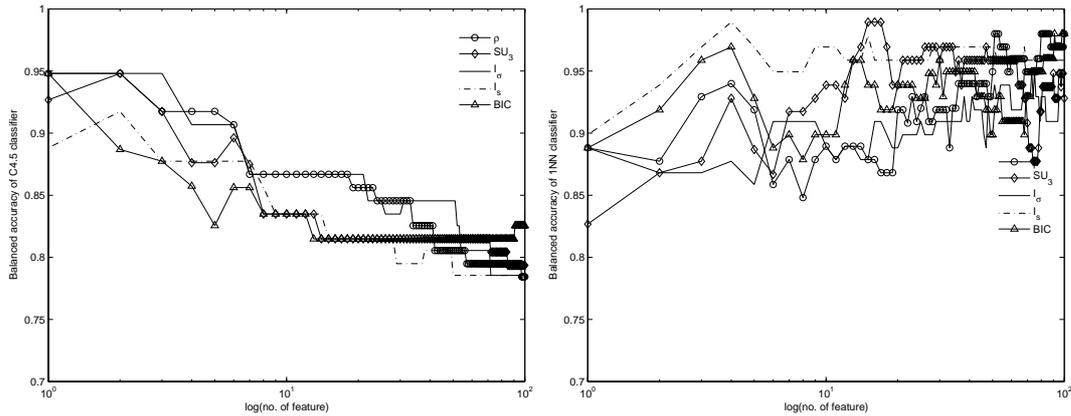


Figure 1. Leave-one-out accuracy for the Leukemia dataset with the C4.5 and the 1NN classifiers.

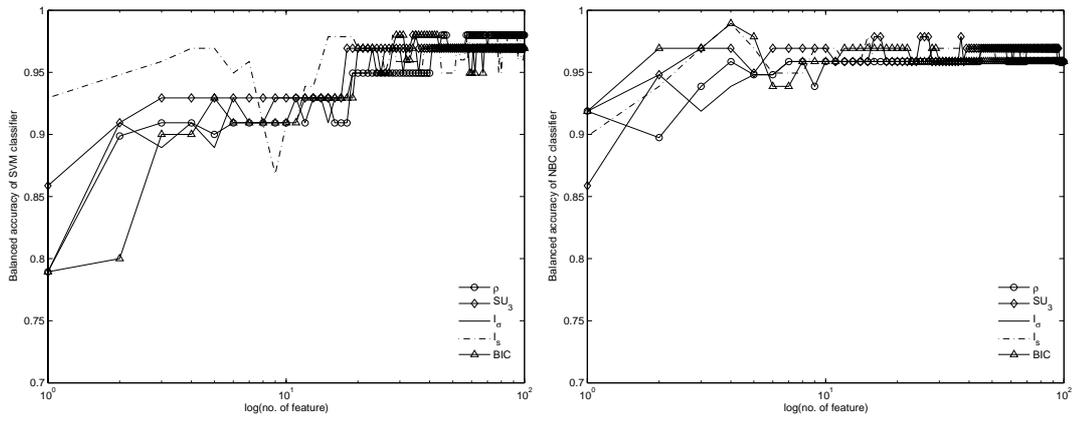


Figure 2. Leave-one-out accuracy for the Leukemia dataset with the SVM and the Naive Bayes (NB) classifiers.

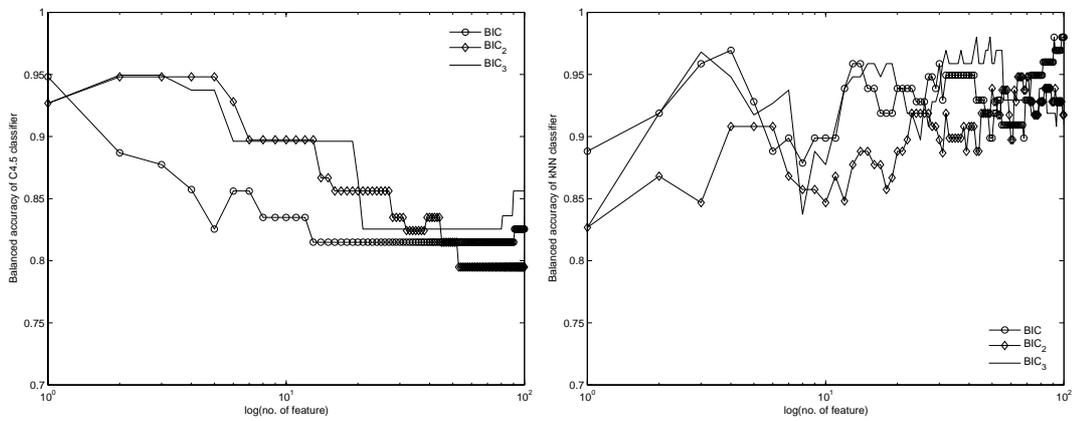


Figure 3. Classification accuracy for the Leukemia dataset using LVO crossvalidation with the C4.5 and INN classifiers and Bayesian ranking.

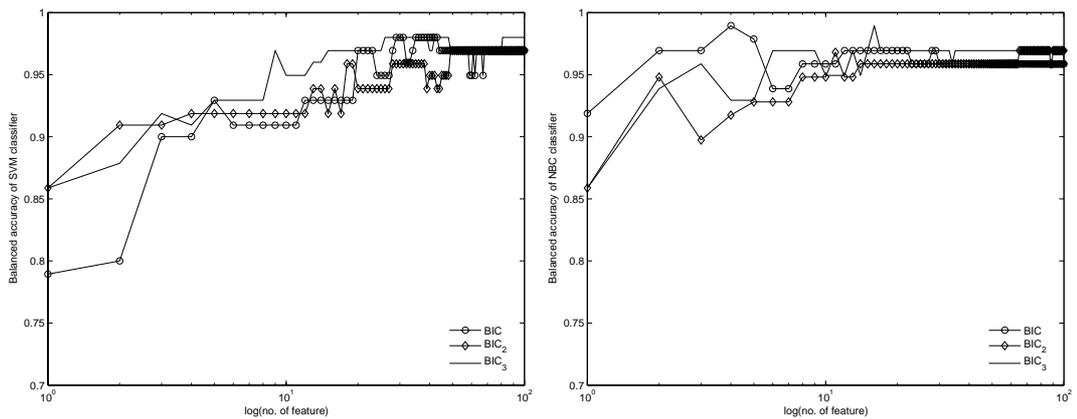


Figure 4. Classification accuracy for the Leukemia dataset using LVO crossvalidation with Bayesian ranking and the SVM and NB classifiers.

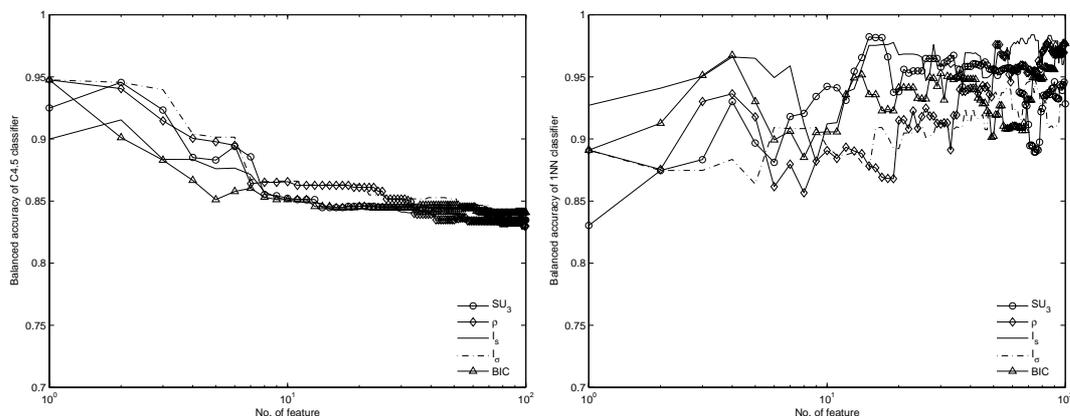


Figure 5. Classification accuracy for the Leukemia dataset using 10-fold crossvalidation for the C4.5 and the 1NN classifiers.

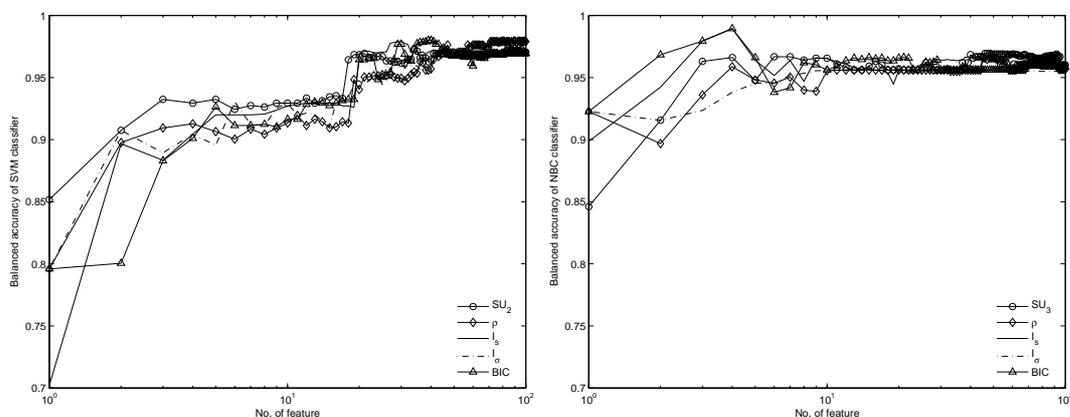


Figure 6. Classification accuracy for the Leukemia dataset using 10-fold crossvalidation for the SBM and NB classifiers.

standard deviation of accuracy estimated by crossvalidation (Figs. 5-6). The leave-one-out and crossvalidation curves are rather similar, therefore only two examples are shown here.

3.2 Colon Tumor

For this dataset ranking results are very different (Tab. 4). Correlation coefficient ρ and soft margin I_σ place genes X_{249} and X_{765} at the top, but do not have X_{513} in the top 15 genes, while SU_3 places it at the 3rd and I_s at the 7th position. Even among BC indices there are significant differences, with gene X_{66} at the 3rd position in BC_2 that is not present in BC , BC_3 , SU_3 and I_s .

Results of classification with different systems show that this time the best leave-one-out results are obtained with the C4.5 tree giving 7 errors for 3-6 genes and dropping to 4 errors for 60-80 genes with I_s ranking (the same result is obtained in 10-fold crossvalidation claulations). SVM and the Naive Bayes approach give 6-9 errors for 30-45 genes. The number of errors tends

Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ρ	249	765	493	1423	245	267	377	822	1892	1772	66	897	1771	1582	780
SU_3	765	1423	513	249	245	267	1582	897	1771	1772	493	1414	780	1671	1060
I_σ	249	765	245	267	1423	1892	493	822	897	415	66	1494	377	1635	1967
I_s	1900	245	625	493	190	657	513	1892	602	576	433	1666	1018	47	1567
BC	1671	249	493	1771	1423	513	267	245	765	625	1772	1042	822	415	1892
BC_2	1423	249	66	286	415	267	245	1387	897	1967	1843	1836	1635	1494	822
BC_3	1423	249	765	267	245	513	1671	415	1892	1582	780	1967	1917	1772	1771

Table 4. Top 15 features obtained from various rankings for Colon Tumor.

Classifier	Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
C4.5	ρ	9	13	13	13	12	12	10	10	10	13	13	13	13	13	14	14	14	14	10	10	10	10	10	10	11
	SU_3	11	11	10	12	12	12	20	21	9	11	12	12	12	10	10	10	10	10	10	10	10	10	11	11	11
	I_s	22	11	13	11	9	9	9	13	13	13	13	13	11	11	11	11	10	10	10	10	10	10	11	11	12
	I_σ	9	13	14	14	12	12	12	13	13	13	13	13	10	10	10	11	11	11	11	11	11	11	14	14	14
	BC	9	9	9	10	7	8	9	10	10	9	9	9	9	10	10	10	9	9	9	9	9	9	9	9	9
	BC_2	11	9	11	11	11	11	11	11	11	11	11	11	11	11	11	11	12	12	12	12	12	12	12	12	12
	BC_3	11	9	12	12	12	11	9	9	10	10	10	10	11	11	11	10	10	10	10	10	10	10	10	10	10
INN	ρ	19	14	13	13	11	13	17	17	15	16	11	12	13	14	12	12	11	11	12	12	14	15	15	13	13
	SU_3	25	12	10	13	12	12	11	13	10	11	12	12	12	12	12	12	9	10	11	10	12	10	11	11	11
	I_s	12	17	16	14	14	11	9	8	9	8	11	9	7	8	8	8	9	10	9	9	10	10	10	12	11
	I_σ	19	14	14	14	15	14	13	16	11	13	11	11	18	16	18	18	16	16	16	16	18	19	19	18	17
	BC	19	12	16	14	14	11	10	13	13	11	11	11	11	9	11	10	11	12	11	10	11	10	14	14	14
	BC_2	17	15	15	12	14	14	13	11	12	14	13	19	18	18	19	13	12	13	12	15	11	12	11	11	13
	BC_3	17	15	16	16	15	12	13	13	12	11	13	15	14	13	13	12	10	9	9	8	10	10	10	10	10
SVM	ρ	14	12	11	11	11	11	10	9	9	9	8	8	8	8	8	7	7	8	8	8	8	8	8	8	8
	SU_3	13	13	11	9	11	11	11	8	8	8	8	8	8	8	8	8	8	8	8	8	8	9	9	7	7
	I_s	22	12	12	11	11	11	11	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	9
	I_σ	14	12	11	11	11	11	11	11	11	10	10	11	10	11	11	11	11	11	11	11	11	11	11	11	11
	BC	22	14	12	10	10	9	10	10	11	11	11	11	10	12	10	10	11	10	8	8	8	8	7	8	8
	BC_2	15	10	10	10	10	10	10	9	9	10	10	11	11	11	9	9	9	9	9	9	9	9	9	9	9
	BC_3	15	10	10	11	11	11	11	11	11	10	10	10	10	10	10	10	9	9	9	9	9	9	9	9	9
NBC	ρ	9	11	10	8	9	10	7	8	9	8	9	9	8	8	8	7	8	7	7	7	7	7	7	8	
	SU_3	11	9	9	8	8	8	8	8	8	8	8	8	8	8	8	7	7	9	10	10	10	11	11	11	
	I_s	24	12	17	11	13	14	12	11	11	12	12	15	15	15	13	14	14	15	16	17	16	16	16	15	15
	I_σ	9	11	10	10	8	8	7	8	8	9	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	BC	26	9	9	8	9	8	9	8	8	8	8	8	9	8	9	8	9	10	8	8	9	8	9	9	9
	BC_2	11	8	10	9	10	11	11	10	10	10	10	10	10	10	11	9	9	9	9	9	9	9	9	9	9
	BC_3	11	8	8	9	8	8	8	9	8	8	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10

Table 5. Leave-one errors for the Colon Tumor dataset with up to 25 top-ranked features; $\sigma = \sqrt{0.3}$.

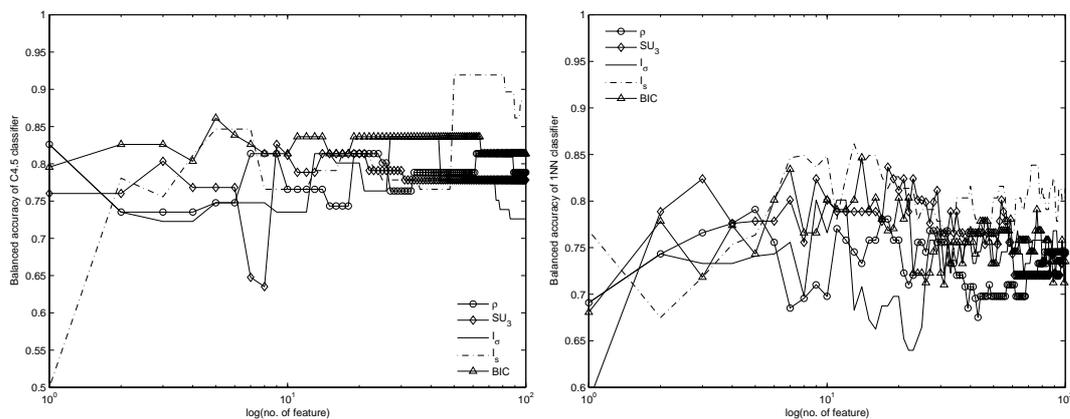


Figure 7. Leave-one-out classification accuracy for the Colon Tumor dataset with the C4.5 and the 1NN classifiers.

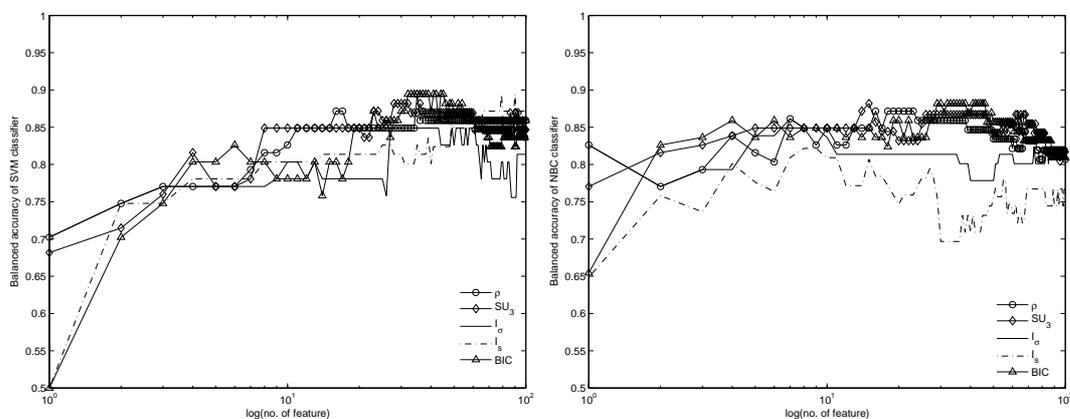


Figure 8. Leave-one-out classification accuracy for the Colon Tumor dataset with the SVM and the NB classifiers.

to oscillate with the increasing number of features, as seen in Figs. 7-8. The leave-one-out and the 10-fold crossvalidation curves have the same character, reaching similar number of errors.

3.3 Lymphoma results

For the Lymphoma dataset top genes are from X_{1275} – X_{1281} range; SU_3 and BC_2 select X_{1281} as the most important, while I_s and BC select X_{1279} (Tab. 6). Some of these genes are probably redundant, but bearing in mind possible errors in measurement of their activity it is better to keep them.

C4.5 has again problems with this data, but the 3 other classifiers achieve error-free leave-one-out results for larger number of genes: 1NN for 60 or more genes with SU_3 ranking (although in crossvalidation tests zero errors are achieved with much smaller number of genes, about 10); SVM and NB for most rankings with 60 or more genes, although in crossvalidation Naive Bayes needs more features. SU_3 index is giving consistently the best results here, with

small fluctuations in the number of errors when the number of genes increases to 100 (Fig. 9-Fig. 10).

Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ρ	1277	1276	1279	1281	1317	1291	1278	1275	75	1280	1316	2439	2417	1315	2438
SU_3	1281	1317	75	2436	1277	1291	1275	1279	2244	3020	1314	1320	3861	1312	1276
I_σ	1276	1279	1281	1277	1278	1280	1317	1291	1275	75	2439	1247	1312	1316	1284
I_s	1279	1276	1314	2244	1264	2496	1317	1316	1281	1278	37	2438	2243	2136	1469
BC	1279	1276	2438	1281	1278	1277	1317	1264	3019	2439	2244	1616	1316	1312	1310
BC_2	1281	1280	1279	1276	75	2439	1275	1278	1277	1267	1312	1284	2496	1144	809
BC_3	2439	1281	1279	1312	1280	1277	1276	1275	1267	75	3860	3085	2467	1320	1317

Table 6. Indices of genes of the first 15 highest-rank features for Lymphoma dataset.

Classifier	Method	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25		
C4.5	ρ	5	5	6	6	7	7	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	9	
	SU_3	7	4	5	6	9	9	10	7	7	7	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	9	
	I_s	4	4	4	5	5	7	7	7	8	8	9	9	9	9	9	9	10	11	11	11	11	11	11	11	11	11	
	I_σ	3	4	4	4	5	6	6	6	7	7	7	7	7	7	7	7	8	8	8	8	8	8	9	9	9	10	
	BC	4	4	6	5	6	6	7	8	8	8	8	8	8	8	9	10	11	11	11	11	11	11	10	11	11	11	9
	BC_2	7	8	4	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	6	7	7	7	7	7	7	
	BC_3	9	7	4	6	6	6	6	6	5	5	5	5	5	6	7	8	8	8	8	8	8	9	9	9	9	9	
1NN	ρ	8	7	7	6	4	3	4	0	0	1	1	1	1	1	1	2	2	2	2	2	2	2	1	2	2	2	
	SU_3	8	5	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	2	2	2	2	2	2	2	
	I_s	6	5	6	2	2	4	4	3	3	4	4	3	4	3	3	3	3	3	2	2	3	3	3	3	3	3	
	I_σ	7	5	4	6	5	6	3	3	5	0	2	2	1	1	2	2	2	2	2	2	2	1	1	1	2	1	
	BC	6	5	5	5	5	5	3	3	4	4	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	
	BC_2	8	9	8	7	2	3	3	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
	BC_3	11	7	5	5	3	5	4	6	5	3	3	2	3	4	3	2	2	2	2	3	3	3	3	2	3	3	
SVM	ρ	6	3	3	3	3	3	3	2	2	2	2	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	
	SU_3	5	3	2	2	1	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	1	1	
	I_s	4	3	3	3	3	4	2	3	4	4	4	2	2	2	2	2	2	2	2	4	2	2	3	1	1	1	
	I_σ	5	3	3	3	3	3	3	2	2	2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
	BC	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	3	3	3	3	1	1	1	1	1	
	BC_2	5	4	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	3	3	2	0	1	1	1	1	1	
	BC_3	10	4	3	3	3	3	3	3	3	3	3	4	2	2	2	3	4	4	2	1	1	1	1	0	0	0	
NBC	ρ	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	3	3	2	2	2	2	2	
	SU_3	5	3	1	1	0	0	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
	I_s	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3	2	2	3	3	3	3	3	3	3	
	I_σ	5	3	4	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2	
	BC	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
	BC_2	5	4	4	3	3	3	4	4	3	4	4	4	4	4	3	3	4	4	4	3	3	2	2	2	2	2	
	BC_3	8	6	4	3	3	3	3	3	3	2	2	2	2	3	3	2	2	2	2	2	2	2	2	2	2	2	

Table 7. Leave-one errors for the Lymphoma dataset with up to 25 top-ranked features; $\sigma = \sqrt{0.3}$.

4 Discussion

Comparison of the leave-one-out error rates achieved with different rankings here with the best results found in literature using various sophisticated feature selection methods is presented in Tab. 8. Although the margin ideas introduced here certainly can be explored in many other

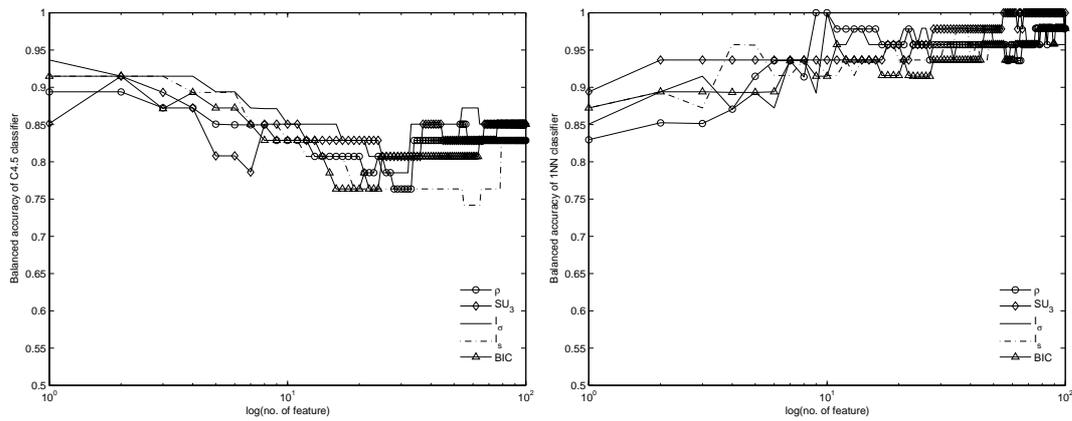


Figure 9. Leave-one-out classification accuracy for the Lymphoma dataset with the C4.5 and 1NN classifiers.

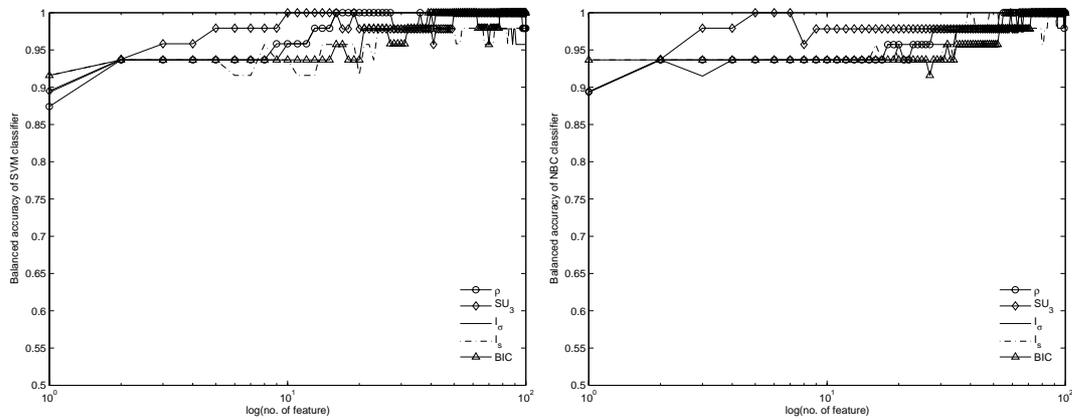


Figure 10. Leave-one-out classification accuracy for the Lymphoma dataset with the SVM and the NB classifiers.

ways results obtained so far are encouraging. For Lymphoma and Leukemia feature subsets leading to zero errors for NBC, 1NN and SVM classifiers have been identified using soft margin (I_σ) index or hard margin indices I_s, SU_3 . Bayesian indices BC , although theoretically well justified, cannot be reliably estimated [2] and have not lead to such good results.

Best leave-one-out results found in the literature for Colon cancer have been replicated only with I_s index, and are better on by 3 errors [10] comparing to the results obtained with SU_3 index. The reference results [10] were found using a selection rather than a ranking method, and removal of redundancy may improve the results. The crossvalidation results show that differences of a few errors are easily within the variance of all classification methods used here.

Sometimes a very small number of features gives the best results; bearing in mind small sample effects for this type of data larger subsets of features should be preferred, even though classification results on a given dataset may be slightly worse. The problem with model selection for these datasets is quite severe, as there is little correlation between results on the training and

Data	Method	NBC	SVM	INN	C4.5	Literature
Lymphoma	I_σ	0.00	0.00	0.00	6.38	
	I_s	0.00	2.22	4.25	8.51	
	SU_3	0.00	0.00	0.00	4.25	0.0, AIC, BIC, MDL [9]
Leukemia	I_σ	2.77	2.77	4.16	5.55	0.0, MRMR [10];
	I_s	1.39	1.39	1.39	9.72	
	SU_3	0.00	0.00	0.00	5.55	1.39, SVM [11]; 1.39, NLProbit [12]
Colon cancer	I_σ	12.90	12.90	17.74	14.51	6.45, MRM [10];
	I_s	17.74	9.67	11.29	6.45	
	SU_3	11.29	11.29	14.51	14.51	6.45, PLS, LD, QDA [13]

Table 8. Comparison of the lowest error rates (in %) for SU and I_σ ranking indices with the best results found in literature (using 100 features).

test partitions in the crossvalidation runs.

The erratic behavior of accuracy as a function of the number of features is a major drawback of all ranking methods, affecting not only the gene expression data, but also many other data with large number of features. In case of Bayesian indices BC and their variants problems with accurate estimation may be responsible for such behavior [2], while in case of other indices this may be the effect of redundancy and small sample size. One way to improve and stabilize the results is to use crossvalidation or bootstrap techniques to calculate cumulative ranking indices. However, tests of this idea did not led to more monotonic dependence of accuracy on the number of selected features. Perhaps a simple removal of redundant features will lead to more stable behavior. Reduction of computational costs may be achieved by ordering features according to their ranking indices, and then expanding the feature set starting from the best one and adding them consecutively, but skipping those features that do not increase accuracy on the training partition in crossvalidation. In addition one may try boosting techniques on individual vectors, adding only the features that contribute to handling errors and do not degrade the quality of correctly classified cases.

Good performance of the I_s index is somehow surprising, bearing in mind that this is a very simple-minded index that can be improved in many ways. For this type of data simplest solutions (discretization and naive ranking) tend to work well and thus it is hard to see the advantage of margin-based filters that perform in a similar way as other ranking indices. More tests on larger datasets should be done and several improvements of the basic margin feature selection idea should be investigated. The position and the size of the margins should be optimized, and other indices to measure overlap of probability distributions should be introduced to model the “soft margin” idea. These ideas will be tested soon.

Acknowledgement: We are grateful for the support by the Polish Committee for Scientific Research, research grant 2005-2007; Jacek Biesiada is also grateful for support by the Foundation for Polish Science.

References

- [1] W. Duch, R. Setiono, and J. Zurada. Computational intelligence methods for understanding of data. *Proceedings of the IEEE*, 92(5):771–805, 2004.

- [2] W. Duch. Filter methods. In I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, editors, *Feature extraction, foundations and applications*, pages 406–411. Physica Verlag, Springer, Berlin, Heidelberg, New York, 2006.
- [3] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge, UK, 1988.
- [4] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. London, U.K. Chapman & Hall, 1986.
- [5] T.R. Golub et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
- [6] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *PNAS*, 96:6745–6750, 1999.
- [7] A.A. Alizadeh et al. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, 2000.
- [8] W. Duch, T. Wiecek, J. Biesiada, M. Blachnik, Comparison of feature ranking methods based on information entropy. Proc. of International Joint Conference on Neural Networks (IJCNN), Budapest 2004, IEEE Press, pp. 1415-1420
- [9] Xiaobo Zhou, Xiaodong Wang, and Edward R. Dougherty. Gene selection using logistic regressions based on AIC, BIC and MDL criteria. *New Mathematics and Natural Computation*, 1(1):129–145, 2005.
- [10] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 3(2):185–205, 2005.
- [11] Y. Lee and C-K. Lee. Classification of multiple cancer types by multicategory support vector machines using gene expression data. *Bioinformatics*, 19:1132–1139, 2003.
- [12] X. Zhou, X. Wang, and E.R. Dougherty. Nonlinear probit gene classification using mutual information and wavelet-based feature selection. *Biological Systems*, 12(3):371–386, 2004.
- [13] D.V. Nguyen and D.M. Rocke. Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*, 18:39–50, 2002.