IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 29, 2021, accepted February 7, 2021, date of publication February 11, 2021, date of current version February 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3058428

Empirical Comparison of the Feature Evaluation Methods Based on Statistical Measures

ADAM ŁYSIAK AND MIROSŁAW SZMAJDA
Faculty of Electrical Engineering, Automatic Control and Informatics, Opole University of Technology, 45-758 Opole, Poland
Corresponding author: Adam Łysiak (a.lysiak@doktorant.po.edu.pl)

ABSTRACT One of the most important classification problems is selecting proper features, i.e. features that describe the classified object in the most straightforward way possible. Then, one of the biggest challenges of the feature selection is the evaluation of the feature's quality. There is a plethora of feature evaluation methods in the literature. This paper presents the results of a comparison between nine selected feature evaluation methods, both existing in literature and newly defined. To make a comparison, features from ten various sets were evaluated by every method. Then, from every feature set, best subset (according to each method) was chosen. Those subsets then were used to train a set of classifiers (including decision trees and forests, linear discriminant analysis, naive Bayes, support vector machines, k nearest neighbors and an artificial neural network). The maximum accuracy of those classifiers, as well as the standard deviation between their accuracies, were used as a quality measures of each particular method. Furthermore, it was determined, which method is the most universal in terms of the data set, i.e. for which method, obtained accuracies were dependent on the feature set the least. Finally, computation time of each method was compared. Results indicated that for applications with limited computational power, method based on the average overlap between feature's values seem best suited. It led to high accuracies and proved to be fast to compute. However, if the data set is known to be normally distributed, method based on two-sample *t*-test may be preferable.

INDEX TERMS Classification, dimensionality reduction, distribution overlap, feature evaluation, feature extraction, feature selection, filter methods, machine learning, overlap coefficient, pattern recognition.

I. INTRODUCTION

In the current, Big Data driven research, the quality of the data is often neglected in favor of the quantity [1]. The data quality estimation, however, is not as straightforward as it may seem. There are lots of evaluation methods and the vastness of the literature devoted to the feature extraction and selection makes the matter quite complicated.

Features are evaluated depending on their application, however, most often the feature is considered ''good'' if it makes the classification task easier. In other words, features that are characteristic for considered class, and for this class only, are desirable. Every object, real or abstract, can be described in infinitely many ways, thus generating infinitely many features. Therefore, specific features are often defined for specific tasks. For example, feature of ''having feathers'' would be considered useful to distinguish a parrot from a cat, but not from a pigeon. The main problem is: how to quantitatively represent the ''goodness'' of a feature, i.e. how to evaluate it.

Different feature evaluation methods are often divided into three main categories: filter, wrapper and embedded methods [2]. Wrapper and embedded methods evaluate feature (or a set of features) based on their effectiveness as an input for some selected classification algorithm. Filter methods, on the other hand, evaluate feature quality independently of the classification algorithm. They can be further grouped based on their definition; there can be filters based on the correlation measures, information theory, probability distributions etc. Since the last group is easiest to interpret and implement, this paper provides a summary of the research devoted to it. That is, the comparison of some existing and some newly proposed filter methods based on the probability distributions.

It was assumed that the features are continuous, i.e. their values are real numbers. The simplest methods utilize basic statistical measures, while more sophisticated ones require probability density estimates. All methods were used to evaluate features distinguishing two classes. The evaluation itself

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson.

was conducted using discrete features, i.e. two sets of values representing those classes.

Features used in this study came from ten data sets: three describing vibroarthrographic signals [3], [4], two describing breast cancer [5], [6], one describing swarm behavior [6], one devoted to gene expression in various tumors [6], [7], one composed of chemical characteristics of some wine cultivars [6], [8], and two concerning voice processing [6], [9], [10]. This ensured obtained results to be most possibly universal.

There are several filter comparisons present in the literature already. However, most studies are devoted to the feature selection problem rather than feature evaluation itself (for example, see: [11]–[15]). That is, for a given feature set, a subset of arbitrarily big size can be selected [11]. In such approach, the feature evaluation step is also present, however, different feature evaluation methods are not subjected to comparison.

There is a plethora of different feature selection algorithms present in literature. For example, in [16], authors proposed an algorithm based on particle swarm optimization rules, significantly reducing feature space in identification of potential clinical syndromes of Hepatocellular carcinoma. For similar task, different algorithm was later proposed, based on non-negative matrix factorization [17].

In [18] author proposed a feature selection algorithm based on global sensitivity analysis, improving prediction performance obtained by state-of-the-art principal component analysis dimensionality reduction method.

In [19] authors studied a particle swarm optimization based, unsupervised feature selection method. To achieve this, they used filter methods based on information theory, such as average mutual information.

In this research, however, most trivial feature selection algorithm was implemented: selecting one best feature to differentiate one class pair. That is, for every pair of classes, one feature, evaluated to be the best, was selected. This resulted in $\binom{n}{2}$ features for each data set, where n is number of classes. For example, for 5 classes, there would be 10 features selected: one to distinguish classes 1-2, one for classes 1-3, one for 1-4, 1-5, 2-3, 2-4, 2-5, 3-4, 3-5, and one for classes 4-5. Different feature selection algorithms would greatly increase final classification results. However, classification accuracy was not focus of this study. Implementing more sophisticated algorithms would probably lower classification variance for bigger feature sets, since including more features in classification task would probably enhance accuracy regardless of the classifier used. Because variance of classification between different classifiers was one of parameters used to compare feature evaluation methods, described simple algorithm was utilized.

In summary, every method was used to evaluate every feature in given feature set. Best features were then selected and used to train a set of classification algorithms. Their accuracies allowed to decide on the qualities of different evaluation methods. Computation time of every method was additionally measured. Presented methodology allowed to answer four main questions:

1) Which method is the most precise, leading to the highest classification accuracy?
2) In this research, classification accuracy depended on three elements: the original feature set, the feature subset (chosen using a method) and the algorithm itself. The second question then is: which method is the most robust? That is, for which method the classification accuracy depends on the algorithm the least?
3) Which method is most universal? That is, for which method the classification accuracy depends on the original data set the least?
4) And which method is the least computationally expensive?

II. COMPARED METHODS

All of the compared methods are functions of two vectors, i.e. values of the first and the second class. The outputs of those functions are coefficients taking values from 0 (when the feature is perfect, or the most informative possibly) to 1 (when the feature cannot be less informative). Therefore, features with lower values are preferred to those, which values are higher.

There were nine methods compared in this research, giving nine different coefficients: two based on values' ranges, two with additional information about the spreads of values, two based on p-values of statistical tests, normalized Kullback-Liebler divergence, overlap coefficient and the Bhattacharyya coefficient. Brief description of each was given below.

A. THE $coef_{MEAN}$ COEFFICIENT

The average overlap between values' ranges. Visualization of this method was given in Figure 1*a* and 1*b*. Grayed areas visualize the range of the values for class I (Figure 1*a*) and class II (Figure 1*b*). The overlap is defined as the number of values in one class that overlap with another class, divided by the number of observations. This definition generates two values (the overlap between class I and II and the overlap between class II and I). Their average will be called $coef_{MEAN}$ in the rest of this paper. The coefficient can be defined as:

$$coef_{MEAN} = \frac{1}{2n}\sum_{k=1}^{n}[P_1(k) + P_2(k)], \text{ where} \tag{1}$$

$$P_1(k) = \begin{cases} 1 & min(S_2) \leq S_1(k) \leq max(S_2) \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

$$P_2(k) = \begin{cases} 1 & min(S_1) \leq S_2(k) \leq max(S_1) \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

S_1 is the class I sample set and S_2 is the class II sample set. The $coef_{MEAN}$ coefficient indicates what average fraction of the values is unclassifiable. As it can be seen on Figure 1, the coefficient is highly susceptible to outliers - even one extreme measurement can highly affect the coefficient's

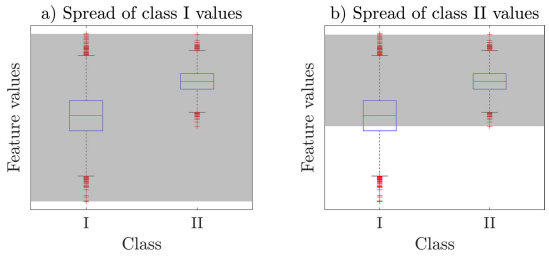

FIGURE 1. Visualization of the two spreads of values. Red line indicates the median of the feature values for a given class, blue box indicates the interquartile range, whiskers indicate the rest of the values, which are not considered outliers. Outliers are indicated by red crosses. Plots a) and b) show the spread of the first and the second class values, respectively.

value. Authors of this paper could not find definition of such coefficient in the literature, but given its rudimentary character, it was, most possibly, used before.

B. THE $coef_{MAX}$ COEFFICIENT

The greater value of two overlaps will be called $coef_{MAX}$ in the rest of the paper. This coefficient indicates, again, what fraction of the values is unclassifiable. It is defined as

$$coef_{MAX} = max\left(\frac{1}{n}\sum_{k=1}^{n} P_1(k), \frac{1}{n}\sum_{k=1}^{n} P_2(k)\right), \tag{4}$$

where all of the values are defined same as in Equation 1. Using the greater value, instead of the mean, makes it much more rigorous. For example, if one range of values is contained in the other one, this coefficient will take a maximum value. Additionally, it is also very susceptible to outliers.

This coefficient is even simpler than $coef_{MEAN}$. Hence, most probably, it was defined somewhere before. However, authors could not find any references, in which it would be used.

C. THE $coef_{MOR}$ COEFFICIENT

The distance between medians to overall spread ratio (DBM to OVS ratio) [20]–[22] is another measure which does not need probability density estimation. The visualization of the DMB and the OVS values was given in Figure 2. In the rest of this paper, the coefficient will be called $coef_{MOR}$. It is defined as:

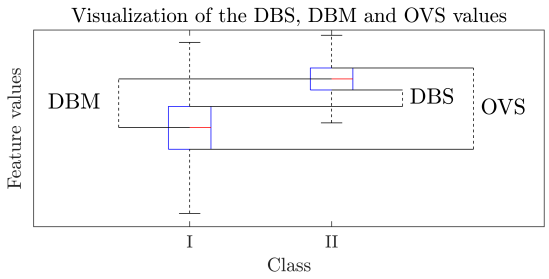

FIGURE 2. Visual representation of the distance between medians (DMB), the overall spread (OVS) and the distance between spreads (DBS) values.

$$coef_{MOR} = 1 - \frac{|m_1 - m_2|}{max(Q_{U1}, Q_{U2}) - min(Q_{L1}, Q_{L2})}, \tag{5}$$

where m_1 and m_2 are medians of the specific classes, Q_{U1} and Q_{U2} are values of their third quartiles, and Q_{L1} and Q_{L2} are values of their first quartiles.

The $coef_{MOR}$ coefficient is insusceptible to outliers, as only interquartile range of vales is considered. When the medians are identical, the coefficient is equal to one. Originally, it was defined without the difference, taking maximal value of one, when the feature was distinguishing the classes in the best way possible. However, the definition with the difference (as in Equation 5) will be consistent with the rest of the compared coefficients.

D. THE $coef_{SOR}$ COEFFICIENT

The distance between spreads to overall spread ratio (DBS to OVS ratio) is similar to the previous coefficient. It is defined as:

$$coef_{SOR} = 1 - \frac{|DBS|}{max(Q_{U1}, Q_{U2}) - min(Q_{L1}, Q_{L2})}, \tag{6}$$

$$DBS = \begin{cases} Q_{L1} - Q_{U2} & \text{if } Q_{U1} > Q_{U2} \text{ and } Q_{L1} > Q_{L2} \\ Q_{U1} - Q_{L2} & \text{if } Q_{U1} < Q_{U2} \text{ and } Q_{L1} < Q_{L2} \\ \frac{Q_{L1} - Q_{U2} + Q_{U1} - Q_{L2}}{2} & \text{otherwise} \end{cases} \tag{7}$$

where DBS is the distance between spreads, defined in Equation 7 and visualized in Figure 2. Note, that when one distribution is contained in another one, the distance between spreads is the mean value of two distances. The coefficient will be called $coef_{SOR}$ in the rest of the paper.

Authors could not find any references, in which such coefficient was defined.

E. THE $coef_{TT}$ COEFFICIENT

The p-value of the two-sample t-test [23], will be called in the rest of this paper $coef_{TT}$. The student t-test was widely used as a feature evaluation tool [24]–[31]. It is used to test the null hypothesis that two sets of values come from the same distribution. The test statistic is given by [23]:

$$t = \frac{\bar{x_1} - \bar{x_2}}{\sqrt{\frac{\sigma_{x_1}^2}{n_1} + \frac{\sigma_{x_2}^2}{n_2}}}, \tag{8}$$

where x_1 and x_2 are the means of the feature values, σ_1 and σ_2 are their standard deviations, and n_1 and n_2 are the sizes of the classes.

The p-value is the probability of observing t values equal or more extreme than the observed one. Therefore, the p-value is lower, when two vectors are more statistically different.

The biggest disadvantage of this method is that, in order for the test to be interpretable, the values in compared vectors should follow a normal distribution [23]. Despite this assumption, estimation of the probability density is not required to calculate $coef_{TT}$.

F. THE $coef_{KS}$ COEFFICIENT

The p-value of the Kolmogorov–Smirnov (K-S) test [32] will be called $coef_{KS}$ in the rest of this paper. It also was widely used as a feature evaluation method [33]–[35], since it eliminates the data normality requirement. The K-S test evaluates the biggest difference in (empirical) cumulative distribution functions (cdf) of two classes. Visualization of the difference was given in Figure 3.

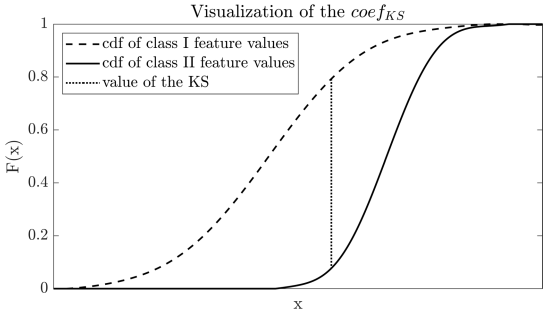

FIGURE 3. Visual representation of the Kolmogorov-Smirnov statistic. The dashed line corresponds to the cumulative distribution function $\hat{F}$ from Equation 9, the solid line corresponds to the $\hat{G}$, and the dotted line shows the d statistic.

The test statistic is given by [36]:

$$d = \max_X |\hat{F}(x) - \hat{G}(x)|, \tag{9}$$

where d is the statistic, X is the feature values domain, and $\hat{F}$ and $\hat{G}$ are empirical cdfs. As in $coef_{TT}$, the p-value is the probability of observing d values equal or more extreme than the observed one.

This is the last coefficient, for which the explicit knowledge of the probability density is not required.

G. THE $coef_{KLD}$ COEFFICIENT

The Kullback–Leibler divergence (KLD) is another measure often used as a feature evaluation tool [24], [37]–[40]. It is defined as [41]:

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log \left(\frac{P(x)}{Q(x)} \right), \tag{10}$$

where P and Q are the probability density functions of two classes, and X is the feature values domain. The visualization of $D_{KL}(P||Q)$ and $D_{KL}(Q||P)$ was given in Figure 4.

Note, that the fraction makes the KLD unbounded. It is also non-symmetric (meaning that $D_{KL}(P||Q) \neq D_{KL}(Q||P)$). To overcome these drawbacks, the definition of $coef_{KLD}$ coefficient used in this comparison was slightly modified:

$$coef_{KLD} = \exp\left(-\left(D_{KL}(P||Q) + D_{KL}(Q||P)\right)\right), \tag{11}$$

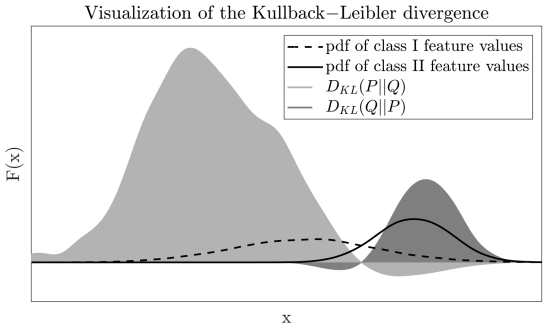

FIGURE 4. Visual representation of the Kullback-Leibler divergence. The dashed line corresponds to the probability density function P from Equation 10, the solid line corresponds to the Q from Equation 10. The divergences areas were marked with dark and light gray, for $D_{KL}(P||Q)$ and $D_{KL}(Q||P)$ respectively.

making it symmetric and bounded in the same range as the other coefficients.

To obtain $coef_{KLD}$, the information about the probability density is required. To estimate it, the Kernel Density Estimator (KDE) was used, described in more detail later.

H. THE $coef_{OVL}$ COEFFICIENT

The overlap coefficient OVL, proposed in [42], is defined as the intersection area of two probability density functions:

$$coef_{OVL} = \sum_{x \in X} \min\left(P(x), Q(x)\right), \tag{12}$$

where P and Q are the probability density functions of two classes, and X is the feature values domain. The visualization of $coef_{OVL}$ was given in Figure 5.

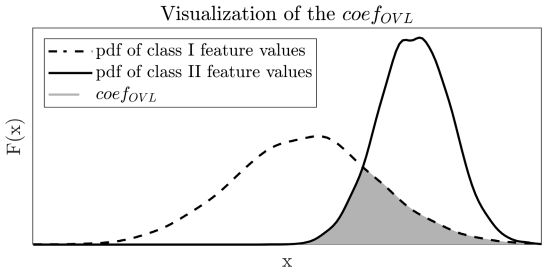

FIGURE 5. Visual representation of $coef_{OVL}$. The dashed line corresponds to the probability density function P from Equation 12, the solid line corresponds to the Q from Equation 12. The overlap between them was grayed.

This coefficient was often used [43]–[47] in both, continuous and discrete data analysis. In the literature, it can be also encountered under the names of histogram overlap coefficient or Jaccard index (as it was proposed to measure overlap between sets in [48]). As in $coef_{KLD}$, the KDE was used to obtain the probability density functions.

I. THE $coef_B$ COEFFICIENT

The Bhattacharyya coefficient is another measure widely described in the literature [49]–[55]. It is defined as:

$$coef_B = \sum_{x \in X} \sqrt{P(x) \cdot Q(x)}, \tag{13}$$

where P and Q are the probability density functions of two classes, and X is the feature values domain. The visualization of $coef_B$ was given in Figure 6.

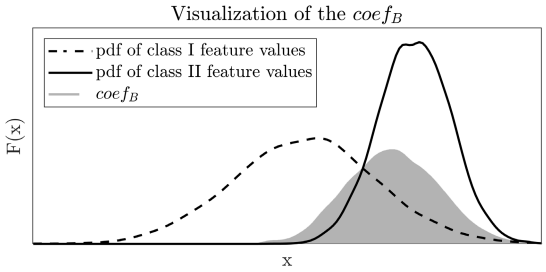

FIGURE 6. Visual representation of the Bhattacharyya coefficient. The dashed line corresponds to the probability density function *P* from Equation 13, the solid line corresponds to the *Q* from Equation 13. The Bhattacharyya coefficient was grayed.

Similarly to two previous coefficients, to obtain $coef_B$, the KDE was used.

III. MATERIALS AND METHODS

A. SUMMARY OF THE RESEARCH METHODOLOGY

As presented, there are lots of different feature evaluation methods present in the literature. However, it is not clear, which method is the most precise. That is, which results in the highest accuracy of the classifier constructed with features suggested by it. This paper presents an attempt to answer this question.

There were ten feature sets used in this research. In every set, all the features were evaluated by the compared coefficients. So, for every feature in a set, there was a 9-element vector of specific coefficients values. Then, for every coefficient, $\binom{n}{2}$ best features were selected (where n is number of classes); one best feature to differentiate one class pair. That step was graphically presented in the upper part of Figure 7.

Subsequently to the best features subset selection, its values were normalized in 0 to 1 range. This step ensured that dispersion of data will not negatively affect final classification accuracy.

After that, the best features were used to train 11 different classification algorithms. Therefore, for each feature set there were 99 classifiers constructed and trained (9 coefficients times 11 classifiers). That part was presented in the lower part of Figure 7. The accuracy of the classification algorithm strongly depends on the split of the data into the training, testing and validation subsets [56]. To compensate possible biases, the accuracies were obtained as a mean values of the 512 classifiers of a given type, with different random splits.

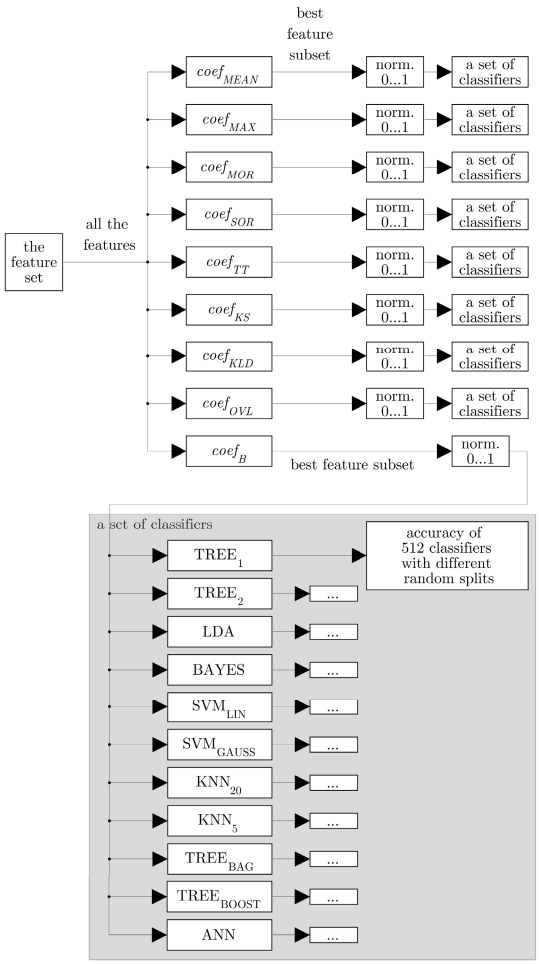

FIGURE 7. Flowchart of the methodology. Note, that in this research it was executed for ten different feature sets. The block before the classifiers indicates step of normalizing the data. Particular classifiers were described in subsection III-C.

The research was concluded with the computational cost study. Each coefficient was evaluated with an exemplary feature for 100 000 times.

B. FEATURE SETS USED

To ensure universality of obtained results, ten different feature sets were used in this research: three sets of vibroarthrographic (VAG) signals frequency characteristics [3], [4], two sets of breast cancer features [5], [6], a feature set describing grouping in a swarm of boids (bird androids) [6], a feature set of gene expression for various types of tumor [6], [7], a set of features describing chemical characteristics of different types of wine [6], [8], a set of features enabling classification of Parkinson's Disease patients [6], [9], and feature set of voice rehabilitation [6], [10].

In the rest of the paper, term ''instance'' will be used to indicate one case of an object described by a feature. For example, 184 instances in VAG feature sets mean that 184 knee joints were examined and diagnosed. Exact feature sets will be described in more details below.

1) VIBROARTHROGRAPHY DATA SETS

Vibroarthrography is a noninvasive knee-joint examination, in which the joint generates vibrations, while performing flexion/extension motions [3]. Signals used in this research were obtained and analyzed for the first time in [3]. They were diagnosed by the radiologists into five classes: the control group (healthy knee joint), three stages of chondromalacia patellae, and the osteoarthritis group. There are 184 signals in total.

The frequency analysis of the VAG signal is fruitful and easily interpretable [4], so it was used in this research to generate three feature sets. Those sets were quite large and diverse, i.e. containing informative features as well as uninformative ones. Specifically, there were:

- The frequency range map set [4], composed of about 470 000 000 features. The features were defined as every possible frequency range of the Discrete Fourier Transform (DFT). In the rest of this paper, it will be called *VAG FRM* .
- The Discrete Fourier Transform (DFT) of the VAG signals, which constituted a set of about 30 000 features. In the rest of this paper, it will be called *VAG DFT* .
- The squared DTF of the VAG signals, also constituted a set of about 30 000 features. In the rest of this paper, it will be called $VAG\ DFT^2$.

2) BREAST CANCER WISCONSIN DATA SETS

Two Breast Cancer Wisconsin data sets were also used in this research [5], [6]. Features describing cell nuclei were computed from a fine-needle aspiration breast mass image. They differentiate between malignant and benign type of cancer. Those are rather small sets of data compared to the previous ones. However, they are widely used in classification and feature-extraction literature, so it would be beneficial to include them in the research. Specific sets of features are:

- Diagnostic data set, composed of 30 features with 569 instances. It will be called *WDBC* in the rest of the paper.
- Prognostic data set, made out of 33 features for 198 instances. In the rest of the paper, it will be called *WPBC* .

3) SWARM BEHAVIOR DATA SET

Swarm Behavior [6] is a set used to classify a swarm of boids (bird androids) into ''grouped'' or ''non-grouped'' classes, deepening understanding of human perception of flocking behavior. The set is composed of 2400 features, for a set of 24016 instances. In the rest of the paper, it will be called *SB* .

4) GENE EXPRESSION DATA SET

Gene Expression data set [6], [7] is composed of random gene expression extractions of five types of tumors: breast, kidney, colon, lung and prostate. It is one of the biggest data sets included in this research, containing over 20000 features for about 800 instances. In the rest of this paper, it will be called *GEC* .

5) WINE DATA SET

Wine [6], [8] is the smallest data set used in this research, being composed of only 13 features with 178 instances. Features are defined as chemical characteristics of three different wine cultivars. It will be called *WDS* in the rest of this paper.

6) PARKINSON's DISEASE DATA SET

Parkinson's Disease [6], [9] is a data set of various speech processing features defined for Parkinson's Disease patients vocalizing vowel ''a''. It contains 753 features of 756 individuals. This feature set will be called *PDC* in the rest of this paper.

7) VOICE REHABILITATION DATA SET

In Lee Silverman Voice Treatment (LSVT) data set [6], [10], as in previous one, speech processing features were included. However, they were used to classify phonations as ''acceptable'' or ''unacceptable'' after the LSVT treatment. This set consist of 311 features of 126 individuals. In the rest of this paper, it will be abbreviated as *LSVT* .

8) SUMMARY OF THE FEATURE SETS

Feature sets used in this study were chosen to be representative of big as well as small data sets, in terms of both number of features and instances. Their summary was presented in table 1. Class distribution column contains numbers of instances in particular classes. Number of selected features is $\binom{n}{2}$, where n is number of classes.

To check if the features' distributions of particular sets and classes are normal, a bunch of normality tests was performed: the Shapiro-Wilk test, the Lilliefors' test, the Anderson-Darling test and the Jarque-Bera test [57]. Almost all distributions proved to be other than normal, with exception for one class in *WPBC* , most classes in *GEC* and all classes in *WDS* . For more details see Appendix A.

C. CLASSIFIERS USED

To somehow make the comparison independent of the classifier used, eleven different classifiers were trained with data normalized in 0 to 1 range (as in Figure 7):

1) Decision tree with maximum of $2n$ splits (where n is a number of classes); in the rest of paper abbreviated as $TREE_1$,
2) Decision tree with maximum of n splits (where n is a number of classes); in the rest of paper abbreviated as $TREE_2$,
3) Discriminant analysis classifier; in the rest of paper abbreviated as *LDA*,
4) Naive Bayes classifier; in the rest of paper abbreviated as *BAYES*,
5) Support vector machine with linear kernel; in the rest of paper abbreviated as SVM_{LIN},

TABLE 1. Summary of feature sets used. Class distribution column contains number of instances in particular classes.

no.	name	abbreviated name	number of all features	number of instances	number of classes	class distribution	number of selected features	references
1	Vibroarthrography Frequency Range Map	*VAG FRM*	471 843 840	184	5	66/26/30/36/26	10	[3], [4]
2	Vibroarthrography Discrete Fourier Transform	*VAG DFT*	30 720	184	5	66/26/30/36/26	10	[3], [4]
3	Vibroarthrography Discrete Fourier Transform squared	$VAG\ DFT^2$	30 720	184	5	66/26/30/36/26	10	[3], [4]
4	Breast Cancer Wisconsin (Diagnostic)	*WDBC*	30	569	2	357/212	1	[5], [6]
5	Breast Cancer Wisconsin (Prognostic)	*WPBC*	33	198	2	151/47	1	[5], [6]
6	Swarm Behavior (Grouped)	*SB*	2 400	24 016	2	15 010/9 006	1	[6]
7	Gene Expression Cancer RNA-Seq	*GEC*	50 531	801	5	300/78/146/141/136	10	[6], [7]
8	Wine Data Set	*WDS*	13	178	3	59/71/48	3	[6], [8]
9	Parkinson's Disease Classification	*PDC*	753	756	2	192/564	1	[6], [9]
10	LSVT Voice Rehabilitation	*LSVT*	311	126	2	42/84	1	[6], [10]

6) Support vector machine with gaussian kernel; in the rest of paper abbreviated as SVM_{GAUSS},
7) k-nearest neighbors algorithm with $k = 20$; in the rest of paper abbreviated as KNN_{20},
8) k-nearest neighbors algorithm with $k = 5$; in the rest of paper abbreviated as KNN_5,
9) Decision forest using bagging algorithm, with maximum of $k/10$ splits (where k is number of instances), constructed on 64 learners; in the rest of paper abbreviated as $TREE_{BAG}$,
10) Decision forest using boosting algorithm, with maximum of $k/10$ splits (where k is number of instances), constructed on 64 learners; in the rest of paper abbreviated as $TREE_{BOOST}$. Random undersampling boosting (RUSBoost [58]) was used, as it is suitable for both, binary and multiclass classification problems,
11) Neural network classifier with *n tansig* neurons in the hidden layer (where n is a number of classes); in the rest of paper abbreviated as *ANN*.

To avoid overfitting of the classification model, so the situation in which the model loses its generalization abilities, usually the dataset is divided into training, validation and testing subsets [56]. The final classification accuracy is then based on the testing subset, ensuring that new data will be classified with similar accuracy. The division of the original set greatly impacts the accuracy (the less data, the greater the impact). In order to avoid potential biases deriving from this fact, every classifier algorithm was implemented 512 times, with different, random divisions each time. The results are based on the average accuracies of those 512 classifiers.

D. PROBABILITY DENSITY ESTIMATION

1) THE KERNEL DENSITY ESTIMATOR

To obtain the density of the probability required to calculate $coef_{KLD}$, $coef_{OVL}$ and $coef_B$ coefficients, the Kernel Density Estimator (KDE) was used. This concept consists in applying the appropriate smoothing kernel to each of the observations and then adding the obtained functions. The visualization of this approach was shown in Figure 8.

Smoothing functions are normal distributions with an average value equal to the value of a given observation and a standard deviation equal to h. The h value is called the smoothing parameter and allows to control the shape of the

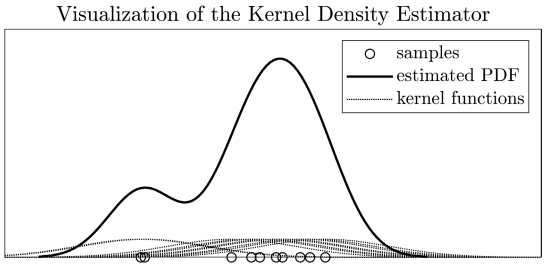

FIGURE 8. Visualization of the Kernel Density Estimator. To all of the measured values (indicated by circles), the kernel function was applied (dotted lines). The estimation (solid line) is the sum of the kernel functions.

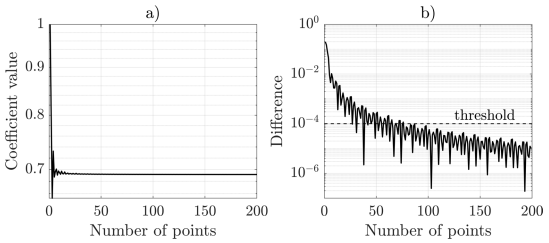

FIGURE 9. Visualization of the KDE precision study. Plot a) shows coefficient values as a function of the number of KDE points, plot b) shows the absolute value of the difference between consecutive points in a). Dashed line in b) indicate the threshold, i.e. 10^{-4}. Note, that the b) plot has logarithmic y-axis scale.

estimated density. The KDE is defined as:

$$\hat{f}(y) = \frac{1}{n}\sum_{i=1}^{n} w(y_i, h), \tag{14}$$

where y is the value, for which the probability density is estimated, n is the sample size, w is the kernel function, y_i is the value of the i-th observation and h is the soothing parameter (variance of the kernel function), given by [59]:

$$h = \left(\frac{4}{3n}\right)^{\frac{1}{5}} \cdot \sigma, \tag{15}$$

where σ is the standard deviation of the set.

2) ACCURACY OF THE ESTIMATION

To obtain particular coefficients, the KDE should be calculated for specific number of points. The more points is there to calculate, the more time is needed to calculate the coefficient's value (for exact correlation between the number of points and calculation time, see Appendix B). In order to determine specific number of points necessary for precise coefficient calculation, simple study was conducted. Each coefficient was evaluated for x-point KDE, with growing x. When the difference in coefficient's value between consecutive x's was smaller than the threshold, then the specific x was chosen as a number of points in KDE. This was visualized in Figure 9. The threshold was set to 10^{-4}. It was chosen as such, since higher accuracy would raise the computational cost significantly, and lower would negatively affect the interpretability of the results.

To make it more robust, this process was repeated for the set of 30 000 features, and the highest x value was selected as a number of points in coefficient calculation process. As a result, the x value was equal to:

1) 150 for $coef_{KLD}$,
2) 100 for $coef_{OVL}$,
3) 50 for $coef_{B}$.

Different number of estimated points would highly affect computational time, since those values are linearly correlated (ρ>0.99 for all the coefficients, see Appendix B).

E. COEFFICIENT COMPUTATION TIME

To compare different coefficients in terms of the computational complexity, every coefficient was evaluated for a 100 000 times with a representative feature. The mean values of time needed to calculate them were utilized to conduct the comparison. To minimize the effect of specific machine, on which the coefficients were being calculated, the highest value was used to normalize the results. That is, every time value was divided by the maximum time.

F. COEFFICIENTS' RANKINGS

To make the comparison between coefficients easier, values of maximal accuracy and standard deviation of accuracy were ranked. Ranks took values from 1 (for the best value) to 9 (for the worst value). In the ranking process, maximal obtained accuracy was preferred to be higher, while standard deviation and time were preferred to be lower. There were two types of ranks defined:

- R_1 which is rank of the average value of maximum accuracy, i.e. rank was obtained after averaging the maximal accuracies of all classifiers. For detailed values, on which this ranking was based, see "max" rows in Tables 4 to 13 included in Appendix C. This rank allows to compare precise values of the accuracy, without taking to account differences between classifiers of different feature sets.
- R_2 which is rank of the average value of maximum accuracy's ranks, i.e. rank was obtained after ranking the maximal accuracies. Consequently, this rank allows to compare the accuracies while taking into account different classifiers. That means, that coefficient with the best R_2 obtained highest accuracies most frequently.

Big differences between R_1 and R_2 could indicate that accuracies obtained using specific coefficient highly depend on a classified data set. Consequently, that could indicate a need for additional data analysis step before evaluating features using some specific coefficient, since one coefficient could lead to high accuracies for one type of data, and low accuracies for another type of data.

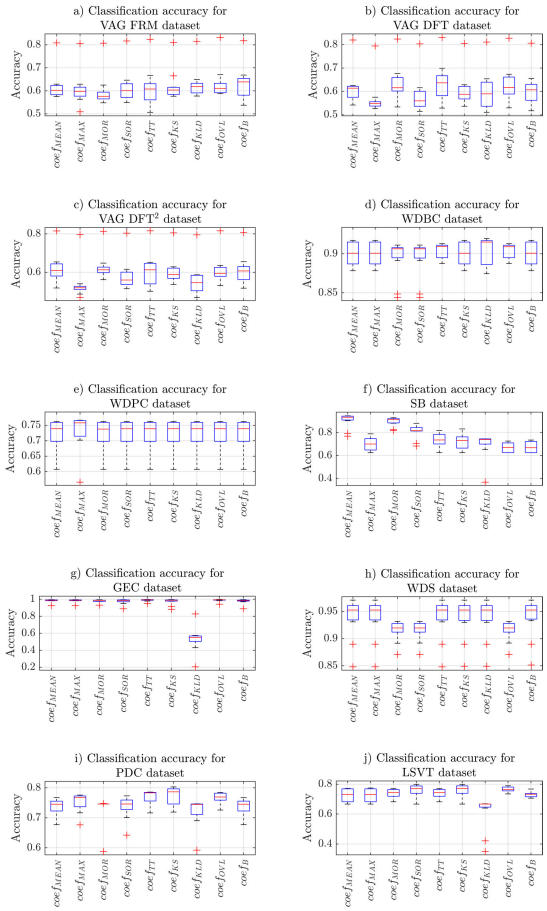

FIGURE 10. Classification accuracy of *VAG FRM* (plot a), *VAG DFT* (plot b), *VAG DFT*^{2} (plot c), *WDBC* (plot d), *WPBC* (plot e), *SB* (plot f), *GEC* (plot g), *WDS* (plot h), *PDC* (plot i) and *LSVT* (plot j) feature sets. There were 11 values used to build every box and every value was the average of 512 classification accuracies with different random splits.

IV. RESULTS

Figure 10 shows boxplots of the classification accuracy for particular sets of features. Spread of the values apparent on the boxplots shows that even though some coefficients can achieve better accuracies, they are not robust, i.e. their accuracy depends strongly on the classification algorithm used.

Most important information from the perspective of this research, so the maximal values and their standard deviations, with corresponding R_1 and R_2 ranks, were presented in Table 2. Note, that the maximal values are from 11 classifiers, and each of those 11 values is actually the mean value of 512 classifiers with random splits.

Detailed tables with specific classifiers accuracies for each coefficient were included in the Appendix C.

V. DISCUSSION

There were nine feature evaluation coefficients compared in this study. Each coefficient was used to evaluate every feature of ten different feature sets. $\binom{n}{2}$ best features of every set (where n is number of classes), in terms of each coefficient, created nine feature vectors. Each vector was then used to train a set of classification algorithms. Their accuracy allowed to compare coefficients in terms of:

TABLE 2. Results table. The maximal values are from 11 classifiers, and each is the average of 512 classifiers with different splits. Columns R_1 and R_2 contain ranks described in subsection III-F, while R is a simple ranking based on corresponding time values. Note, that the computation times were normalized by the highest value (the $coef_B$'s). In the headings, max is the maximal value and std is the standard deviation.

coefficient	max accuracy			accuracy std			time	
	value	R_1	R_2	value	R_1	R_2	value	R
$coef_{MEAN}$	0.857	1	3	0.047	4	3	0.009	2
$coef_{MAX}$	0.838	6	6	0.050	7	8	0.006	1
$coef_{MOR}$	0.849	3	8	0.044	2	2	0.804	7
$coef_{SOR}$	0.848	5	7	0.050	8	7	0.800	5
$coef_{TT}$	0.849	4	1	0.048	5	6	0.088	4
$coef_{KS}$	0.850	2	2	0.049	6	5	0.018	3
$coef_{KLD}$	0.806	9	9	0.076	9	9	0.980	8
$coef_{OVL}$	0.838	7	3	0.039	1	1	0.804	6
$coef_B$	0.834	8	5	0.045	3	4	1.000	9

- precision, that is: which coefficient led to the highest classification accuracy, regardless of the classifier used,
- robustness, that is: accuracies of which coefficient depended on the classification algorithm the least, and
- universality, that is: accuracies of which coefficient depended on the original data set the least.

Additionally, time of the computation was measured for a 100 000 executions of every coefficient, allowing to compare their computational complexity.

Knowledge of the probability density was required to evaluate three coefficients: namely $coef_{KLD}$, $coef_{OVL}$, and $coef_B$. The relationship between the KDE's estimation points and the computational time is linear (check Appendix B), so the time results from Table 2 for different number of points estimated in KDE, would also be different. Also, h parameter (variance) of a smoothing kernel could have additional impact on the values of particular coefficients.

Feature sets used in this research were chosen to ensure as thorough results as possible. Yet, no feature sets with missing values were studied. This matter could be investigated in future research. Based on obtained results, however, some universal conclusions can be drawn.

According to ranks in Table 2, the highest accuracy, in terms of R_1 was obtained for $coef_{MEAN}$, meaning that this coefficient gave the highest average of maximal accuracy values. However, according to R_2 rank, $coef_{MEAN}$ at times led to lower accuracy than $coef_{TT}$ and $coef_{KS}$. Minor difference between R_1 and R_2 suggests that obtained accuracies were slightly affected by the type of classified data set. Additionally, in terms of standard deviation (std), this coefficient proved to be rather mediocrely robust. Difference between std's R_1 and R_2 suggests that robustness of $coef_{MEAN}$ hardly depends on classified data set. Also, $coef_{MEAN}$ proved to be almost the least computationally expensive, giving way only to $coef_{MAX}$.

Besides this advantage, $coef_{MAX}$ turn out to be rather poor in every other aspect. It ranked 6 in both R_1 and R_2 accuracy ranks, and proved to be almost the worst when it comes to robustness. Small differences between R_1 and R_2 point out that those results are to be taken despite of the classified feature set. It is to be expected to some extent,

considering its rudimentary definition and rigorousness. Even for the features that have clearly different distributions for each class (like exemplary feature from Figures 5 and 6), value of $coef_{MAX}$ can be maximal. It is also very susceptible to outliers.

Insusceptible to outliers is subsequent coefficient: $coef_{MOR}$. In this case, however, difference between R_1 and R_2 accuracy ranks points rather high susceptibility to the data set change. Ranks of the standard deviation indicate, that this coefficient is noticeably robust, especially compared to the rest. Therefore, it could seem preferable for some specific data sets. However, $coef_{MEAN}$ led to higher maximum accuracy for almost every data set (with exception of the *VAG DFT* feature set), which makes the high robustness rather negligible. Additionally, almost highest computational cost excludes this coefficient from applications with low computational power potential, like, for example, embedded systems.

Similarly defined $coef_{SOR}$ is another coefficient insusceptible to outliers. Both are determined only by the values from the interquartile range. However, $coef_{SOR}$ seems much less robust. It is most probably caused by the lack of the median or the mean value in its definition. Some features could have same or very close medians (or means) for different classes, and still be considered informative by $coef_{SOR}$. In terms of the accuracy it is also not very impressive, but seems quite invulnerable to the change of the feature set; difference between R_1 and R_2 seems fairly small. Similarly to the previous coefficient, $coef_{SOR}$ is rather computationally expensive. It is surprising, considering that both do not require probability density estimation.

Another coefficient without the KDE requirement, so $coef_{TT}$, proved to be one of the most accurate, with the first place in R_2 rank and fourth place in R_1. This difference suggests, that the accuracy could be dependent on a feature set used. It is to be expected, as two-sample *t*-test is interpretable only for normally distributed data. However, according to performed tests, most classes in most feature sets turn out to be other than normal. Even for those sets (for example *VAG DFT*), $coef_{TT}$ achieved surprisingly well accuracy. Standard deviation ranks suggest, that the accuracy depends on the classifier more than on the feature set. With comparatively low computational cost, $coef_{TT}$ could constitute preferable choice, especially when the feature set is normally distributed.

When the distribution is not known, $coef_{KS}$, in which the empirical CDFs are utilized, seem to be the best choice. According to R_1 and R_2, accuracies obtained by this coefficient were high, independently of the feature set. It is not surprising, considering its definition. Additionally, it is also one of the least expensive coefficients. Only drawback of $coef_{KS}$ could be high dependency on a classifier used. According to tables in Appendix C, the highest accuracy was obtained using *ANN* and SVM_{GAUSS} classifiers. This is the last coefficient without the probability density in its definition.

TABLE 3. Non-normal parts of the features from particular classes in particular feature sets, according to the Shapiro-Wilk (SW) test, the Lilliefors' (LL) test, the Anderson-Darling (AD) test, and the Jarque-Bera (JB) test, with the significance level of 0.05.

dataset	class	SW	LL	AD	JB
VAG FRM	I	1.0000	0.9984	1.0000	0.9986
	II	0.9997	0.9301	0.9991	0.9985
	III	0.8979	0.6746	0.8414	0.8272
	IV	0.9918	0.9768	0.9880	0.9927
	V	0.9081	0.8227	0.8903	0.9137
VAG DFT	I	1.0000	0.9993	0.9999	0.9999
	II	0.9689	0.8813	0.9537	0.9188
	III	0.9690	0.8602	0.9493	0.9069
	IV	0.9816	0.9436	0.9719	0.9636
	V	0.9421	0.8292	0.9190	0.8716
$VAG\ DFT^2$	I	1.0000	1.0000	1.0000	1.0000
	II	0.9999	0.9976	0.9997	0.9965
	III	1.0000	0.9988	0.9999	0.9977
	IV	1.0000	0.9991	1.0000	0.9991
	V	0.9995	0.9950	0.9990	0.9925
WDBC	I	0.8333	0.7000	0.7000	0.8000
	II	0.9667	0.7667	0.9000	0.9667
WPBC	I	0.9091	0.8182	0.8485	0.8788
	II	0.5152	0.2121	0.3939	0.5152
SB	I	1.0000	1.0000	1.0000	0.9983
	II	1.0000	1.0000	1.0000	0.9971
GEC	I	0.7786	0.6440	0.7342	0.7740
	II	0.4243	0.3661	0.4116	0.4190
	III	0.7069	0.5208	0.6251	0.6994
	IV	0.5991	0.4679	0.5434	0.5984
	V	0.6351	0.5004	0.5828	0.6346
WDS	I	0.2308	0.3077	0.2308	0.2308
	II	0.5385	0.5385	0.3846	0.5385
	III	0.4615	0.4615	0.5385	0.2308
PDC	I	0.9748	0.9124	0.9562	0.9097
	II	0.9602	0.9535	0.9708	0.9535
LSVT	I	0.7749	0.5723	0.6238	0.7267
	II	0.7974	0.7363	0.7878	0.7685

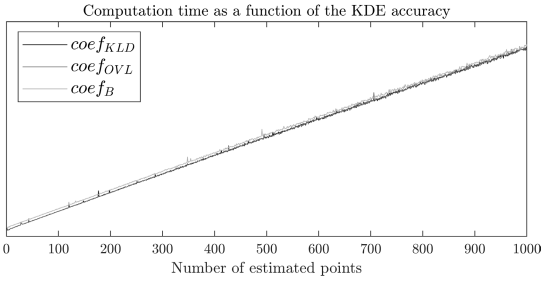

FIGURE 11. Calculation time as a function of the number of points estimated by the KDE. The function in clearly linear for every coefficient.

First coefficient which uses the Kernel Density Estimator is $coef_{KLD}$. It is, without doubt, worst coefficient in terms of precision and robustness. It could be caused by the highly nonlinear transformation of the original definition produced by the exponent term in Equation 11.

Another coefficient requiring the KDE is $coef_{OVL}$. Despite being the best coefficient in terms of the robustness, difference between accuracy's R_1 and R_2 suggests that accuracy obtained by $coef_{OVL}$ varies from data set to data set. Considering its high computational cost, it does not seem preferable in any situation.

TABLE 4. Exact classification results for the *VAG FRM* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.578	0.586	0.587	0.611	0.549	0.595	0.606	0.588	0.657
2	$TREE_2$	0.583	0.611	0.583	0.592	0.507	0.616	0.586	0.611	0.668
3	LDA	0.593	0.574	0.574	0.605	0.622	0.577	0.628	0.592	0.561
4	$BAYES$	0.629	0.602	0.569	0.601	0.596	0.598	0.618	0.602	0.577
5	SVM_{LIN}	0.628	0.591	0.597	0.637	0.667	0.613	0.625	0.671	0.645
6	SVM_{GAUSS}	0.575	0.563	0.557	0.549	0.603	0.578	0.577	0.632	0.613
7	KNN_{20}	0.609	0.629	0.625	0.646	0.634	0.610	0.611	0.612	0.639
8	KNN_5	0.602	0.618	0.576	0.554	0.617	0.665	0.649	0.588	0.594
9	$TREE_{BAG}$	0.604	0.598	0.564	0.591	0.608	0.604	0.634	0.632	0.644
10	$TREE_{BOOST}$	0.583	0.509	0.548	0.565	0.546	0.581	0.584	0.592	0.538
11	ANN	0.808	0.806	0.807	0.817	0.824	0.810	0.815	0.832	0.818
	std	0.066	0.073	0.072	0.074	0.083	0.067	0.065	0.071	0.075
	mean	0.618	0.608	0.599	0.615	0.616	0.623	0.630	0.632	0.632
	max	0.808	0.806	0.807	0.817	0.824	0.810	0.815	0.832	0.818

TABLE 5. Exact classification results for the *VAG DFT* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.567	0.575	0.613	0.536	0.529	0.571	0.532	0.586	0.575
2	$TREE_2$	0.572	0.538	0.616	0.539	0.601	0.567	0.551	0.617	0.543
3	LDA	0.612	0.555	0.598	0.560	0.637	0.599	0.587	0.616	0.607
4	$BAYES$	0.625	0.556	0.656	0.606	0.654	0.626	0.654	0.651	0.655
5	SVM_{LIN}	0.623	0.549	0.677	0.559	0.699	0.588	0.623	0.655	0.627
6	SVM_{GAUSS}	0.542	0.530	0.534	0.515	0.536	0.537	0.521	0.530	0.517
7	KNN_{20}	0.618	0.548	0.661	0.616	0.666	0.609	0.645	0.664	0.631
8	KNN_5	0.619	0.537	0.652	0.585	0.668	0.628	0.601	0.673	0.612
9	$TREE_{BAG}$	0.614	0.527	0.614	0.565	0.616	0.587	0.590	0.597	0.603
10	$TREE_{BOOST}$	0.586	0.547	0.592	0.525	0.575	0.564	0.511	0.571	0.558
11	ANN	0.820	0.794	0.824	0.804	0.830	0.805	0.811	0.827	0.806
	std	0.072	0.076	0.073	0.080	0.084	0.071	0.084	0.077	0.076
	mean	0.618	0.569	0.640	0.583	0.637	0.607	0.602	0.635	0.612
	max	0.820	0.794	0.824	0.804	0.830	0.805	0.811	0.827	0.806

TABLE 6. Exact classification results for the *VAG DFT*^{2} feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.589	0.487	0.620	0.536	0.502	0.571	0.497	0.577	0.575
2	$TREE_2$	0.578	0.519	0.613	0.539	0.535	0.567	0.524	0.615	0.543
3	LDA	0.609	0.529	0.562	0.560	0.630	0.599	0.579	0.586	0.607
4	$BAYES$	0.634	0.514	0.648	0.606	0.641	0.626	0.556	0.636	0.655
5	SVM_{LIN}	0.647	0.520	0.629	0.559	0.651	0.588	0.587	0.629	0.627
6	SVM_{GAUSS}	0.518	0.521	0.565	0.515	0.555	0.537	0.493	0.531	0.517
7	KNN_{20}	0.654	0.513	0.612	0.616	0.648	0.609	0.586	0.600	0.631
8	KNN_5	0.583	0.540	0.605	0.585	0.613	0.628	0.546	0.592	0.612
9	$TREE_{BAG}$	0.624	0.509	0.614	0.565	0.590	0.587	0.531	0.594	0.603
10	$TREE_{BOOST}$	0.580	0.469	0.596	0.525	0.501	0.564	0.469	0.535	0.558
11	ANN	0.815	0.796	0.813	0.804	0.815	0.805	0.795	0.816	0.806
	std	0.075	0.088	0.067	0.08	0.089	0.071	0.087	0.076	0.076
	mean	0.621	0.538	0.625	0.583	0.607	0.607	0.560	0.610	0.612
	max	0.815	0.796	0.813	0.804	0.815	0.805	0.795	0.816	0.806

The last coefficient included in this study, $coef_B$, turn out to be the most computationally expensive. It is surprising given that the KDE for $coef_B$ was evaluated at less points than $coef_{OVL}$ and $coef_{KLD}$ (50, 100 and 150 points accordingly). In terms of accuracy it proved to be rather mediocrely precise and dependent on the feature set used. Standard deviation ranks point out that the coefficient is moderately robust, making it unremarkable in all compared aspects.

Results obtained using coefficients $coef_{KLD}$, $coef_{OVL}$ and $coef_B$ suggest, that maybe the Kernel Density Estimator used for this study could be defined differently. Future studies could focus on the influence of the h (variance) parameter on the classification accuracy of particular coefficients. Furthermore, maybe whole smoothing function of the KDE could depend on the feature's distribution.

Despite this drawback, presented research can constitute an answer to the question: "Which coefficient should I use to my specific task?". Feature sets used in this study seem sufficient to draw some conclusions.

VI. CONCLUSION

For an applications without the limited computational power, $coef_{MEAN}$ or $coef_{KS}$ seem best suited. They are both fast to compute (though $coef_{MEAN}$ is about two times faster) and very

TABLE 7. Exact classification results for the *WDBC* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.887	0.887	0.906	0.904	0.909	0.887	0.919	0.909	0.887
2	$TREE_2$	0.888	0.888	0.908	0.907	0.910	0.888	0.917	0.910	0.888
3	LDA	0.907	0.907	0.891	0.891	0.912	0.907	0.874	0.912	0.907
4	$BAYES$	0.917	0.917	0.906	0.906	0.911	0.917	0.917	0.911	0.917
5	SVM_{LIN}	0.917	0.917	0.908	0.908	0.909	0.917	0.919	0.909	0.917
6	SVM_{GAUSS}	0.915	0.915	0.907	0.907	0.911	0.915	0.915	0.911	0.915
7	KNN_{20}	0.901	0.901	0.909	0.909	0.909	0.901	0.916	0.909	0.901
8	KNN_5	0.897	0.897	0.906	0.906	0.890	0.897	0.905	0.890	0.897
9	$TREE_{BAG}$	0.883	0.883	0.844	0.844	0.890	0.883	0.880	0.890	0.883
10	$TREE_{BOOST}$	0.878	0.878	0.849	0.849	0.888	0.878	0.876	0.888	0.878
11	ANN	0.914	0.914	0.911	0.911	0.913	0.914	0.911	0.913	0.914
	std	0.015	0.015	0.025	0.025	0.010	0.015	0.018	0.010	0.015
	mean	0.900	0.900	0.895	0.895	0.905	0.900	0.904	0.905	0.900
	max	0.917	0.917	0.911	0.911	0.913	0.917	0.919	0.913	0.917

TABLE 8. Exact classification results for the *WPBC* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.744	0.766	0.738	0.744	0.744	0.744	0.744	0.744	0.744
2	$TREE_2$	0.735	0.742	0.732	0.735	0.735	0.735	0.735	0.735	0.735
3	LDA	0.763	0.766	0.763	0.763	0.763	0.763	0.763	0.763	0.763
4	$BAYES$	0.720	0.765	0.720	0.720	0.720	0.720	0.720	0.720	0.720
5	SVM_{LIN}	0.763	0.763	0.763	0.763	0.763	0.763	0.763	0.763	0.763
6	SVM_{GAUSS}	0.763	0.754	0.763	0.763	0.763	0.763	0.763	0.763	0.763
7	KNN_{20}	0.739	0.758	0.739	0.739	0.739	0.739	0.739	0.739	0.739
8	KNN_5	0.688	0.702	0.688	0.688	0.688	0.688	0.688	0.688	0.688
9	$TREE_{BAG}$	0.690	0.705	0.690	0.690	0.690	0.690	0.690	0.690	0.690
10	$TREE_{BOOST}$	0.607	0.566	0.607	0.607	0.607	0.607	0.607	0.607	0.607
11	ANN	0.750	0.767	0.750	0.750	0.750	0.750	0.750	0.750	0.750
	std	0.047	0.060	0.047	0.047	0.047	0.047	0.047	0.047	0.047
	mean	0.724	0.732	0.723	0.724	0.724	0.724	0.724	0.724	0.724
	max	0.763	0.767	0.763	0.763	0.763	0.763	0.763	0.763	0.763

TABLE 9. Exact classification results for the *SB* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.943	0.714	0.918	0.820	0.710	0.730	0.745	0.682	0.695
2	$TREE_2$	0.943	0.682	0.918	0.815	0.710	0.700	0.745	0.670	0.667
3	LDA	0.768	0.625	0.816	0.683	0.642	0.636	0.711	0.625	0.625
4	$BAYES$	0.904	0.700	0.881	0.814	0.739	0.744	0.367	0.625	0.625
5	SVM_{LIN}	0.793	0.625	0.825	0.706	0.625	0.625	0.740	0.625	0.625
6	SVM_{GAUSS}	0.933	0.672	0.898	0.813	0.735	0.722	0.745	0.625	0.625
7	KNN_{20}	0.938	0.756	0.922	0.841	0.785	0.764	0.652	0.702	0.714
8	KNN_5	0.925	0.718	0.913	0.845	0.778	0.756	0.695	0.712	0.725
9	$TREE_{BAG}$	0.946	0.789	0.930	0.881	0.819	0.831	0.745	0.726	0.735
10	$TREE_{BOOST}$	0.946	0.785	0.929	0.877	0.815	0.825	0.734	0.724	0.734
11	ANN	0.924	0.640	0.890	0.816	0.697	0.653	0.745	0.626	0.626
	std	0.063	0.059	0.040	0.062	0.064	0.069	0.112	0.044	0.049
	mean	0.906	0.701	0.895	0.810	0.732	0.726	0.693	0.667	0.672
	max	0.946	0.789	0.930	0.881	0.819	0.831	0.745	0.726	0.735

precise. A little better robustness of $coef_{MEAN}$ is balanced by $coef_{KS}$'s lower dependency on feature set. The least computationally expensive coefficient to evaluate, so $coef_{MAX}$ is not recommended, as, compared to the rest of the coefficients, it is quite inaccurate.

In applications without computational power limitations, $coef_{TT}$ could be preferable. Although it is not as computationally expensive as some other coefficients, it requires the data to be normally distributed to be precisely classified. Therefore, to use it, it would be beneficial to analyze the feature set beforehand.

APPENDIX A
FEATURE SETS' DISTRIBUTION NORMALITY

In order to determine if the features' distributions are normal, the normality tests were performed:

1) Shapiro-Wilk (SW) test, which is a regression test [57],

TABLE 10. Exact classification results for the *GEC* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.984	0.984	0.973	0.976	0.989	0.976	0.560	0.989	0.982
2	$TREE_2$	0.982	0.982	0.973	0.887	0.978	0.882	0.526	0.984	0.887
3	LDA	0.989	0.989	0.988	0.984	0.992	0.991	0.522	0.991	0.985
4	$BAYES$	0.985	0.985	0.981	0.968	0.987	0.984	0.204	0.993	0.977
5	SVM_{LIN}	0.992	0.992	0.989	0.993	0.997	0.988	0.574	0.993	0.991
6	SVM_{GAUSS}	0.925	0.925	0.930	0.949	0.951	0.914	0.549	0.941	0.970
7	KNN_{20}	0.992	0.992	0.989	0.990	0.992	0.992	0.555	0.993	0.990
8	KNN_5	0.993	0.993	0.990	0.994	0.993	0.990	0.497	0.993	0.990
9	$TREE_{BAG}$	0.989	0.989	0.987	0.987	0.995	0.992	0.571	0.995	0.989
10	$TREE_{BOOST}$	0.985	0.985	0.977	0.977	0.988	0.989	0.431	0.990	0.985
11	ANN	0.995	0.995	0.993	0.994	0.997	0.996	0.827	0.997	0.995
	std	0.020	0.020	0.018	0.031	0.013	0.038	0.145	0.016	0.030
	mean	0.983	0.983	0.979	0.973	0.987	0.972	0.529	0.987	0.976
	max	0.995	0.995	0.993	0.994	0.997	0.996	0.827	0.997	0.995

TABLE 11. Exact classification results for the *WDS* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.931	0.931	0.919	0.920	0.931	0.930	0.930	0.920	0.933
2	$TREE_2$	0.848	0.848	0.892	0.892	0.848	0.849	0.849	0.892	0.851
3	LDA	0.947	0.947	0.910	0.910	0.947	0.947	0.947	0.910	0.947
4	$BAYES$	0.962	0.962	0.932	0.932	0.962	0.962	0.962	0.932	0.962
5	SVM_{LIN}	0.945	0.945	0.926	0.926	0.945	0.945	0.945	0.926	0.945
6	SVM_{GAUSS}	0.971	0.971	0.922	0.922	0.971	0.971	0.971	0.922	0.971
7	KNN_{20}	0.953	0.953	0.929	0.929	0.953	0.953	0.953	0.929	0.953
8	KNN_5	0.955	0.955	0.916	0.916	0.955	0.955	0.955	0.916	0.955
9	$TREE_{BAG}$	0.964	0.964	0.920	0.920	0.964	0.964	0.964	0.920	0.964
10	$TREE_{BOOST}$	0.890	0.890	0.871	0.871	0.890	0.890	0.890	0.871	0.890
11	ANN	0.956	0.956	0.929	0.929	0.956	0.956	0.956	0.929	0.956
	std	0.037	0.037	0.018	0.018	0.037	0.037	0.037	0.018	0.036
	mean	0.938	0.938	0.915	0.915	0.938	0.938	0.938	0.915	0.939
	max	0.971	0.971	0.932	0.932	0.971	0.971	0.971	0.932	0.971

TABLE 12. Exact classification results for the *PDC* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.768	0.774	0.746	0.770	0.783	0.797	0.746	0.784	0.767
2	$TREE_2$	0.758	0.776	0.746	0.773	0.784	0.803	0.746	0.762	0.758
3	LDA	0.740	0.755	0.746	0.746	0.785	0.746	0.745	0.768	0.740
4	$BAYES$	0.750	0.770	0.746	0.701	0.785	0.795	0.723	0.784	0.750
5	SVM_{LIN}	0.746	0.746	0.746	0.746	0.746	0.746	0.746	0.746	0.746
6	SVM_{GAUSS}	0.738	0.774	0.746	0.746	0.784	0.793	0.746	0.781	0.738
7	KNN_{20}	0.767	0.769	0.746	0.762	0.762	0.787	0.729	0.776	0.767
8	KNN_5	0.708	0.734	0.746	0.737	0.766	0.754	0.690	0.758	0.708
9	$TREE_{BAG}$	0.718	0.717	0.746	0.725	0.754	0.744	0.707	0.769	0.718
10	$TREE_{BOOST}$	0.677	0.677	0.587	0.642	0.716	0.719	0.592	0.726	0.677
11	ANN	0.745	0.767	0.748	0.751	0.786	0.798	0.747	0.783	0.745
	std	0.027	0.031	0.048	0.037	0.023	0.030	0.046	0.018	0.027
	mean	0.738	0.751	0.732	0.736	0.768	0.771	0.720	0.767	0.738
	max	0.768	0.776	0.748	0.773	0.786	0.803	0.747	0.784	0.767

2) Lilliefors' (LL) test, which is based on the empirical cumulative distribution functions (ECDF) and is used when the parameters of the tested distribution are not known [57],
3) Anderson-Darling test, which also utilizes the ECDFs and is usually used for heavy-tailed distributions [57],
4) Jarque-Bera test, which is based on the feature's skewness and kurtosis [57].

Tests' results were presented in Table 3. Decisions if the sets were normally distributed, were made with the significance level of 5%. Values presented in tables are non-normal parts of all feature values (for example, in the *VAGFRM* feature set, the Anderson-Darling test indicated that 84.14% of features from the third class are non-normal). Because of big size of *VAG FRM* feature set, its values were calculated not on full set, but on 100 000 randomly selected features.

TABLE 13. Exact classification results for the *LSVT* feature set. Every value is the mean accuracy of 512 trained classifiers with different random splits. Ordinal numbers in the tables correspond to subsection III-C. Three last rows present the standard deviation (std), average (mean) and maximal (max) values.

no.	classifier	$coef_{MEAN}$	$coef_{MAX}$	$coef_{MOR}$	$coef_{SOR}$	$coef_{TT}$	$coef_{KS}$	$coef_{KLD}$	$coef_{OVL}$	$coef_{B}$
1	$TREE_1$	0.731	0.729	0.696	0.781	0.694	0.783	0.667	0.760	0.733
2	$TREE_2$	0.771	0.774	0.770	0.770	0.770	0.771	0.667	0.764	0.734
3	LDA	0.667	0.667	0.768	0.667	0.768	0.667	0.666	0.758	0.706
4	$BAYES$	0.770	0.770	0.759	0.790	0.759	0.790	0.349	0.789	0.726
5	SVM_{LIN}	0.667	0.667	0.766	0.667	0.766	0.667	0.667	0.785	0.706
6	SVM_{GAUSS}	0.667	0.667	0.758	0.732	0.758	0.732	0.667	0.786	0.734
7	KNN_{20}	0.729	0.729	0.743	0.766	0.743	0.766	0.667	0.768	0.767
8	KNN_5	0.743	0.743	0.718	0.799	0.718	0.799	0.640	0.766	0.727
9	$TREE_{BAG}$	0.769	0.769	0.721	0.799	0.721	0.799	0.667	0.735	0.738
10	$TREE_{BOOST}$	0.731	0.731	0.682	0.780	0.682	0.780	0.421	0.746	0.745
11	ANN	0.768	0.768	0.728	0.759	0.728	0.759	0.668	0.754	0.713
	std	0.043	0.043	0.031	0.048	0.031	0.048	0.114	0.017	0.018
	mean	0.728	0.728	0.737	0.755	0.737	0.756	0.613	0.765	0.730
	max	0.771	0.774	0.770	0.799	0.770	0.799	0.668	0.789	0.767

TABLE 14. Detailed valued results of maximal classification accuracy for particular coefficients and feature sets. Ranks corresponding to this table were presented in table 15.

coefficient	*VAG FRM*	*VAG DFT*	*VAG DFT²*	*WDBC*	*WPBC*	*SB*	*GEC*	*WDS*	*PDC*	*LSVT*	mean value
$coef_{MEAN}$	0.808	0.820	0.815	0.917	0.763	0.946	0.995	0.971	0.768	0.771	0.857
$coef_{MAX}$	0.806	0.794	0.796	0.917	0.767	0.789	0.995	0.971	0.776	0.774	0.838
$coef_{MOR}$	0.807	0.824	0.813	0.911	0.763	0.930	0.993	0.932	0.748	0.770	0.849
$coef_{SOR}$	0.817	0.804	0.804	0.911	0.763	0.881	0.994	0.932	0.773	0.799	0.848
$coef_{TT}$	0.824	0.830	0.815	0.913	0.763	0.819	0.997	0.971	0.786	0.770	0.849
$coef_{KS}$	0.810	0.805	0.805	0.917	0.763	0.831	0.996	0.971	0.803	0.799	0.850
$coef_{KLD}$	0.815	0.811	0.795	0.919	0.763	0.745	0.827	0.971	0.747	0.668	0.806
$coef_{OVL}$	0.832	0.827	0.816	0.913	0.763	0.726	0.997	0.932	0.784	0.789	0.838
$coef_{B}$	0.818	0.806	0.806	0.917	0.763	0.735	0.995	0.971	0.767	0.767	0.834

TABLE 15. Detailed ranked results of maximal classification accuracy for particular coefficients and feature sets. Values corresponding to this table were presented in table 14.

coefficient	*VAG FRM*	*VAG DFT*	*VAG DFT²*	*WDBC*	*WPBC*	*SB*	*GEC*	*WDS*	*PDC*	*LSVT*	R_1 rank	R_2 rank
$coef_{MEAN}$	7	4	3	2	2	1	5	1	6	5	1	3
$coef_{MAX}$	9	9	8	2	1	6	5	1	4	4	6	6
$coef_{MOR}$	8	3	4	8	2	2	8	7	8	6	3	8
$coef_{SOR}$	4	8	7	8	2	3	7	7	5	1	5	7
$coef_{TT}$	2	1	2	6	2	5	1	1	2	6	4	1
$coef_{KS}$	6	7	6	2	2	4	3	1	1	1	2	2
$coef_{KLD}$	5	5	9	1	2	7	9	1	9	9	9	9
$coef_{OVL}$	1	2	1	6	2	9	2	7	3	3	7	3
$coef_{B}$	3	6	5	2	2	8	4	1	7	8	8	5

Note, that the vast majority of features' distributions are non-normal according to all performed test. Only one class in *WPBC*, few classes in *GEC* and most classes in the *WDS* data set proved to be normal in grater percentages than the rest.

APPENDIX B
KERNEL DENSITY ESTIMATION TIME

The Kernel Density Estimator was used to calculate $coef_{KLD}$, $coef_{OVL}$ and the $coef_{B}$ for different numbers of points, up to 1000. Each calculation was performed for a 100 times. Computation time as a function of the number of estimated points was plotted on Figure 11. The function is clearly linear for all the coefficients (Pearson's $\rho > 0.999$).

Note, that while $coef_{KLD}$ and $coef_{OVL}$ need similar time to compute, the $coef_{B}$ is slightly more computationally expensive.

APPENDIX C
EXACT CLASSIFICATION RESULTS

In Tables 4 - 13, exact classification results of particular feature sets were presented. Note, again, that every value is actually the mean of 512 classifiers with different random splits.

Tables 14 and 16 show accuracy's maximum values and standard deviations for particular coefficients and data sets. For easier interpretation, the values were also ranked (analogously to Table 2) and presented in Tables 15 and 17.

TABLE 16. Detailed valued results of classification accuracy's standard deviation for particular coefficients and feature sets. Ranks corresponding to this table were presented in table 17.

coefficient	VAG FRM	VAG DFT	VAG DFT^2	WDBC	WPBC	SB	GEC	WDS	PDC	LSVT	mean value
$coef_{MEAN}$	0.066	0.072	0.075	0.015	0.047	0.063	0.020	0.037	0.027	0.043	0.047
$coef_{MAX}$	0.073	0.076	0.088	0.015	0.060	0.059	0.020	0.037	0.031	0.043	0.050
$coef_{MOR}$	0.072	0.073	0.067	0.025	0.047	0.040	0.018	0.018	0.048	0.031	0.044
$coef_{SOR}$	0.074	0.080	0.080	0.025	0.047	0.062	0.031	0.018	0.037	0.048	0.050
$coef_{TT}$	0.083	0.084	0.089	0.010	0.047	0.064	0.013	0.037	0.023	0.031	0.048
$coef_{KS}$	0.067	0.071	0.071	0.015	0.047	0.069	0.038	0.037	0.030	0.048	0.049
$coef_{KLD}$	0.065	0.084	0.087	0.018	0.047	0.112	0.145	0.037	0.046	0.114	0.076
$coef_{OVL}$	0.071	0.077	0.076	0.010	0.047	0.044	0.016	0.018	0.018	0.017	0.039
$coef_{B}$	0.075	0.076	0.076	0.015	0.047	0.049	0.030	0.036	0.027	0.018	0.045

TABLE 17. Detailed ranked results of classification accuracy's standard deviation for particular coefficients and feature sets. Values corresponding to this table were presented in table 16.

coefficient	VAG FRM	VAG DFT	VAG DFT^2	WDBC	WPBC	SB	GEC	WDS	PDC	LSVT	R_1 rank	R_2 rank
$coef_{MEAN}$	2	2	3	3	2	6	5	7	4	5	4	3
$coef_{MAX}$	6	4	8	3	9	4	4	9	6	6	7	8
$coef_{MOR}$	5	3	1	9	1	1	3	3	9	3	2	2
$coef_{SOR}$	7	7	6	8	2	5	7	1	7	7	8	7
$coef_{TT}$	9	8	9	1	2	7	1	7	2	4	5	6
$coef_{KS}$	3	1	2	6	2	8	8	5	5	8	6	5
$coef_{KLD}$	1	9	7	7	2	9	9	6	8	9	9	9
$coef_{OVL}$	4	6	4	2	2	2	2	1	1	1	1	1
$coef_{B}$	8	5	5	3	2	3	6	4	3	2	3	4

REFERENCES

[1] D. Petri, "Big data, dataism and measurement," *IEEE Instrum. Meas. Mag.*, vol. 23, no. 3, pp. 32–34, May 2020.

[2] I. Guyon, S. Gunn, M. Nikravesh, A. L. Zadeh, and F. Extraction, *Number 207 in Studies in Fuzziness and Soft Computing*, 1st ed. Berlin, Germany: Springer-Verlag, 2006.

[3] K. Kręcisz and D. Bączkowicz, "Analysis and multiclass classification of pathological knee joints using vibroarthrographic signals," *Comput. Methods Programs Biomed.* vol. 154, pp. 37–44, Feb. 2018.

[4] A. Lysiak, A. Fron, D. Baczkowicz, and M. Szmajda, "Vibroarthrographic signal spectral features in 5-class knee joint classification," *Sensors*, vol. 20, no. 17, p. 5015, Sep. 2020.

[5] N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in *Proc. Biomed. Image Process. Biomed. Vis.*, 1993, pp. 861–870.

[6] D. Dua and C. Graff, "UCI machine learning repository," Irvine, School Inf. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech. Rep., 2017. [Online]. Available: http://archive.ics.uci.edu/ml/citation_policy.html

[7] N. J. Weinstein, A. E. Collisson, B. G. Mills, R. K. M. Shaw, A. B. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, and M. J. Stuart, "The cancer genome atlas pan-cancer analysis project," *Nature Genetics* vol. 45, no. 10, pp. 1113–1120, 2013.

[8] S. Aeberhard, D. Coomans, and O. de Vel, "Comparison of classifiers in high dimensional settings," Dept. Comput. Sci., Math. Statist., James Cook Univ. North Queensland, Douglas QLD, Australia, Tech. Rep. 92-02, 1992.

[9] C. O. Sakar, G. Serbes, A. Gunduz, H. C. Tunc, H. Nizam, B. E. Sakar, M. Tutuncu, T. Aydin, M. E. Isenkul, and H. Apaydin, "A comparative analysis of speech signal processing algorithms for Parkinson's disease classification and the use of the tunable Q-factor wavelet transform," *Appl. Soft Comput.*, vol. 74, pp. 255–263, Jan. 2019.

[10] A. Tsanas, M. A. Little, C. Fox, and L. O. Ramig, "Objective automatic assessment of rehabilitative speech treatment in Parkinson's disease," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 1, pp. 181–190, Jan. 2014.

[11] N. Sánchez-Marono, A. Alonso-Betanzos, and M. Tombilla-Sanromán, "Filter methods for feature selection—A comparative study," in *Intelligent Data Engineering and Automated Learning—IDEAL* (Lecture Notes in Computer Science), vol. 4881, H. Yin, P. Tino, E. Corchado, W. Byrne, and X. Yao, Eds. Berlin, Germany: Springer, 2007, pp. 178–187.

[12] B. Jantawan and C.-F. Tsai, "A comparison of filter and wrapper approaches with data mining techniques for categorical variables selection," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 2, no. 6, pp. 4501–4508, 2014.

[13] R. Porkodi, "Comparison of filter based feature selection algorithms: An overview," *Int. J. Innov. Res. Technol. Sci.*, vol. 2, no. 2, pp. 108–113, 2014.

[14] J. Suto, S. Oniga, and P. P. Sitar, "Comparison of wrapper and filter feature selection algorithms on human activity recognition," in *Proc. 6th Int. Conf. Comput. Commun. Control (ICCCC)*, May 2016, pp. 124–129.

[15] K. Ren, W. Fang, J. Qu, X. Zhang, and X. Shi, "Comparison of eight filter-based feature selection methods for monthly streamflow forecasting–three case studies on CAMELS data sets," *J. Hydrol.*, vol. 586, Jul. 2020, Art. no. 124897.

[16] Z. Ji and B. Wang, "Identifying potential clinical syndromes of hepatocellular carcinoma using PSO-based hierarchical feature selection algorithm," *BioMed Res. Int.*, vol. 2014, pp. 1–12, Mar. 2014.

[17] Z. Ji, G. Meng, D. Huang, X. Yue, and B. Wang, "NMFBFS: A NMF-based feature selection method in identifying pivotal clinical symptoms of hepatocellular carcinoma," *Comput. Math. Methods Med.*, vol. 2015, pp. 1–12, Jul. 2015, doi: 10.1155/2015/846942.

[18] P. Zhang, "A novel feature selection method based on global sensitivity analysis with application in machine learning-based prediction model," *Appl. Soft Comput.*, vol. 85, Dec. 2019, Art. no. 105859.

[19] Y. Zhang, H.-G. Li, Q. Wang, and C. Peng, "A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection," *Int. J. Speech Technol.*, vol. 49, no. 8, pp. 2889–2898, Aug. 2019.

[20] C. J. Wild, M. Pfannkuch, M. Regan, and N. J. Horton, "Towards more accessible conceptions of statistical inference," *J. Roy. Stat. Soc. A, Statist. Soc.*, vol. 174, no. 2, pp. 247–295, 2011.

[21] N. Nachkebia, M. Alexander, W. House, and H. North, "The simple theory of informal rules," *Math. Teach. Res. J. Online*, vol. 6, no. 1, p. 17, 2013.

[22] J. S. Rao and H. Liu, "Discordancy partitioning for validating potentially inconsistent pharmacogenomic studies," *Sci. Rep.*, vol. 7, no. 1, p. 15169, Dec. 2017.

[23] R. Peck and J. Devore, *Statistics: The Exploration and Analysis of Data*, 6th ed. London, U.K.: Duxbury Press, 2011.

[24] P. Baldi and A. D. Long, "A Bayesian framework for the analysis of microarray expression data: Regularized *t*-test and statistical inferences of gene changes," *Bioinformatics*, vol. 17, no. 6, pp. 509–519, Jun. 2001.
[25] J. Jaeger, R. Sengupta, and W. L. Ruzzo, "Improved gene selection for classification of microarrays," in *Proc. Pacific Symp. Biocomput.*, vol. 8, Jan. 2003, pp. 53–64.
[26] Y. Su, T. M. Murali, V. Pavlovic, M. Schaffer, and S. Kasif, "RankGene: Identification of diagnostic genes based on expression data," *Bioinformatics*, vol. 19, no. 12, pp. 1578–1579, Aug. 2003. [Online]. Available: https://academic.oup.com/bioinformatics/article-pdf/19/12/1578/715741/btg179.pdf
[27] B. Wu, T. Abbott, D. Fishman, W. Mcmurray, G. Mor, K. Stone, D. Ward, K. Williams, and H. Zhao, "Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data," *Bioinformatics*, vol. 19, no. 13, pp. 1636–1643, Sep. 2003.
[28] I. Levner, "Feature selection and nearest centroid classification for protein mass spectrometry," *BMC Bioinf.*, vol. 6, no. 1, p. 68, 2005.
[29] N. Zhou and L. Wang, "A modified T-test feature selection method and its application on the HapMap genotype data," *Genomics, Proteomics Bioinf.*, vol. 5, nos. 3–4, pp. 242–249, 2007.
[30] B. Chandra and M. Gupta, "An efficient statistical feature selection approach for classification of gene expression data," *J. Biomed. Informat.*, vol. 44, no. 4, pp. 529–535, Aug. 2011.
[31] T. K. P. Shri and N. Sriraam, "Comparison of *t*-test ranking with PCA and SEPCOR feature selection for wake and stage 1 sleep pattern recognition in multichannel electroencephalograms," *Biomed. Signal Process. Control*, vol. 31, pp. 499–512, Jan. 2017.
[32] F. J. Massey, "The Kolmogorov-Smirnov test for goodness of fit," *J. Amer. Stat. Assoc.*, vol. 46, no. 253, pp. 68–78, Mar. 1951.
[33] R. R. Wilcox, "Some practical reasons for reconsidering the Kolmogorov-Smirnov test," *Brit. J. Math. Stat. Psychol.*, vol. 50, no. 1, pp. 9–20, May 1997.
[34] J. Biesiada and W. Duch, "Feature selection for high-dimensional data: A Kolmogorov-Smirnov correlation-based filter," in *Computer Recognition Systems* (Advances in Soft Computing), M. Kurzyński, E. Puchała, M. Woźniak, A. Żołnierek, Eds. Berlin, Germany: Springer, 2005, pp. 95–103.
[35] A. Ivanov and G. Riccardi, "Kolmogorov-Smirnov test for feature selection in emotion recognition from speech," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 5125–5128.
[36] W. J. Pratt and D. J. Gibbons, *Concepts Nonparametric Theory* (Springer Series in Statistics). New York, NY, USA: Springer, 2012.
[37] J. Novovicova, P. Pudil, and J. Kittler, "Divergence based feature selection for multimodal class densities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 2, pp. 218–223, Feb. 1996.
[38] F. M. Coetzee, "Correcting the Kullback–Leibler distance for feature selection," *Pattern Recognit. Lett.*, vol. 26, no. 11, pp. 1675–1683, Aug. 2005.
[39] Z. Zhen, X. Zeng, H. Wang, and L. Han, "A global evaluation criterion for feature selection in text categorization using Kullback-Leibler divergence," in *Proc. Int. Conf. Soft Comput. Pattern Recognit. (SoCPaR)*, Oct. 2011, pp. 440–445.
[40] Y. Lifang, Q. Sijun, and Z. Huan, "Feature selection algorithm for hierarchical text classification using Kullback-Leibler divergence," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Big Data Anal. (ICCCBDA)*, Apr. 2017, pp. 421–424.
[41] D. J. C. MacKay and D. J. C. M. Kay, *Information Theory, Inference & Learning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[42] M. S. Weitzman, *Measures of Overlap of Income Distributions of White and Negro Families in the United States*. Suitland-Silver Hill, MD, USA: U.S. Bureau of the Census, 1970.
[43] H. F. Inman and E. L. Bradley, "The overlapping coefficient as a measure of agreement between probability distributions and point estimation of the overlap of two normal densities," *Commun. Statist. Theory Methods*, vol. 18, no. 10, pp. 3851–3874, Jan. 1989.
[44] B. Milanovic and S. Yitzhaki, "Decomposing world income distribution: Does the world have a middle class?" Policy Res. Work. Papers, World Bank, Washington, DC, USA, Policy Res. Working Paper 2562, 2002.
[45] F. Schmid and A. Schmidt, "Nonparametric estimation of the coefficient of overlapping—Theory and empirical application," *Comput. Statist. Data Anal.*, vol. 50, no. 6, pp. 1583–1596, Mar. 2006.
[46] G. Anderson, O. Linton, and Y.-J. Whang, "Nonparametric estimation and inference about the overlap of two distributions," *J. Econometrics*, vol. 171, no. 1, pp. 1–23, Nov. 2012.
[47] M. Pastore and A. Calcagnì, "Measuring distribution similarities between samples: A distribution-free overlapping index," *Frontiers Psychol.*, vol. 10, p. 1089, May 2019.
[48] P. Jaccard, "Distribution comparée de la flore alpine dans quelques régions des Alpes occidentales et orientales," *Bulletin de la Murithienne*, vol. 37, no. 140, pp. 241–272, 1902.
[49] A. Djouadi, O. Snorrason, and F. D. Garber, "The quality of training sample estimates of the Bhattacharyya coefficient," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 92–97, Jan. 1990.
[50] B. Mak and E. Barnard, "Phone clustering using the Bhattacharyya distance," in *Proc. 4th Int. Conf. Spoken Lang. Process. (ICSLP)*, vol. 4, 1996, pp. 2005–2008.
[51] X. Guorong, C. Peiqi, and W. Minhui, "Bhattacharyya distance feature selection," in *Proc. 13th Int. Conf. Pattern Recognit.*, vol. 2, 1996, pp. 195–199.
[52] F. J. Aherne, N. A. Thacker, and P. I. Rockett, "The Bhattacharyya metric as an absolute similarity measure for frequency coded data," *Kybernetika*, vol. 34, no. 4, pp. 363–368, 1998.
[53] E. Choi and C. Lee, "Feature extraction based on the Bhattacharyya distance," *Pattern Recognit.*, vol. 36, no. 8, pp. 1703–1709, Aug. 2003.
[54] S. Bi, M. Broggi, and M. Beer, "The role of the Bhattacharyya distance in stochastic model updating," *Mech. Syst. Signal Process.*, vol. 117, pp. 437–452, Feb. 2019.
[55] J. Lu, J. Yue, L. Zhu, and G. Li, "Variational mode decomposition denoising combined with improved Bhattacharyya distance," *Measurement*, vol. 151, Feb. 2020, Art. no. 107283.
[56] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-validation," in *Encyclopedia of Database Systems*, L. Liu and M. T. Özsu, Eds. New York, NY, USA: Springer, 2009, pp. 532–538.
[57] B. W. Yap and C. H. Sim, "Comparisons of various types of normality tests," *J. Stat. Comput. Simul.*, vol. 81, no. 12, pp. 2141–2155, Dec. 2011.
[58] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "RUSBoost: A hybrid approach to alleviating class imbalance," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 40, no. 1, pp. 185–197, Jan. 2010.
[59] A. W. Bowman and A. Azzalini, *Applied Smoothing Techniques for Data Analysis* (Number 18 in Oxford Statistical Science Series). Oxford, U.K.: Oxford Univ. Press, 1997.

ADAM ŁYSIAK received the engineering and M.Sc. degrees in automatic control and robotics from the Opole University of Technology, Poland, in 2018 and 2019, respectively, where he is currently pursuing the Ph.D. degree in automation, electronic and electrical engineering. His current research interests include biomedical signals analysis, especially vibroarthrography and electroencephalography, feature extraction, and statistics.

MIROSŁAW SZMAJDA graduate from the Faculty of Electronics and Telecommunications, Wroclaw University of Technology, in 2000. He received the habilitation degree from the Opole University of Technology in 2014. In 2006, he defended his Ph.D. thesis in electrical engineering with the Opole University of Technology. Since 2000, he has been working with the Faculty of Electrical Engineering, Automatic Control and Informatics, Opole University of Technology. He is currently an Associate Professor. He is also the Head of the Division of Control Science and Engineering with the Faculty of Electrical Engineering, Automatic Control and Informatics, Opole University of Technology. He lectures metrology, electrical engineering, electronics, and computer science. He is the author of many publications, industrial studies, and a manager and participant of research projects, including international. He conducts scientific research in three key areas, such as power quality and electrical disturbances, use of advanced methods of digital signal processing in the processing of electrical and biomedical signals, developing of measurement systems based on microcontrollers and signal processors.