Computing and Informatics, Vol. 36, 2017, 597–617, doi: 10.4149/cai_2017_3_597

COMPARISON OF FILTER TECHNIQUES FOR TWO-STEP FEATURE SELECTION

Peter Drotár, Slavomír Šimoňák, Emília Pietriková
Martin Chovanec, Eva Chovancová, Norbert Ádám
Csaba Szabó, Anton Baláž, Miroslav Biňas

Department of Computers and Informatics
Technical University of Košice
Letná 9
042 00 Košice, Slovakia
e-mail: peter.drotar@tuke.sk

Abstract. In the last decade, the processing of the high dimensional data became inevitable task in many areas of research and daily life. Feature selection (FS), as part of the data processing methodology, is an important step in knowledge discovery. This paper proposes nine variation of two-step feature selection approach with filter FS employed in the first step and exhaustive search in the second step. The performance of the proposed methods is comparatively analysed from the stability and predictive performance point of view. As the obtained results indicate the choice of the filter FS in the first stage has strong influence on the resulting stability. Here, the choice of univariate Pearson correlation coefficient based FS method appears to provide the most stable results.

Keywords: Feature selection, selection stability, high dimensionality, exhaustive search, bioinformatics

1 INTRODUCTION

The amount of available data is increasing by unprecedented rate. According to the recent estimates the amount of acquired and stored data will be several exabytes per year in 2025 [1]. There is a number of specific research areas that notably contribute to the amount of available big data. In addition to well known generators of big data such as YouTube or Twitter, bioinformatics is another potent contributor. There is expectation that only in genomics, as a part of bioinformatics, the amount of generated data will exceed one exabyte in 2025. The more courageous estimates mention even more then several zettabytes of data [1].

Large volume of data posses many challenges on subsequent processing. Besides the inevitable improvement in computational power and storage needs, more advanced data processing is also required. The ultimate goal is to be able to interpret the data and find the patterns that can answer underlying physical concept or phenomenon.

Thanks to the technologies such as microarrays or mass spectrometry, the disease diagnosis became an area of huge application of machine learning techniques. Biomarker discovery is important research topic. With the help of gene expressions obtained from microarray technology disease samples and control samples can be compared. Especially in cancer diagnosis, gene expressions provide more reliable and accurate diagnosis.

Bioinformatics domain is specific in the sense that data consists from several tens to hundreds of samples with very high dimensionality. Only small number of samples can be collected since the available resources are very limited and collection of specific sample types is often difficult. On the other hand, modern biotechnological tools enable collection of high number of attributes from one sample. This form of data gives rise to the curse of dimensionality, the challenging phenomenon that covers issues related to high dimensionality data not existing in low dimensional setting. High dimensionality can lead to over-fitting and degraded classification performance. One of the solutions is to employ feature selection or dimensionality reduction. Reducing number of features helps to better understand the data, improves classification performance and reduces computational requirements.

The feature selection (FS) is very vivid area of research [2, 3, 4]. The main goal of FS is to identify subset of the most significant features out of the original set and remove redundant or irrelevant features. There are several motivations for doing the feature selection. The FS as a preprocessing step, applied prior to classifier, helps to avoid over-fitting and reduces computational time of the prediction process. Moreover, there are several domains where the identification of the most important features is the goal.

In general, we recognize three different types of FS: filter, wrapper and embedded methods. Filter methods compute a score for each feature, and then select only the features with the best scores. Filter FS methods do not interact with classifier during selection process. Wrapper methods train a predictive model on subsets of features, and select the subset with the best score. The search for subsets can be deterministic or random. Finally, embedded methods determine the optimal subset of features directly by the trained weights of the classification method [5, 6].

One of the most popular applications of FS is the identification of small subset of the features that indicate occurrence of some disease. Recent advances and breakthroughs in molecular biology and bioinformatics brought huge number of microarray data that can be explored and utilized for cancer diagnosis. Cancer microarray data consists of a large number of genes with a small number of samples, and cancer is usually marked by a change in the expression levels of certain genes [6, 7]. Usually, there are only few genes that are relevant for diagnosis but the dimensionality of data is thousands of features.

A very relevant question arising during feature selection is how sensitive is the subset of selected features to data perturbation. The domain experts assume that the selected subset of features contains the most relevant discriminative information and try to explain underlying physical phenomenon. If there are numerous different subsets of the similar quality and the expert is presented only with one subset the resulting conclusion may be misleading. As such, it is of paramount importance to analyse stability of selection that qualitatively captures variations in the output of the FS procedure.

Recently, several gene selection methods have been proposed, for example [8, 9, 10, 11, 12]. These methods are based on very different approaches, such as mutual-information, principal component analysis or clustering. Another approach based on two stage feature selection was proposed in [13]. The method is relatively simple. First, to choose the subset of features using some simple filter method. And second, to apply search through all feature combinations to identify the subset of small dimension yielding highest prediction performance. Our aim in this work is to further extend this idea by proposing several new two stage feature selection methods. We investigate how the choice of the method in the first stage influences the overall performance of the method. We investigate not only the predictive performance, but also the stability of the methods. Stability is frequently neglected but it is very relevant aspect of FS [14].

The rest of the paper is organized as follows. In Section 2, the two-step feature selection is described. The term of feature selection stability is defined in Section 3 and datasets used in experiments are briefly described in Section 4. Finally, results of numerical experiments are given, followed by conclusions.

2 TWO-STEP FEATURE SELECTION

The proposed gene selection methods consist of two steps. First, filter FS algorithm selects subset of features for further processing. We focus mostly on univariate feature selection methods because of their relatively simple nature that does not pose high computational requirements. In the second step, the greedy feature selection is applied to search through all possible combination of features (genes) to find a combination providing the highest classification score.

In the next section we briefly describe nine filter feature selection algorithms that were used in the first stage of feature selection. A part of our analysis is also to compare how the selection of FS technique in the first stage influences the performance of the whole feature selection process.

2.1 Filter Techniques for Feature Selection

2.1.1 Decision Trees Feature Selection

Decision trees FS is based on extremely randomized trees proposed by [15]. Extremely randomized trees are ensemble method frequently used for supervised classification and regression tasks [16]. The main difference between extremely randomized trees and other tree based ensemble method is that extreme trees utilize all samples to grow the tree. They do not deploy bootstrapping. Additionally, the cutoff points for nodes splits are selected randomly. They have been proven to provide very competitive results in terms of accuracy and computational efficiency.

2.1.2 Spectral Feature Selection

Spectral feature selection is based on spectral graph theory, where graph spectrum is utilized to measure feature relevance [17]. The algorithm identifies features that align with the leading eigenvectors of similarity matrix. The leading eigenvectors contain information about structure of sample distribution and group similar samples into clusters. In this study, we used the unsupervised version of SPEC algorithm. This is the only unsupervised FS algorithm included in our study.

2.1.3 Feature Selection Based on ANOVA F Test Statistic

Analysis of variance (ANOVA) is the separation of variance attributable to one cause from the variance attributable to others [18]. In ANOVA, by utilization of F-tests the differences between means can be evaluated. The F-value can be calculated as follows:

$$F = \frac{MS_B}{MS_W} \tag{1}$$

where MS_B defined as $MS_B = \frac{\sum_i n_i(\overline{x}_i - \overline{x})^2}{m-1}$ characterizes between group variance and $MS_W = \frac{\sum_{ij} n_i(\overline{x}_{ij} - \overline{x}_i)^2}{n-m}$. Here, n_i is the number of samples in the i^{th} group, x_i denotes the sample mean in the i^{th} group, x_{ij} denotes the j^{th} observation in the i^{th} group and x denotes the overall mean of the data [19].

2.1.4 Bhattacharyya Distance Feature Selection

Bhattacharyya distance is a quantitative measure of class separability frequently used for feature selection. Bhattacharyya distance between two classes is defined as [20]:

$$D_B = \frac{1}{4}\ln\left\{\frac{1}{4}\left(\frac{\sigma_1^2}{\sigma_2^2} + \frac{\sigma_2^2}{\sigma_1^2} + 2\right)\right\} + \frac{1}{4}\left\{\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}\right\} \tag{2}$$

where σ_i is variance of i^{th} class and μ_i represents the mean of the i^{th} class. In case of the multidimensional distance, the variances are replaced by covariance matrices and the means become vectors. The further details can be found in [21].

2.1.5 Maximal Information Coefficient

Maximal information coefficient (MIC) is based on idea that if an association (linear or nonlinear) exists between two variables, then a grid can be drawn on scatter-plot of the two variables that partitions the data to encapsulate that relationship [22]. Consider all grids partitioning data into x and y bins, and let $I_G(x, y)$ be the empirical mutual information of a grid G with x and y bins, such that the probability distribution functions are replaced by the fraction of observations falling in that cell. MIC is the maximum value in characteristic matrix $M = (m_{x,y})$, where

$$m_{x,y} = \frac{\max I_G(x, y)}{\log \min\{x, y\}} \tag{3}$$

and maximum is taken over all x-by-y grids. The calculation of M through all possible grids is computationally demanding, therefore, the dynamic programming is used in practice.

2.1.6 Pearson Correlation Coefficient

The Pearson correlation coefficient ρ_p is given by

$$\rho_p = \frac{\sum_{i=1}^{N}(X_i - \overline{X})(Y - \overline{Y})}{(N - 1)S_x S_y} \tag{4}$$

where $S_x^2 = \sum_{i=1}^{N}(X_i - \overline{X})^2/(N - 1)$ and $S_y^2 = \sum_{i=1}^{N}(Y_i - \overline{Y})^2/(N - 1)$ and N is the number of samples.

2.1.7 ReliefF

Relief is feature selection algorithm [23] that relies entirely on statistical analysis and employs only few heuristics. It selects most of the relevant features even though only a small number of them is necessary for prediction. In most cases it does not help with redundant features. The ReliefF algorithm [24] is not limited to two class problems, is more robust and can deal with incomplete and noisy data. Similarly to Relief, ReliefF randomly selects an instance, but then searches for k of its nearest neighbours from the same class, called the nearest hits, and also k nearest neighbours from each of the different classes, called the nearest misses [24]. In comparison with other filter techniques used in this study, this is the multivariate filter FS method.

2.1.8 Gini Index

The Gini index is frequently used to estimate and analyse the distribution of features through different classes. It was proposed by Gini and firstly used for estimation of income over population. The inequality is represented by area under the Lorenz curve. According [25] the Gini coefficient can be estimated as

$$G = \frac{\sum_{i=1}^{n}(2i - n - 1)wcp(t, c_i)}{n^2\mu} \tag{5}$$

where μ is the sample mean and n is the number of classes. The term $wcp(t, c_i)$ is defined as $wcp(f, c_i) = \frac{Pr(f|c_i)}{\sum_{k=1}^{|C|} Pr(f|c_k)}$ [26], where C is set of class labels.

2.2 Brute Force Feature Selection

After selection of the top N_{fs1} genes we search for the smaller group that bears the discriminative information. Even though we have significantly reduced dimension of data, searching through remaining genes is still computationally demanding. Results presented in [13] suggest that selecting three features leads to excellent classification performance in many cases. Therefore in order to reduce computational time we chose number of selected features in second stage $N_{fs2} = 3$, and search through all possible combination of three features. To evaluate predictive performance we use linear support vector machines and 4-fold cross-validation with stratified sampling.

2.3 Classifiers

2.3.1 Support Vector Machines

To evaluate prediction performance of different feature combination in second stage of FS we use linear SVM. Then, to evaluate the classification performance on test data and asses influence FS on classification we utilize nonlinear SVM with radial basis function (RBF) kernel [27].

The underlying idea of SVM classifiers is to calculate a maximal margin hyperplane $h(x)$ separating two classes of the data. The hyperplane is defined as

$$h(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b \tag{6}$$

where $\mathbf{w}$ is the normal vector of this hyperplane, $\mathbf{x}$ is the input vector, $-b/\mathbf{w}$ is the distance from the origin perpendicular to the hyperplane [28]. By replacing the dot product with nonlinear kernel function the hyperplane is found in new transformed feature space. The RBF kernel used in this paper can be expressed as

$$K(x, x_i) = e^{\frac{-\|x - x_i\|^2}{2\gamma^2}} \tag{7}$$

where γ controls the width of RBF function. The parameters kernel gamma γ and penalty parameter C were optimized using a grid search of possible values. Specifically, we searched over the grid (C, γ) defined by the product of the sets $C = [10^{-2}, \ldots, 10^2]$, $\gamma = [10^{-4}, \ldots, 10^2]$.

2.3.2 AdaBoost

AdaBoost belongs to the group of classifiers producing accurate prediction results by combining simple learners [29]. This is known as boosting. The main idea behind the AdaBoost algorithm is to combine multiple weak learners through voting to form strong learner. The only requirement for the weak learner is that the correct recognition rate should be slightly more than random guess.

It works by repeatedly adding new weak classifiers in a series of stages. Assuming training data $\mathbf{X}$ and class labels $\mathbf{y}$, $y_i \in \{-1, 1\}$ and $i = 1, \dots, n$, let us initialize weights β_i for each sample. Then fit the training data with weights β_i to produce weak classifier $b_j((\mathbf{X}))$. And for each classifier compute the empirical risk err_j and basis weight w_j given by $w_j = \log((1 - err_j)/err_j)$. After, the weights are repeatedly updated as $\beta_i = \beta_i \exp(w_j l(y_i \neq b_j(\mathbf{x}_i)))$. The resulting strong classifier is defined as

$$C(\mathbf{x}) = \operatorname{sign}\left(\sum_{j=1}^{m} w_j \cdot b_j((\mathbf{x}))\right). \tag{8}$$

The distribution of weights over training set is kept, and the weight of incorrectly classified samples is increased in each stage. By doing this, the weak learners are more focused on the samples that are difficult to correctly classify. The resulting strong classifier is build by selected weak classifiers through weighted majority vote. AdaBoost has proven effective in many real-world applications.

3 FEATURE SELECTION STABILITY

The *stability* of FS algorithm was defined by Kalousis [14] as *the robustness of the feature preferences it produces to differences in training sets drawn from the same generating distribution.* Stability is indicator of the feature selection reproducibility. The instability of FS technique reduces confidence in importance of selected features. As a such, high stability of FS is crucial for reliable results. It is equally important as high classification accuracy when evaluating FS performance.

There are several reasons for instability of the FS method. One possible cause is that the majority of FS algorithms are designed without consideration of stability aspects, and aim only on selecting minimal subset of features with highest classification accuracy [30, 31]. Another cause is the existence of multiple sets of true markers, i.e. it is possible that for given data there are many markers that are highly correlated with the data. Finally, it is known that very high dimensionality combined with small sample size causes serious problems in machine learning resulting also in algorithm instability [32].

To measure the stability of FS we used two different metrics Kuncheva stability index [33] and Average Tanimoto Index (ATI) [14].

Assume set of all available features $F = \{f_1, f_2, \dots, f_D\}$, where D is the dimensionality of the dataset. Next, denote the subset of features selected by feature selection algorithm as S, where $S \subset F$. Then, $\mathcal{S} = \{S_1, S_2, \dots, S_n\}$ is a system of n feature subsets acquired from n repetitive runs of FS algorithm on different samplings of given dataset. The feature subsets of the same dimensionality d are denoted as $S_{i,d}, S_{j,d} \subset F$, where $1 \leq d \leq D$.

The first stability measure considered in this work was proposed by Kuncheva [33]. Kuncheva stability index $\mathcal{K}$ is based on cardinality of the intersection of two feature subsets with correction for chance. Kuncheva index $\mathcal{K}$ for set of features $\mathcal{S} = \{S_{1d}, \dots, S_{nd}\}$ with subset size d is defined as:

$$\mathcal{K}(\mathcal{S}) = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} K_C(S_{id}, S_{jd}) \tag{9}$$

where $\mathcal{K}_C$ is consistency index. Consistency index for two subsets of features $S_{i,d}, S_{j,d} \subset F$ is given as

$$\mathcal{K}_C(S_{i,d}, S_{j,d}) = \frac{|S_{i,d} \cap S_{j,d}| \cdot |F| - d^2}{d(|F| - d)} \tag{10}$$

where $|S_{i,d}| = S_{j,d} = d$.

Alternatively, we measure stability by Tanimoto index that is basically adaptation of Tanimoto distance measuring the overlap between two sets of arbitrary cardinality [14]. Usage of another metric provides more objectivity when evaluating the results. Different metric can additionally provide different view on the data and uncover hidden patterns in data. The average Tanimoto index between two feature subsets S_i and S_j is computed as follows:

$$ATI(S_i, S_j) = 1 - \frac{|S_i| + |S_j| - 2|S_i \cap S_j|}{|S_i| + |S_j| - |S_i \cap S_j|}. \tag{11}$$

Both measures $\mathcal{K}$ and ATI can achieve maximum value 1 when subsets are identical. The empty intersection, i.e. completely dissimilar subsets, are indicated by $ATI = 0$ and $\mathcal{K}_{\mathcal{C}} = -1$.

4 DATA

For the purpose of our study, we used six high dimensional microarray datasets. Datasets contain only tens of samples but several thousand of features. This is typical dataset that tends to suffer from the curse of dimensionality.

The overview of datasets is summarized in Table 1. All datasets represent binary classification task or were converted to binary classification task. The datasets that required conversion were BUR and GOL. In the case of BUR dataset the class of Crohn's disease and ulcerative colitis were pooled together. G1999 dataset is modified to two-class dataset by merging ALL-T and ALL-B together.

All datasets are publicly available.

Dataset Name	Source	# Samples	# Features	# Class 0	# Class 1
BUR	Burczynski [34]	127	22 283	85	42
CHO	Chowdary [35]	104	22 283	62	42
GOL	Golub [36]	72	7 129	47	25
GOR	Gordon [37]	181	12 533	94	87
TIA	Tian [38]	173	12 625	36	137
POM	Pomeroy [39]	60	7 128	21	39

Table 1. Datasets used in this study

5 NUMERICAL EXPERIMENTS

We evaluated and compared performance of the proposed FS methods from stability point of view and analyzed how the particular FS method influences the prediction performance.

To asses the stability of different FS methods we performed computer experiments. In the first stage, filter FS method was applied to select top 50 features for further processing. In the second stage, exhaustive search, through all possible combinations of three genes, found one combination yielding the highest area under ROC curve (AUC) score.

AUC is objective measure robust against distribution changes in training data and is not affected by subjective factors. Moreover, AUC provides reliable results even in case of class imbalance problems. AUC is defined as:

$$AUC = \frac{1 + TP - FP}{2} \tag{12}$$

where TP is proportion of correctly classified positive examples and FP is the ratio of misclassified negative examples to the total number of negative samples.

5.1 Stability Evaluation

First, we evaluated how the choice of filter FS in the first stage influences the overall stability of the resulting subset of features by comparing FS stability at the output of the second stage of the FS. Three features were selected in each of 500 repetitive runs. For each of 500 runs we used stratified sampling of the dataset with 80 % of samples utilized for every repetition. Nine different filter FS methods were employed at the first stage of FS: decision trees FS (*Tree*), Spectral FS (*Spec*), FS based on ANOVA test ($ANOVA$), Fischer FS, FS based on Bhattacharyya distance (*BDist*), MIC, Pearson correlation coefficient (*Pearson*), ReliefF algorithm (*Relieff*) and Gini Index (*Gini*). The histogram of the FS output is visualised in Figure 1. The histogram shows the distribution of selected features at the output of two-step FS process.

The FS methods of the similar nature, even though they are used only in the first stage, display very similar behaviour at the output. ANOVA, Pearson FS and

Fischer FS that have similar statistical background yield very similar pattern in histogram. Another group is formed by Bhattacharyya distance FS, MIC, Gini FS and ReliefF selecting mostly the features in interval 1800-2200. The rationale behind these methods is different but their outputs are alike. We assume that these methods were able to correctly identify truly significant features and prefer these features in selection. However, our aim here is to focus on FS stability. From stability point of view, the output of algorithms varies with each repetition. As can be seen only few features achieve frequency of the selection higher than one hundred.

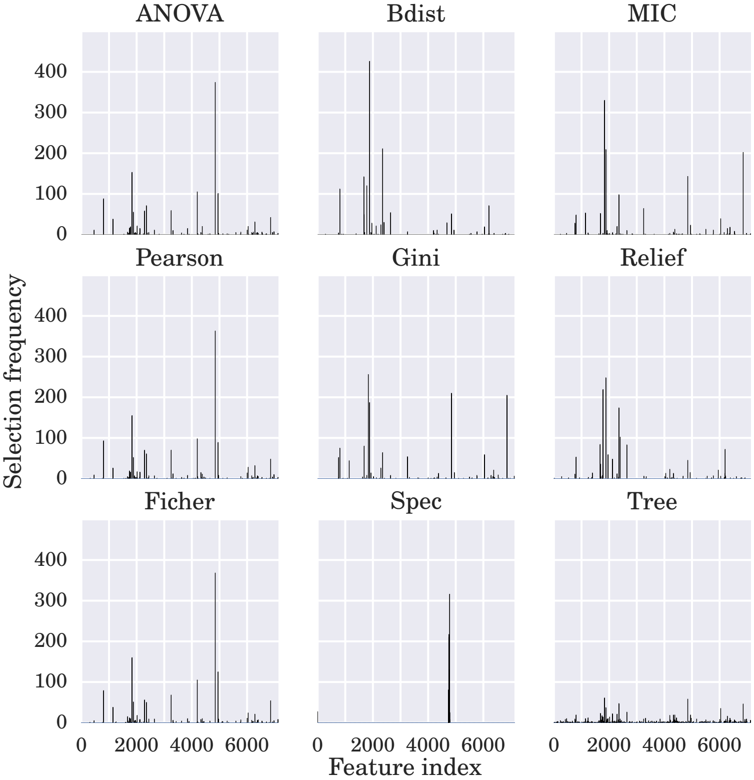

Figure 1. Frequencies of selected features for the Golub dataset

In order to obtain better insight and quantitatively measure stability we employ Kuncheva index and average Tanimoto index. We expect that the stability at the output of the second stage of FS is influenced by FS algorithm used in the first stage. Results are presented in Table 2 for evaluation using Kuncheva index and in Table 3 for average Tanimoto index (*ATI*). The best performances are in bold.

The both measures demonstrate similar trends. This increases confidence towards the results. There are different values of stability for different datasets used in these experiments. This is expected, since datasets can be of different complexity, and the underlying pattern can be more recognizable in not so complex datasets. It should be also noted that there is no single method that would dominate in all datasets. But rather the performance of the method is influenced by the data under the test.

The filter method yielding the highest stability score averaged over all evaluated databases is FS based on Pearson correlation coefficient followed by Fischer FS and Spectral method. On the other hand, the lowest stability was provided when FS based on decision trees algorithm was employed in the first stage of FS, with $\mathcal{K} = 13.2\,\%$ and $ATI = 8.4\,\%$. Slightly higher stability was achieved using relieff FS method. The stability was comparably less than for other tested methods. Note, that decision trees and relieff are both multivariate techniques. In contrast to Pearson correlation coefficient FS and Fischer FS that are typical examples of univariate FS methods. Obtained results are in alignment with our previous conclusions published in [40, 32] which showed that univariate FS methods provide better stability than multivariate FS methods. Presented experiments proved that this is also true when filter FS is being used in the first stage of two-step feature selection. Additionally, our initial hypothesis was confirmed and it is clear that FS method applied in the first-stage significantly influences the resulting stability.

In general we can notice that values of stability are relatively low for two stage FS approach. This is probably due to the fact that only three features are selected at the output of FS process. There can be more features that are relevant for predicting target variable. In this case forcing the algorithm to choose only three of them leads to changes in selection for each repetitive application of FS algorithm. The high number of redundant features can have similar effect causing strong variations in the group of selected features. Increasing subset size to cover all features significant for prediction of target variable may help to increase stability, however our aim is to find small subset of features without redundant or partially relevant features.

Previous results showed that FS technique used in the first stage has influence on the overall stability. The question that arises is what is the relationship between the overall stability and stability of FS in the first stage. Has the stability notably changed in the second stage in comparison with the first stage of FS? In order to obtain better insight on stability we evaluated stability of FS methods used in the first stage, i.e. we measured stability at the output of the first stage of the FS process. Altogether 50 features were selected in every run. Similarly to the previous experiment, we run 500 repetitive loops where for each repetition we used stratified sampling of the dataset with 80 % of samples utilised in each loop.

The stability results measured by $\mathcal{K}$ and ATI are provided in Table 4, and Table 5, respectively. There are two methods showing significantly higher stability than other: *Spec* FS and *Bdist* FS. The *Spec* achieves as high as 100 % stability ratio, that is quite unexpected. Closer analysis of selected features revealed that exactly the same features are chosen in each run of the algorithm. However, these features are located next to each other, indicating that the algorithm is probably stuck in some kind of local minima and is not working properly. This is later confirmed by the results on predictive performance. The further investigation is necessary to explain whether this occur due to the unsupervised nature of the method or it is some method specific issue. Based on this we consider *Bdist* as the most stable method achieving highest stability on all datasets. It is quite unexpected to see that *Bdist* FS scored under average for measurements performed at the second stage of FS after greedy forward selection. The same is true for the second most stable algorithm: *relieff*. This indicates that the stability of the filter method itself does not ensure overall stability when used in the two step feature selection process. On the other hand, the *Tree* method that is highly unstable by itself transfers this behaviour to a two-step method, where *Tree* based two-stage FS is again the least stable method.

	ANOVA	Bdist	MIC	Pearson	Gini	Relieff	Fischer	Spec	Tree	Mean
GOL	34.3	34.3	**45.7**	35.2	43.9	19.5	33.0	29.8	15.8	32.4
GOR	15.5	33.1	62.9	**71.6**	67.0	38.2	71.1	47.2	31.6	48.7
TIA	7.5	6.4	7.2	7.8	8.1	**8.4**	8.0	0.0	7.8	6.8
BUR	10.3	8.8	8.6	12.2	9.6	6.3	11.1	**34.3**	7.6	12.1
CHO	34.7	14.2	13.6	32.9	18.6	17.2	33.0	**46.2**	9.2	24.4
POM	**9.9**	6.3	6.3	9.4	8.2	6.2	9.6	7.7	7.2	7.9
Mean	18.7	17.2	24.1	**28.2**	25.9	16.0	27.6	27.5	13.2	22.0

Table 2. Stability measured by $\mathcal{K}$ at the output of two-step FS. Different filter FS techniques used in the first stage.

	ANOVA	Bdist	MIC	Pearson	Gini	Relieff	Fischer	Spec	Tree	Mean
GOL	22.5	22.3	**31.9**	23.1	30.4	12.4	21.5	20.5	10.0	21.6
GOR	10.4	21.3	48.5	**58.2**	53.1	24.5	58.0	33.6	20.9	36.5
TIA	4.6	3.9	4.4	4.8	5.0	**5.2**	5.0	0.0	4.8	4.2
BUR	6.4	5.5	5.3	7.6	5.9	3.9	6.9	**24.6**	4.7	7.9
CHO	23.1	9.0	8.6	21.6	12.1	10.8	21.7	**33.1**	5.7	16.2
POM	**6.2**	3.9	3.9	5.8	5.1	3.8	6.0	4.8	4.5	4.9
Mean	12.2	11.0	17.1	**20.2**	18.6	10.1	19.8	19.4	8.4	15.2

Table 3. Stability measured by *ATI* at the output of two-step FS. Different filter FS techniques used in the first stage.

For comparison we display the histogram of the selected features at the output of the first (filter) stage of FS in the Figure 2. The depicted behaviour is in agreement with the results captured by Kuncheva and Average Tanimoto index presented in Tables 4 and 5.

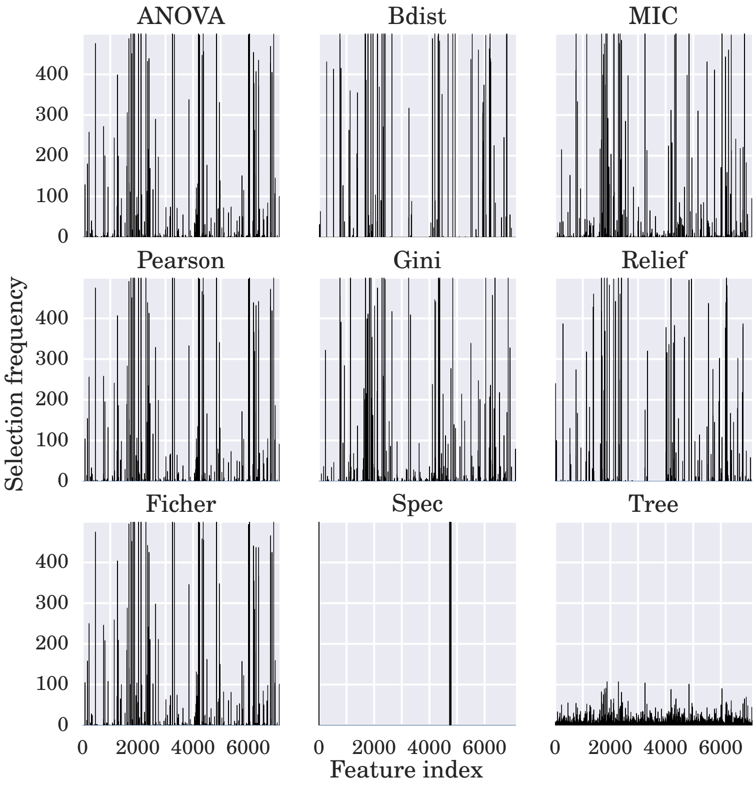

Figure 2. Frequencies of selected features for the Golub dataset at the output of first (filter) stage of FS

	ANOVA	Bdist	MIC	Pearson	Gini	Relieff	Fischer	Spec	Tree	Mean
GOL	70.4	**83.0**	67.7	71.0	67.9	67.7	70.8	**100.0**	2.7	66.8
GOR	80.7	**89.5**	79.1	80.3	76.8	80.5	80.2	**100.0**	2.5	74.4
TIA	43.6	**58.1**	17.9	43.9	35.0	41.9	43.6	**100.0**	0.8	42.8
BUR	60.3	**77.2**	53.2	61.5	52.7	63.3	61.8	**100.0**	1.3	59.0
CHO	55.7	**79.6**	75.4	56.2	78.4	63.6	55.7	**100.0**	1.3	62.9
POM	33.1	**57.6**	16.4	32.9	28.9	39.8	33.0	46.1	0.7	32.0
Mean	57.3	**74.2**	51.6	57.6	56.6	59.5	57.5	**91.0**	1.6	56.3

Table 4. Comparison of stability of filter FS techniques. Stability measured by $\mathcal{K}$ in the first stage.

	ANOVA	Bdist	MIC	Pearson	Gini	Relieff	Fischer	Spec	Tree	Mean
GOL	70.5	**71.4**	51.7	55.6	51.9	51.8	55.4	**100.0**	1.8	56.7
GOR	67.9	**81.3**	65.8	67.4	62.7	67.8	67.2	**100.0**	1.5	64.6
TIA	28.4	**41.7**	10.2	28.7	21.6	26.9	28.5	**100.0**	0.6	31.9
BUR	43.8	**63.3**	36.6	44.9	36.2	46.7	45.2	**100.0**	0.8	46.4
CHO	40.1	**67.0**	60.8	40.6	64.8	48.6	40.1	**100.0**	0.8	51.4
POM	20.5	**41.4**	9.4	20.3	17.5	25.5	20.4	44.2	0.7	22.2
Mean	45.2	**61.0**	39.1	42.9	42.4	44.6	42.8	**90.7**	1.0	45.5

Table 5. Comparison of stability of filter FS techniques. Stability measured by ATI in the first stage.

5.2 Prediction Performance Evaluation

Another aspect relevant for the assessment of FS methods is how the FS influences the prediction performance. In order to evaluate prediction performance, the selected subset of features is fed to classifier. As a classifier we utilized SVM with RBF kernel and AdaBoost. Classifier validation was conducted using stratified 5-fold cross-validation. The process was repeated ten times where the AUC was averaged over ten repetitions. Training and testing features were normalized before classification on a per-feature basis to have zero mean and a standard deviation of one.

The AUC for different FS methods evaluated on databases listed in Table 1 is presented in Table 6 for SVM classifier and in Table 7 for AdaBoost classifier. In contrast to the stability, the choice of FS method does not significantly influence the prediction performance. However, there are still some differences depending on FS used. In average the AUC score of SVM classifier is higher than Adaboost for all datasets and all FS methods. The highest difference between the best and the worst performance on investigated databases occurred for the case of *CHO* database, where the MIC FS outperformed the FS based on Bhattacharyya distance by almost 9 points ($AUC = 97.1$ to $AUC = 88.5$). In average, the highest prediction performance was achieved by application of Fischer FS in the first stage, but the AUC is only slightly better compared to other FS methods.

	ANOVA	Bdist	MIC	Pearson	Gini	relieff	Fischer	Spec	Tree	mean
GOL	90.5	93.0	91.7	91.7	90.8	92.5	**94.4**	73.8	89.5	89.8
GOR	95.4	94.6	96.6	96.4	93.0	95.7	**96.8**	93.7	95.6	95.3
TIA	75.2	76.8	74.7	75.3	73.8	77.2	75.0	**78.2**	77.2	75.9
BUR	**93.6**	91.9	92.0	91.3	88.5	93.2	92.6	69.1	86.8	88.8
CHO	94.9	88.5	**97.1**	95.8	95.9	88.7	95.0	80.2	88.9	91.7
POM	63.6	64.7	59.8	58.5	60.9	**65.4**	60.2	61.5	60.8	61.7
mean	85.5	84.9	85.3	84.8	83.8	85.4	**85.7**	76.1	83.1	83.9

Table 6. AUC of two-step FS methods. SVM classifier. Different filter FS techniques used in first stage.

	ANOVA	Bdist	MIC	Pearson	Gini	Relieff	Fischer	Spec	Tree	Mean
GOL	90.6	90.1	**91.0**	90.6	90.1	90.4	89.4	74.3	87.0	88.2
GOR	**95.5**	94.8	95.3	94.8	92.1	94.6	**95.5**	92.6	94.1	94.4
TIA	72.1	**73.6**	71.2	72.4	70.1	70.2	70.8	70.5	72.3	71.5
BUR	90.3	**91.0**	87.5	89.6	87.9	89.2	88.5	65.5	85.0	86.1
CHO	89.7	87.7	92.3	90.9	**91.7**	91.1	86.0	78.2	89.2	88.5
POM	57.9	**63.0**	59.3	58.7	60.3	61.3	59.3	53.4	59.5	59.2
Mean	82.7	**83.4**	82.8	82.8	82.0	82.8	81.6	72.4	81.2	81.3

Table 7. AUC of two-step FS methods. AdaBoost classifier. Different filter FS techniques used in first stage.

6 CONCLUSION

We presented an approach for selection of very small subset of genes in high dimensional setup and analysed the stability of this approach. The stability of filter techniques was analysed and in the next step these filter techniques were used to build two-step feature selection methods with different filter FS applied in the first stage of FS process. The two stage feature selection consists of filter method used in the first stage and the exhaustive search over reduced feature space in the second stage. When comparing only filter FS techniques on high dimensional datasets the highest stability was reached by *BDist* FS that was considerably more stable than other techniques. Surprisingly, this high stability of *BDist* FS did not propagate to the stability of two stage method, since, the method utilizing *BDist* FS in the first stage achieved one of the lowest stabilities. However, from the results presented in this paper it is clear that the choice of FS in the first stage of the proposed approach clearly influences the overall stability of feature selection. This is confirmed by the fact that *Tree* FS, as the least stable, when evaluated in the first stage, yields also the lowest stability when being assembled into two stage FS algorithm. The exact relationship between the choice of FS and the resulting stability is still unknown and it will be subject of further research. Recommended approach is to use either Pearson correlation based FS or Fischer score based FS in the first stage to gain the higher stability. The selection of filter FS is not so crucial from prediction performance point of view, but still, the choice of a filter can boost the prediction accuracy by several points.

Acknowledgement

This work was supported by the Slovak Research and Development Agency under the contract No. APVV-15-0055, grant FEI-2015-5 and European Intergovernmental Framework COST Action CA15140: Improving Applicability of Nature-Inspired Optimization by Joining Theory and Practice.

REFERENCES

[1] Stephens, Z. D.—Lee, S. Y.—Faghri, F.—Campbell, R. H.—Zhai, C.—Efron, M. J.—Iyer, R.—Schatz, M. C.—Sinha, S.—Robinson, G. E.: Big Data: Astronomical or Genomical? PLoS Biology, Vol. 13, No. 7, 2015, doi: 10.1371/journal.pbio.1002195.

[2] Bolon-Canedo, V.—Sanchez-Marono, N.—Alonso-Betanzos, A.: Feature Selection for High Dimensional Data. Springer, London, 2015, doi: 10.1007/978-3-319-21858-8.

[3] Boor, S.: Maximum Coverage Method for Feature Subset Selection for Neural Network Training. Computing and Informatics, Vol. 30, 2011, No. 5, pp. 901–912.

[4] Kovalčík, M.—Feciľak, P.—Jakab, F.—Dudiak, J.—Kolcun, M.: Cost-Effective Smart Metering System for the Power Consumption Analysis of Household. International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 5, 2014, No. 8.

[5] Guyon, I.—Elisseeff, A.: An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, Vol. 3, 2003, pp. 1157–1182.

[6] Chen, H.—Zhang, Y.—Gutman, I.: A Kernel-Based Clustering Method for Gene Selection with Gene Expression Data. Journal of Biomedical Informatics, Vol. 62, 2016, pp. 12–20. Available on: `http://www.sciencedirect.com/science/article/pii/S1532046416300375`.

[7] Nanni, L.—Brahnam, S.—Lumini, A.: Combining Multiple Approaches for Gene Microarray Classification. Bioinformatics, Vol. 28, 2012, No. 8, pp. 1151–1157. Available on: `http://bioinformatics.oxfordjournals.org/content/28/8/1151.abstract`.

[8] Liu, H.—Mo, Y.—Zhao, J.: Conditional Dynamic Mutual Information-Based Feature Selection. Computing and Informatics, Vol. 31, 2012, No. 6, pp. 1193–1216.

[9] Rakkeitwinai, S.—Lursinsap, C.—Aporntewan, C.—Mutirangura, A.: New Feature Selection for Gene Expression Classification Based on Degree of Class Overlap in Principal Dimensions. Computers in Biology and Medicine, Vol. 64, 2015, pp. 292–298. Available on: `http://www.sciencedirect.com/science/article/pii/S0010482515000396`.

[10] Bala, R.—Agrawal, R. K.: Clustering in Conjunction with Wrapper Approach to Select Discriminatory Genes for Microarray Dataset Classification. Computing and Informatics, Vol. 31, 2012, No. 5, pp. 921–938.

[11] Kleinová, K.—Feciľak, P.: New Approach to Remote Laboratory in Regard to Topology Change and Self-Repair Feature. Central European Journal of Computer Science, Vol. 2, 2012, No. 3, pp. 344–353.

[12] Kriska, M.—Janitor, J.—Feciľak, P.: Dynamic Routing of IP Traffic Based on QoS Parameters. International Journal of Computer Networks and Communications, Vol. 6, 2014, No. 4, p. 11.

[13] Wang, L.—Chu, F.—Xie, W.: Accurate Cancer Classification Using Expressions of Very Few Genes. IEEE/ACM Transactions on Computational Biology and Bioinformatics, Vol. 4, 2007, No. 1, pp. 40–53.

[14] Kalousis, A.—Prados, J.—Hilario, M.: Stability of Feature Selection Algorithms: A Study on High-Dimensional Spaces. Knowledge and Information Systems, Vol. 12, 2007, No. 1, pp. 95–116. Available on: `http://dx.doi.org/10.1007/s10115-006-0040-8`.

[15] Geurts, P.—Ernst, D.—Wehenkel, L.: Extremely Randomized Trees. Machine Learning, Vol. 63, 2006, No. 1, pp. 3–42. Available on: `http://dx.doi.org/10.1007/s10994-006-6226-1`.

[16] Chiang, D.—Chen, W.—Wang, Y. F.—Hsu, C. F.: The Irrelevant Values Problem in the ID3 Tree. Computers and Artificial Intelligence, Vol. 19, 2000, No. 2, pp. 169–182.

[17] Zhao, Z.—Liu, H.: Spectral Feature Selection for Supervised and Unsupervised Learning. Proceedings of the 24th International Conference on Machine Learning (ICML '07), New York, NY, USA. ACM, 2007, pp. 1151–1157. Available on: `http://doi.acm.org/10.1145/1273496.1273641`.

[18] Everitt, B.: The Cambridge Dictionary of Statistics. Cambridge University Press, Cambridge, UK, 2006.

[19] Grünauer, A.—Vincze, M.: Using Dimension Reduction to Improve the Classification of High-Dimensional Data. OAGM Workshop, 2015.

[20] Coleman, G. B.—Andrews, H. C.: Image Segmentation by Clustering. Proceedings of the IEEE, Vol. 67, 1979, No. 5, pp. 773–785.

[21] Reyes-Aldasoro, C. C.—Bhalerao, A.: The Bhattacharyya Space for Feature Selection and Its Application to Texture Segmentation. Pattern Recognition, Vol. 39, 2006, No. 5, pp. 812–826. Available on: `http://dx.doi.org/10.1016/j.patcog.2005.12.003`.

[22] Reshef, D. N.—Reshef, Y. A.—Finucane, H. K.—Grossman, S. R.—McVean, G.—Turnbaugh, P. J.—Lander, E. S.—Mitzenmacher, M.—Sabeti, P. C.: Detecting Novel Associations in Large Data Sets. Science, Vol. 334, 2011, pp. 1518–1524.

[23] Kira, K.—Rendell, L. A.: A Practical Approach to Feature Selection. Proceedings of the Ninth International Workshop on Machine Learning (ML '92), San Francisco, CA, USA. Morgan Kaufmann Publishers Inc., 1992, pp. 249–256.

[24] Robnik-Šikonja, M.—Kononenko, I.: Theoretical and Empirical Analysis of ReliefF and RReliefF. Machine Learning, Vol. 53, 2003, No. 1, pp. 23–69.

[25] Dixon, P. M.—Weiner, J.—Mitchell-Olds, T.—Woodley, R.: Bootstrapping the Gini Coefficient of Inequality. Ecology, Vol. 65, 1987, No. 5, pp. 1548–1551.

[26] Singh, S. R.—Murthy, H.—Gonsalves, T.: Feature Selection for Text Classification Based on Gini Coefficient of Inequality. Proceedings of the Fourth Workshop on Feature Selection in Data Mining, Proceedings of Machine Learning Research (PMLR), Vol. 10, 2010, pp. 76–85.

[27] Vapnik, V.: Statistical Learning Theory. 1st ed. John Willey & Sons, London, 1998.

[28] Vo, V.—Luo, J.—Vo, B.: Time Series Trend Analysis Based on K-Means and Support Vector Machine. Computing and Informatics, Vol. 35, 2016, No. 1, pp. 111–127.

[29] Collins, M.—Schapire, R. E.—Singer, Y.: Logistic Regression, AdaBoost and Bregman Distances. Proceedings of the Thirteenth Annual Conference on Computational Learning Theory, San Francisco, USA, 2000, pp. 158–169.

[30] He, Z.—Yu, W.: Stable Feature Selection for Biomarker Discovery. Computational Biology and Chemistry, Vol. 34, 2010, No. 4, pp. 215–225.

[31] Awada, W.—Khoshgoftaar, T.—Dittman, D.—Wald, R.—Napolitano, A.: A Review of the Stability of Feature Selection Techniques for Bioinformatics Data. 2012 IEEE 13th International Conference on Information Reuse and Integration (IRI), August 2012, pp. 356–363.

[32] Drotar, P.—Gazda, J.—Smekal, Z.: An Experimental Comparison of Feature Selection Methods on Two-Class Biomedical Datasets. Computers in Biology and Medicine, Vol. 66, 2015, pp. 1–10. Available on: `http://www.sciencedirect.com/science/article/pii/S0010482515002917`.

[33] Kuncheva, L. I.: A Stability Index for Feature Selection. Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications (AIAP '07), Anaheim, CA, USA. ACTA Press, 2007, pp. 390–395.

[34] Burczynski, M. E.—Peterson, R. L.—Twine, N. C.—Zuberek, K. A.—Brodeur, B. J.—Casciotti, L.—Maganti, V.—Reddy, P. S.—Strahs, A.—Immermann, F.—Spinelli, W.—Schwertschlag, U.—Slager, A. M.—Cotreau, M. M.—Dorner, A. J.: Molecular Classification of Crohn's Disease and Ulcerative Colitis Patients Using Transcriptional Profiles in Peripheral Blood Mononuclear Cells. The Journal of Molecular Diagnostics, Vol. 8, 2006, No. 1, pp. 51–61.

[35] Chowdary, D.—Lathrop, J.—Skelton, J.—Curtin, K.—Briggs, K.—Zhang, Y.—Yu, J.—Wang, Y.—Mazumder, A.: Prognostic Gene Expression Signatures Can Be Measured in Tissues Collected in RNAlater Preservative. The Journal of Molecular Diagnostics, Vol. 8, 2006, No. 1, pp. 31–39.

[36] Golub, T. R.—Slonim, D. K.—Tamayo, P.—Huard, C.—Gaasenbeek, M.—Mesirov, J. P.—Coller, H.—Loh, M. L.—Downing, J. R.—Caligiuri, M. A.—Bloomfield, C. D.—Lander, E. S.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science, Vol. 286, 1999, No. 5439, pp. 531–537.

[37] Gordon, G. J. G.—Jensen, R. V. R.—Hsiao, L.-L. L.—Gullans, S. R. S.—Blumenstock, J. E. J.—Ramaswamy, S. S.—Richards, W. G. W.—Sugarbaker, D. J. D.—Bueno, R. R.: Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gene Expression Ratios in Lung Cancer and Mesothelioma. Cancer Research, Vol. 62, 2002, No. 17, pp. 4963–4967.

[38] Tian, E.—Zhan, F.—Walker, R.—Rasmussen, E.—Ma, Y.—Barlogie, B.—Shaughnessy, J. D. Jr.: The Role of the Wnt-Signaling Antagonist DKK1 in the Development of Osteolytic Lesions in Multiple Myeloma. New England Journal of Medicine, Vol. 349, 2003, No. 26, pp. 2483–2494.

[39] Pomeroy, S. L.—Tamayo, P.—Gaasenbeek, M.—Sturla, L. M.—Angelo, M.—McLaughlin, M. E.—Kim, J. Y. H.—Goumnerova, L. C.—Black, P. M.—Lau, C.—Allen, J. C.—Zagzag, D.—Olson, J. M.—Curran, T.—Wetmore, C.—Biegel, J. A.—Poggio, T.—Mukherjee,

S.—Rifkin, R.—Califano, A.—Stolovitzky, G.—Louis, D. N.—Mesirov, J. P.—Lander, E. S.—Golub, T. R.: Prediction of Central Nervous System Embryonal Tumour Outcome Based on Gene Expression. Nature, Vol. 415, 2002, No. 6870, pp. 436–442.

[40] Drotar, P.—Smekal, Z.: Stability of Feature Selection Algorithms and Its Influence on Prediction Accuracy in Biomedical Datasets. 2014 IEEE Region 10 Conference (TENCON 2014), Bangkog, Thailand, 2014, pp. 1–5.

Peter Drotár received his M.Sc. from Technical University of Košice in 2007 and Ph.D. from the same university in 2010. From 2010 to 2012 he worked as a scientist for Honeywell Advanced Technology Europe and from 2012 to 2015 as a postdoctoral researcher at Brno University of Technology, Czech Republic. Currently he is Associate Professor at the Department of Computers and Informatics, Technical University of Košice. His research interests include data analysis, biomedical signal processing and decision support systems.

Slavomír Šimoňák received his M.Sc. degree in computer science in 1998 and his Ph.D. degree in computer tools and systems in 2004, both from the Technical University of Košice, Slovakia. He is currently Assistant Professor at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, Slovakia. His research interests include formal methods integration and application, communication protocols, algorithms, and data structures.

Emília Pietriková is Assistant Professor at the Department of Computers and Informatics, Technical University of Košice, Slovakia. She received her M.Sc. in 2010 and her Ph.D. in 2014 in informatics from Technical University of Košice. In 2010 she spent 1 month at the Department of Telematics at Norwegian University of Science and Technology, Norway. In 2011 she spent 1 semester at the Department of Computer Architecture at University of Málaga, Spain. The subject of her research is abstraction and generation in programming languages, and quality of education.

Martin Chovanec received his Master's degree in informatics in 2005 at the Faculty of Electrical Engineering and Informatics, Technical University of Košice. 2008 he received his Ph.D. degree at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice and his scientific research was focused on network security and encryption algorithms. Currently, he is the director of the Computer Technology Centre at the Technical University of Košice.

Eva Chovancová graduated (Ing.) at the Department of Computers and Informatics at the Faculty of Electrical Engineering and Informatics of the Technical University in Košice, Slovakia in 2009. She defended her Ph.D. thesis in the field of computers and computer systems in 2012; her thesis title was "Specialized processor for computing acceleration in the field of computer vision". Since 2012, she has been working as Assistant Professor at the Department of Computers and Informatics. Her scientific research is focused on the multicore computer architectures and security.

Norbert Ádám graduated (M.Sc.) with distinction at the Department of Computers and Informatics at the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. He defended his Ph.D. in the field of computers and computer systems in 2007; his thesis title was "Contribution to the simulation of feed-forward neural networks on parallel computer architectures". Since 2006 he is working as Assistant Professor at the Department of Computers and Informatics. Since 2008 he is the Head of the Computer Architectures and Security Laboratory at the Department of Computers and Informatics. His scientific research is focused on parallel computers architectures.

Csaba Szabó graduated with distinction at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics (FEI) at Technical University of Košice in 2003. He obtained his Ph.D. in Program- and Information Systems at the FEI at Technical University of Košice in 2007. Since 2006 he is affiliated with the Department of Computers and Informatics, FEI, Technical University of Košice. Currently he is involved in research in the field of behavioral description of software, information systems and web services, software and test evolution, and testing and evaluation of software.

Anton Baláž received his Master's degree in informatics in 2004 from Faculty of Electrical Engineering and Informatics, Technical University of Košice. In 2008 he received Ph.D. in the area of computer security. Since 2007 he is working as Assistant Professor at the Technical University of Košice.

Miroslav Biňas works as Assistant Professor at the Department of Computers and Informatics, Technical University of Košice, Slovakia. He received his M.Sc. in 2004 in information technology and informatics and his Ph.D. in 2009 in informatics from Technical University of Košice. In 2007 he was an intern at the University of Zaragoza, Spain. He has participated in several national and international scientific projects, moreover, he leads key programming courses at the DCI TUKE. The subject of his research is quality of education and knowledge transfer, mostly focused on the area of modern programming technologies as well as current trends in computer science, especially development of software for smart technologies and game-based development.