A Kolmogorov-Smirnov Correlation-Based Filter for Microarray Data

Jacek Biesiada[1] and Włodzisław Duch[2]

[1] Division of Computer Methods, Department of Electrotechnology, The Silesian University of Technology, ul. Krasińskiego 8, 40-019 Katowice, Poland
Jacek.Biesiada@polsl.pl
Division of Biomedical Informatics, Cincinnati Children Hosptial Medical Center, 3333 Burnet Ave, Cincinnati, Ohio 45229-3039, USA
[2] Department of Informatics, Nicolaus Copernicus University, Grudziądzka 5, Toruń, Poland
Google: Duch

Abstract. A filter algorithm using F-measure has been used with feature redundancy removal based on the Kolmogorov-Smirnov (KS) test for rough equality of statistical distributions. As a result computationally efficient K-S Correlation-Based Selection algorithm has been developed and tested on three high-dimensional microarray datasets using four types of classifiers. Results are quite encouraging and several improvements are suggested.

1 Introduction

Feature ranking and feature selection algorithms applicable to large data mining problems with very high number of features that are potentially irrelevant for a given task are usually of the filter type [1]. Filter algorithms remove features that have no chance to be useful in further data analysis, independently of particular predictive system (predictor) that may be used on this data. In the simplest case feature filter is a function returning a relevance index $J(\mathcal{S}|\mathcal{D}, C)$ that estimates, given the data $\mathcal{D}$, how relevant a given feature subset $\mathcal{S}$ is for the task C (usually classification, association or approximation of data). Since the data and the task are usually fixed and only the subsets $\mathcal{S}$ vary, the relevance index will be written as $J(\mathcal{S})$. This index may result from a simple calculation of a correlation coefficient or entropy-based index, or it may be computed using more involved algorithmic procedures (for example, requiring creation of partial decision tree, or finding nearest neighbors of some vectors). For large problems simpler indices have an obvious advantage of being easier to calculate, requiring an effort on the order of $O(n)$, while more sophisticated procedures based on distances may require $O(n^2)$ operations.

Relevance indices may be computed for individual features $X_i, i = 1 \ldots N$, providing indices that establish a ranking order $J(X_{i_1}) \leq J(X_{i_2}) \cdots \leq J(X_{i_N})$. Those features which have the lowest ranks are subsequently filtered out. For independent features this may be sufficient, but if features are correlated many of them may be redundant. Ranking does not guarantee that a small subset of important features will be found. In pathological situations a single best feature may not even be a member of the best pair of features [2]. Adding many redundant features may create instable behavior

M. Ishikawa et al. (Eds.): ICONIP 2007, Part II, LNCS 4985, pp. 285–294, 2008.

of some predictive algorithms, with chaotic changes of results for a growing number of features. This is a major problem especially for small sample data with very large dimensionality, but has been also observed with large datasets [3]. However, methods that search for the best subset of features may first use filters to remove irrelevant features and then use the same ranking indices on different subsets of features to evaluate their usefulness.

Despite these potential problems in practical applications filter methods for ranking are widely used and frequently give quite good results. There is little empirical experience in matching filters with predictive systems. Perhaps different types of filters could be matched with different types of predictors, but so far no theoretical arguments or strong empirical evidence has been given to support such claim. The value of the relevance index should be positively correlated with accuracy of any reasonable predictor trained for a given task C on the data $\mathcal{D}$ using the feature subset $\mathcal{S}$.

Although filter methods do not depend directly on the predictors obviously the cutoff threshold for relevance index to reject features may either be set arbitrarily at some level, or by evaluation of feature contributions by the predictor. Features are ranked by the filter, but how many best features are finally taken is determined using the predictor. This approach may be called “filtrapper” or “frapper” [1], and it is not so costly as the original wrapper approach, because evaluation of predictor’s performance (for example by crossvalidation tests) is done only after ranking for a few pre-selected feature sets. The threshold for feature rejection is a part of the model selection procedure and may be determined using crossvalidation calculations. To avoid oscillations only those features that really improve the training results should be accepted. This area between filters and wrappers seems to be rather unexplored.

In the next section a new relevance index based on the Kolmogorov-Smirnov (KS) test to estimate correlation between the distribution of feature values and the class labels is introduced (used so far only for datasets with small number of features [4]). Correlation-based filters are very fast and easily compete with information-based filters. In section three empirical comparisons between KS filter, Pearson’s correlation based filter and other filters based on information gain are made on three widely used microarray datasets [5], [6], [7].

2 Theoretical Framework

2.1 Correlation-Based Measures

Pearson’s linear correlation coefficient is very popular in statistics [8]. For feature X with values x and classes C with values c treated as random variables it is defined as

$$\varrho(X,C) = \frac{\sum_i (x_i - \bar{x}_i)(c_i - \bar{c}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2 \sum_j (c_i - \bar{c}_i)^2}}. \tag{1}$$

$\varrho(X,C)$ is equal to ±1 if X and C are linearly dependent, and zero if they are completely uncorrelated. The simplest test estimating probability that two variables are related given the correlation $\varrho(X,C)$ is [8]:

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

where erf is the error function. Thus for $N = 1000$ samples linear correlation coefficients as small as 0.02 really signify probabilities of correlation around 0.5.

The feature list ordered by decreasing values of $\mathcal{P}(X \sim C)$ provides feature ranking. Similar approach is also taken with χ^2 statistics, but the problem in both cases is that for larger values of χ^2 or correlation coefficient probability $\mathcal{P}(X \sim C)$ is so close to 1 that ranking becomes impossible due to the finite numerical accuracy of computations. Therefore initial threshold for $\mathcal{P}(X \sim C)$ may be used in ranking only to determine how many features are worth keeping, although more reliable estimations may be done using crossvalidation or wrapper approaches.

Information theory is frequently used to define relevance indices. Mutual Information (MI) is defined as $MI(f,C) = H(f) + H(C) - H(f,C)$, where the entropy and joint entropy are:

$$H(f) = -\sum_i \mathcal{P}(f_i)\log(\mathcal{P}(f_i); \quad H(C) = -\sum_i \mathcal{P}(C_i)\log\mathcal{P}(C_i) \tag{3}$$

and

$$H(f,C) = -\sum_{i,j} \mathcal{P}(f_i, C_j)\log\mathcal{P}(f_i, C_j) \tag{4}$$

Symmetrical Uncertainty (SU) Coefficient is defined as [8]:

$$SU(f,C) = 2\left[\frac{MI(f,C)}{H(f)+H(C)}\right] \tag{5}$$

If a group of k features has already been selected, correlation coefficient may be used to estimate correlation between this group and the class, including inter-correlations between the features. Denoting the average correlation coefficient between these features and classes as $r_{kc} = \bar{\varrho}(\mathbf{X}_k, C)$ and the average between different features as $r_{kk} = \bar{\varrho}(\mathbf{X}_k, \mathbf{X}_k)$ the relevance of the feature subset may be defined as:

$$J(\mathbf{X}_k, C) = \frac{k r_{kc}}{\sqrt{k + (k-1)r_{kk}}}. \tag{6}$$

This formula has been used in the Correlation-based Feature Selection (CFS) algorithm [9] adding (forward selection) or deleting (backward selection) one feature at a time. Non-parametric, or Spearman's rank correlation coefficients may be useful for ordinal data types.

F-score is another useful index that may be used for ranking [10]:

$$F(C, f_i) = \frac{1}{(K-1)\sigma_i^2}\sum_k n_k\left(\bar{f}_{ik} - \bar{f}_i\right)^2 \tag{7}$$

where n_k is the number of elements in class k, $\bar{f}_{ik}$ is the mean and σ_{ki}^2 is the variance of feature f_i in this class. Pooled variance for feature f_i is calculated from:

$$\sigma_i^2 = \sigma^2(f_i) \ = \frac{1}{(n-K)}\sum_k (n_k - 1)\,\sigma_{ik}^2 \tag{8}$$

where $n = \sum_k n_k$ and K is the number of classes. In the two-class classification case F-score is reduced to the t-score ($F = t^2$).

Predominant correlation proposed by Liu *et al.* [11] in their Fast Correlation-Based Filter (FCBF) compares relations between feature-class and feature-feature. First ranking using the SU coefficient Eq. 5 is performed, and the threshold coefficient determining the number of features left is fixed. In the second step each feature f_i is compared to all f_j lower in ranking, and if their mutual $SU(f_i, f_j)$ coefficient is larger then $SU(C, f_j)$ then f_j is considered redundant and removed.

ConnSF, selection method based on a consistency measure, has been proposed by Dash *et al.* [12]. This measure evaluates for a given feature subset the number of cases in which the same feature values are associated with different classes. More precisely, a subset of feature values that appears n times in the data, most often with the label of class c, has inconsistency $n - n(c)$. If all these cases are from the same class then $n = n(c)$ and inconsistency is zero. The total inconsistency count is the sum of all the inconsistency counts for all distinct patterns of a feature subset, and consistency is defined by the least inconsistency count. Application of this algorithm requires discrete values of the features.

2.2 Kolmogorov-Smirnov Test for Two Distributions

The Kolmogorov-Smirnov (K-S) test [8] is used to evaluate if two distributions are roughly equal and thus may be used as a test for feature redundancy. The K-S test consists of the following steps:

- Discretization process creates k clusters (vectors from roughly the same class), each typically covering similar range of values.
- A much larger number of independent observation $n_1, n_2 > 40$ are taken from the two distributions, measuring frequencies of different classes.
- Based on the frequency table the empirical cumulative distribution functions $F1_i$ and $F2_i$ for two sample populations are constructed.
- λ(K-S statistics) is proportional to the largest absolute difference of $|F1_i - F2_i|$:

$$\lambda = \sqrt{\frac{n_1 n_2}{n_1 + n_2}} \sup |F1_i - F2_i| \quad \text{for} \quad i = 1, 2, ..., k. \tag{9}$$

When $\lambda < \lambda_\alpha$ then the two distributions are equal, where α is the significance level and λ_α is the K-S statistics for α [13]. One of the features with distribution that are approximately equal is then redundant. In experiments described below all training samples $n_1 = n_2 = n$ were used.

2.3 Kolmogorov-Smirnov Correlation-Based Filter Approach

Kolmogorov-Smirnov test is a good basis for the Correlation-Based Selection algorithm (K-S CBS) for feature selection. This algorithm is sketched in Fig. 1. Feature ranking is performed first, requiring selection of the ranking index. F-score index Eq. 7 is used in all calculations here. The threshold for the number of features left for further analysis may be determined in a principal way using the frapper approach, that is evaluating the

```
Algorithm K-S CBS:
Relevance analysis
1. Order features according to the decreasing values of relevance indices creating $\mathcal{S}$ list.
Redundancy analysis
2. Initialize $F_i$ to the first feature in the $\mathcal{S}$ list.
3. Use K-S test to find and remove from $\mathcal{S}$ all features for which $F_i$ forms an approximate redundant cover $\mathcal{C}(F_i)$.
4. Move $F_i$ to the set of selected features, take as $F_i$ the next remaining feature in the list.
5. Repeat step 3 and 4 until the end of the $\mathcal{S}$ list.
```

Fig. 1. A two-step Kolmogorov-Smirnov Correlation Based Selection (K-S CBS) algorithm

quality of results as a function of the number of features. In the second step redundant features are removed using the K-S test. The optimal α significance level for feature removal may also be determined by crossvalidation.

This is of course quite generic algorithm and other ranking indices and tests for equality of distributions may be taken instead. Two parameters – the threshold for relevancy and the threshold for redundancy – are successively determined using crossvalidation, but in some cases there may be a clear change in the value of these parameters, helping to find their optimal values.

3 Empirical Study

To evaluate the usefulness of K-S CBS algorithm experiments on three gene expression datasets [5], [6] [7] have been performed. Datasets used here [1] are quite typical for this type of applications. A summary is presented in Table 1.

1. **Leukemia** data is divided into training set consists of 38 bone marrow samples (27 of the ALL and 11 of the AML type), using 7129 probes from 6817 human genes; 34 test samples are provided, with 20 ALL and 14 AML cases.
2. **Colon Tumor** contains 62 samples collected from colon cancer patients, with 40 biopsies from tumor areas (labelled as "negative") and 22 from healthy parts of the colons of the same patients. 2000 out of around 6500 genes were pre-selected, based on the confidence in the measured expression levels.
3. **Diffuse Large B-cell Lymphoma** [DLBCL] has measurements of gene expression data for two distinct types of diffuse large lymphoma B-cells (this is the most common subtype of non-Hodgkin's lymphoma). There are 47 samples, 24 of them are from "germinal centre B-like" group while 23 are from "activated B-like" group. Each sample is represented by 4026 genes.

Splitting such small data into training and test subsets does not make much sense. Results reported below for all data are from the leave-one-out (LOO) calculations, deterministic procedure that does not require averaging or calculation of variance.

[1] Downloaded from http://sdmc.lit.org.sg/GEDatasets/Datasets.html

Table 1. Summary of microarray dataset properties

Title	# Genes	# Samples	# Samples per class				Source
Colon cancer	2000	62	40	tumor	22	normal	Alon [5]
DLBCL	4026	47	24	GCB	23	AB	Alizadeh [6]
Leukemia	7129	72	47	ALL	25	AML	Golub [7]

The original gene expression data contain real numbers. To calculate mutual information probabilities Eq. (3, 4) are needed, therefore the data has been discretized. This also helps to reduce the amount of noise in the original observations and facilitates direct use of such predictive techniques as the Naive Bayesian Classifier (NBC). Although quite sophisticated methods of discretization exist, for comparison of information selection techniques simple discretization of gene expression levels into 3 intervals is used here. Using the variance σ and the mean μ for a given gene any value larger than $\mu + \frac{\sigma}{2}$ is transformed to $+1$, any value in the $[\mu - \frac{\sigma}{2}, \mu + \frac{\sigma}{2}]$ interval is transformed to 0, and any value smaller than $\mu - \frac{\sigma}{2}$ becomes -1. These three values correspond to the over-expressions, baseline, and under-expression of genes. Results obtained after such discretization are in some cases significantly improved and are given in parenthesis in the tables below.

For each data set K-S CBS algorithm using F-measure (results with SU coefficient are similar) in the filtering stage is compared with the three state-of-the-art feature selection algorithms: FCBF [11], CorrSF [9], ConnSF [12]. The number of features selected obviously depends on the parameters of the feature selection method. The authors of the FCBF algorithm recommend taking the relevance threshold corresponding to the $n \log n$ features, and treating as redundant features with larger SU index between features than between the classes. The CorrSF correlation coefficient Eq. 1 is used in a forward best-first search procedure with backtracking up to 5 times before search is terminated, and selecting only those features that have larger feature-class correlations than correlation to already selected features. For ConsSF the usual practice is followed, searching for the smallest subset with consistency equal to that of the full set of attributes. One could introduce additional parameters in FCBF, CorrSF and ConnSF to change the preference of the relevance vs. redundancy and optimize them in the same way, but we have not done so. For comparison the K-S CBS algorithm is used with $\alpha = 0.05$, representing quite typical value of confidence. This value can easily be optimized for individual classifiers in the frapper approach, therefore results for other values are provided.

Table 2. Number of features selected by each algorithm

Data	Number of features selected				
	Full set	FCBF	CorrSF	ConnSF	K-S CBS$_F$
Colon Cancer	2000	9	17	4	5
DLCBL	4026	33	18	3	16
Leukemia	7129	52	28	3	118

Table 3. Balanced accuracy from the LOO test for C4.5, NBC, 1NN and SVM classifier on features selected by four algorithms, results on discretized data in parenthesis

Method	C 4.5				
Data	All features	FCBF	CorrSF	ConnSF	K-S CBS$_{F,\alpha=0.05}$
Colon Cancer	72.05 (68.30)	81.36 (80.11)	77.84 (80.11)	78.07 (78.07)	73.30 (68.30)
DLCBL	89.40 (74.55)	82.77 (85.14)	72.28 (89.49)	87.14 (85.24)	80.80 (85.24)
Leukemia	73.23 (85.74)	86.68 (95.72)	79.49 (93.74)	96.94 (95.74)	86.55 (85.74)
Average	78.22 (76.20)	83.60 (86.99)	76.53 (87.78)	87.38 (86.35)	80.22 (79.76)

Method	NBC				
Data	All features	FCBF	CorrSF	ConnSF	K-S CBS$_{F,\alpha=0.05}$
Colon Cancer	57.84 (66.59)	85.91 (90.68)	84.43 (88.18)	74.77 (79.32)	78.64 (66.59)
DLCBL	97.92 (91.58)	100.0 (100.0)	100.0 (100.0)	91.49 (89.40)	97.92 (93.66)
Leukemia	100.00 (82.55)	96.94 (100.0)	98.94 (100.0)	86.94 (100.0)	98.00 (82.55)
Average	85.25 (80.24)	94.28 (96.89)	94.46 (96.06)	84.40 (89.57)	91.52 (80.93)

Method	1NN				
Data	All features	FCBF	CorrSF	ConnSF	K-S CBS$_{F,\alpha=0.05}$
Colon Cancer	73.07 (64.55)	82.39 (83.18)	83.41 (78.41)	79.09 (93.75)	74.55 (64.55)
DLCBL	76.27 (74.46)	100.0 (97.83)	100.0 (100.0)	93.66 (93.48)	93.66 (91.39)
Leukemia	84.81 (88.81)	96.94 (100.0)	93.87 (100.0)	94.81 (100.0)	92.94 (88.81)
Average	78.05 (75.94)	93.11 (93.67)	92.42 (92.80)	89.18 (95.74)	87.05 (81.58)

Method	SVM				
Data	All features	FCBF	CorrSF	ConnSF	K-S CBS$_{F,\alpha=0.05}$
Colon Cancer	80.11 (70.80)	84.89 (80.11)	87.16 (83.41)	74.77 (75.80)	82.61 (70.80)
DLCBL	93.66 (95.74)	100.0 (100.0)	100.0 (100.0)	91.58 (91.58)	95.83 (91.49)
Leukemia	98.00 (88.81)	98.00 (100.0)	96.94 (100.0)	85.87 (100.0)	98.00 (96.00)
Average	90.59 (85.12)	94.29 (93.37)	94.70 (94.47)	84.08 (89.13)	92.15 (86.09)

Features selected by each algorithm serve to calculate balanced accuracy using four popular classifiers, decision tree C4.5 (with default Weka parameters), Naive Bayes (with single Gaussian kernel, or discretized probabilities), nearest neighbor algorithm (single neighbor only) and linear SVM with $C = 1$ (using Ghostminer implementation[2]). Each of these classifiers is of quite different type and may be used on raw as well as on the discretized data.

The number of features selected by different algorithms is given in Table 2. K-S CBF selected rather small number of features except for the Leukemia data, where significantly larger number of features has been created. Even for $\alpha = 0.001$ the number of features is 47, which is relatively large. Unfortunately with the small number of samples in the microarray data a single error difference in the LOO test is translated to quite large 1.6% for colon, 2.1% for DLCBL and 1.4% for leukemia. Thus although percentages may clearly differ the number of errors may be similar.

First observation from results given in Table 3 is that feature selection has significant influence on the performance of classifiers. Improvements for C4.5 on Leukemia

[2] http://www.fqs.pl/ghostminer/

Table 4. LOO balanced accuracy for different significance levels α for all data set; KSCBS$_F$ on standarized data

α	0.001	0.01	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45
Dataset	Colon cancer										
No. feat.	2	5	5	8	9	10	10	13	13	17	17
C4.5	77.61	80.34	73.30	77.84	66.25	70.80	70.80	74.09	74.09	69.32	69.32
NBC	82.61	67.95	78.64	74.89	79.89	82.16	82.16	78.64	78.64	81.93	81.93
1NN	78.64	75.34	74.55	72.61	72.05	71.82	71.82	71.82	71.82	76.82	76.82
SVM	72.50	72.50	82.61	81.36	81.36	81.36	81.36	80.34	80.34	84.89	84.89
Average	77.84	74.03	77.28	76.68	74.89	76.54	76.54	76.22	76.22	78.24	78.24
Dataset	DBCL										
No. feat.	7	13	16	22	22	30	43	43	43	63	63
C4.5	85.14	82.97	80.80	93.66	93.66	91.49	74.46	74.46	74.46	74.37	74.37
NBC	91.49	93.57	97.92	93.57	93.57	97.83	97.83	97.83	97.83	100.0	100.0
1NN	87.32	95.83	93.66	93.75	93.75	89.40	93.75	93.75	93.75	93.57	93.57
SVM	89.49	100.0	95.83	89.49	89.49	95.83	100.0	100.0	100.0	100.0	100.0
Average	88.36	93.09	92.05	92.62	92.62	93.64	91.51	91.51	91.51	91.99	91.99
Dataset	Leukemia										
No. feat.	47	75	118	167	207	268	268	331	331	456	456
C4.5	85.74	88.81	86.55	84.68	91.74	77.36	77.36	80.43	80.43	88.68	88.68
NBC	94.94	96.94	98.00	100.0	98.00	100.0	100.0	98.94	98.94	100.0	100.0
1NN	90.94	89.87	92.94	92.94	90.94	92.94	92.94	92.94	92.94	90.94	90.94
SVM	90.00	96.00	98.00	98.00	98.00	96.94	96.94	98.00	98.00	98.00	98.00
Average	90.41	92.91	93.87	93.91	95.17	91.81	91.81	92.58	92.58	94.41	94.41

exceed 20%, for NBC on colon cancer reach almost 30%, for 1NN on DLCBL almost 20% and for SVM on colon data over 7%. Discretization in most cases improves the results. For colon cancer SVM reaches the best result on all features (80.1%), and the highest accuracy on the 17 CorrSF selected features (87.2%), that also happens to be the largest subset. However, on the discretized data better results are achieved with Naive Bayes with 9 FCBF features (90.7%). For DLCBL with all features Naive Bayes reaches 97.9%, and 100% for both FCBF and CorrSF selections, with 1NN and SVM reaching also 100% on these features. For Leukemia again Naive Bayes is the winner, reaching 100% on all data, and for discretized data selected by FCBF, CorrSF and ConnSF achieving 100% balanced accuracy. K-S CBF always gives worse results on the discretized data, but on the raw data (K-S test is more appropriate for real-valued features) is not far behind.

It is clear that the default value for redundancy in K-S CBS is far from optimal; unfortunately Kolmogorov-Smirnov statistics can be used only to discover redundant features, but cannot be directly compared with relevance indices. In real applications estimation of optimal α using crossvalidation techniques for a given classifier will significantly improve results, as is evident from Table 4. Detailed analysis of the dependence of the number of features and balanced accuracy on α is presented in Table 4 starting from very small α.

With optimized α the best results with K-S CBS features are very similar to the best results of the other algorithms. For colon cancer SVM gives 84.9% on 17 features, which translates to 9 instead of 8 errors. For DBCL data SVM and Naive Bayes reach 100%, while for Leukemia 100% is also reached with Naive Bayes, although for somehow larger number of features. However, with such small statistics larger number of features is actually desirable to stabilize the expected profile. For example, with the original division between training and test data [7] a single gene gives 100% accuracy on the training set, but this does not mean that it is sufficient as it makes 3 errors on the test. It is much safer to use leave-one-out evaluation in this case.

4 Conclusions

Information filters may be realized in many ways [1]. They may help to reject some features, but the final selection should remove redundant features, not only to decrease dimensionality, but also to avoid problems that are associated with redundant features. Naive Bayes algorithm is clearly improved by removing redundancy, and the same is true for similarity-based approaches and SVM. Kolmogorov-Smirnov test for determination of redundant features requires only one parameter, the significance level, and is a well-justified statistical test, therefore it is an interesting choice for feature selection algorithms.

The K-S CBS algorithm presented here combines relevance indices (F-measure, Symmetrical Uncertainty Coefficient or other index) to rank and reduce the number of features, and uses Kolmogorov-Smirnov test to reduce the number of features further. It is computationally efficient and gives quite good results. Variants of this algorithm may identify approximate redundant covers $\mathcal{C}(f_i)$ for consecutive features f_i and leave in the $\mathcal{S}$ set only the one that gives best results (this will usually be the first one, with the highest ranking). Some ways of information aggregation could also be used, for example local PCA in the $\mathcal{C}(F_i)$ subspace. In this case the threshold for redundancy may be set to higher values, leaving fewer more stable features in the final set, and assuring that potentially useful information in features that were considered to be redundant is not lost. One additional problem that is evident in Table 4 and that frequently arises in feature selection for small microrarray data, but may also appear with much larger data [3], is stability of results. Adding more features may degrade results instead of improving them. We had no space here to review literature results for microarray data (see comparison in [14] or results in [15]) but they are all unstable and do not significantly differ from our results given in Tables 3 and 4. The instability problem may be addressed using the frapper approach to select most stable (and possible non-redundant) subset of features in $O(m)$ steps, where m is the number of features left for ranking. This and other improvements are the subject of further investigation.

Acknowledgement. This work was financed by the Polish Committee for Scientific Research grant 2005-2007 to WD; JB has been supported by the Polish Fundation of Science, grant (2006-2009) No.: 6ZR9 2006 C/06742 and (2007-2010) No.: N N519 1506 33.

References

1. Duch, W.: Filter methods. In: [3], pp. 89–118 (2006)
2. Toussaint, G.T.: Note on optimal selection of independent binary-valued features for pattern recognition. IEEE Transactions on Information Theory 17, 618–618 (1971)
3. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. (eds.): Feature extraction, foundations and applications. Physica Verlag, Springer, Heidelberg (2006)
4. Biesiada, J., Duch, W.: Feature Selection for High-Dimensional Data: A Kolmogorov-Smirnov Correlation-Based Filter Solution. In: Kurzynski, M., Puchala, E., Wozniak, M., Zolnierek, A. (eds.) Computer Recognition Systems. Proc. of the 4th International Conference on Computer Recognition Systems (CORES 2005). Advances in Soft Computing, vol. 9, pp. 95–104. Springer, Heidelberg (2005)
5. Alon, U., et al.: Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays. PNAS 96, 6745–6750 (1999)
6. Alizadeh, A.A., et al.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 403, 503–511 (2000)
7. Golub, T.R., et al.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science 286, 531–537 (1999)
8. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical recipes in C. The art of scientific computing. Cambridge University Press, Cambridge (1988)
9. Hall, M.A.: Correlation based feature selection for machine learning. PhD thesis, Dept. of Comp. Science, Univ. of Waikato, Hamilton, New Zealand (1999)
10. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8), 1226–1238 (2005)
11. Yu, L., Liu, H.: Feature selection for high-dimensional data: A fast correlation-based filter solution. In: Proceedings of the 12th International Conference on Machine Learning (ICML 2003), Washington, D.C., pp. 856–863. Morgan Kaufmann, San Francisco (2003)
12. Dash, M., Liu, H., Motoda, H.: Consistency based feature selection. In: Proc. 4th Pacific Asia Conference on Knowledge Discovery and Data Mining, pp. 98–109. Springer, Heidelberg (2000)
13. Evans, M., Hastings, N., Peacock, B.: Statistical Distributions. John Wiley & Sons, Chichester (2000)
14. Duch, W., Biesiada, J.: Margin-based feature selection filters for microarray gene expression data. International Journal of Information Technology and Intelligent Computing 1, 9–33 (2006)
15. Ding, C., Peng, H.: Minimum redundancy feature selection from microarray gene expression data. Journal of Bioinformatics and Computational Biology 3(2), 185–205 (2005)