Int. J. Appl. Math. Comput. Sci., 2021, Vol. 31, No. 4, 685–696
DOI: 10.34768/amcs-2021-0047

A WEIGHTED WRAPPER APPROACH TO FEATURE SELECTION

MACIEJ KUSY [a,*], ROMAN ZAJDEL [a]

[a]Faculty of Electrical and Computer Engineering
Rzeszów University of Technology
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
e-mail: mkusy@prz.edu.pl

This paper considers feature selection as a problem of an aggregation of three state-of-the-art filtration methods: Pearson's linear correlation coefficient, the ReliefF algorithm and decision trees. A new wrapper method is proposed which, on the basis of a fusion of the above approaches and the performance of a classifier, is capable of creating a distinct, ordered subset of attributes that is optimal based on the criterion of the highest classification accuracy obtainable by a convolutional neural network. The introduced feature selection uses a weighted ranking criterion. In order to evaluate the effectiveness of the solution, the idea is compared with sequential feature selection methods that are widely known and used wrapper approaches. Additionally, to emphasize the need for dimensionality reduction, the results obtained on all attributes are shown. The verification of the outcomes is presented in the classification tasks of repository data sets that are characterized by a high dimensionality. The presented conclusions confirm that it is worth seeking new solutions that are able to provide a better classification result while reducing the number of input features.

Keywords: feature selection, wrapper approach, feature significance, weighted combined ranking, convolutional neural network, classification accuracy.

1. Introduction

Feature selection (FS) is treated as the process of finding and picking a subset of the most significant attributes in the data. As a consequence, the dimensionality of the considered input space is reduced. In contrast to feature extraction, in FS the original feature values are left unchanged. In general, FS is divided into the solutions known as filters (variables are chosen independently of a learning model), wrappers (a learning model is involved in selection of a set of variables) and embedded methods. In embedded methods, the input variables are chosen within a training process of a learning model (Guyon and Elisseeff, 2003). Since in this research, the filter and wrapper approaches are studied, some attention to these methods is paid below.

In filter based FS, the attributes are selected taking into account the association of particular inputs and the target output. The best known solutions are statistical methods such as Pearson's correlation, chi-square or the Fisher scoring algorithm. However, there are also other algorithms, including ReliefF (Robnik-Šikonja and Kononenko, 2003), where the weight computed for each feature establishes its importance, or the decision tree, an example of which is CART (Breiman *et al.*, 1984), where the information gain within the feature split imposes its significance.

Although frequently applied, in many cases, the use of a single FS method may be insufficient to extract the most important attributes from the data (Bolón-Canedo *et al.*, 2013; Vergara and Estévez, 2014). This results from the fact that some selection techniques may be efficient in resolving particular problems, but useless in others (Awada *et al.*, 2012). For this reason, a lot of up-to-date studies have been published that focus on merging state-of-the-art FS methods so as to discover an optimal subset of important features. The following research is worth mentioning here: combinatorial fusion analysis (Li *et al.*, 2013), a voting scheme combination framework (Rokach *et al.*, 2006) or aggregation function based feature ranking (Pes, 2020). The idea of fusion of various FS methods has also been investigated by the authors of the current work (Kusy *et al.*, 2020; Zajdel *et al.*, 2020).

[*]Corresponding author

The second group of methods, which are considered even more efficient than filter based FS solutions, are known as wrapper approaches (WAs). In contrast to filters, WAs utilize the output of an applied model for detecting the significance of features. WAs conduct a search for a feature subset in the space of possible features. The search requires a state space, an initial state, a termination condition and a search engine (Russell and Norvig, 1995). The feature state representation can be easily understood using a bit-wise notation where each bit indicates whether a feature is present (indicated as 1) or absent (indicated as 0) in the data set. Based on such a representation, the initial state can be determined either with a single feature or all features but one included. Such a configuration allows for applying a search algorithm, the objective of which is to find the state for which the highest accuracy of the model is attained. Unfortunately, the size of the search space for I possible features is $O(2^I)$; therefore, it is impractical to search the whole space exhaustively (Devijver and Kittler, 1982).

Based on the search strategy, WAs can be generally split into the following categories (El Aboudi and Benhlima, 2016): exponential, e.g., the branch and bound algorithm (Narendra and Fukunaga, 1977), population based (Lu *et al.*, 2008; Rodrigues *et al.*, 2014) and sequential, e.g., forward/backward selection (Whitney, 1971; Broughton *et al.*, 2010).

In this paper, we propose a new wrapper approach for feature selection. It relies on merging three known FS methods, i.e., Pearson's linear correlation coefficient (PC), ReliefF (RF) and a single decision tree (DT). Based on the assumed formal criteria, which refer to feature importance ranking and the accuracy of a classification model, the approach selects the most significant subset of attributes from the input patterns. As the investigated model, we utilize a one-dimensional convolutional neural network. This has been receiving greater and greater attention in the literature due to its prominent performance in several applications (Abdeljaber *et al.*, 2018; Abdel-Hamid *et al.*, 2014a; Kiranyaz *et al.*, 2015a; Abdeljaber *et al.*, 2017). Although our method utilizes the idea of wrappers, it can also be perceived as a hybrid solution. This is because the conducted selection of features is based both on the accuracy of the classifier and the application of filtering methods. This paper can be treated as an extension of the work presented by Kusy *et al.* (2020) and Zajdel *et al.* (2020). However, its main contribution is the introduction of a novel weighted combined ranking score that allows for selecting the most significant features. Such a solution creates a new wrapper approach to feature selection.

The rest of the paper is arranged as follows. In Section 2, the issue of feature selection is discussed. Here, the basis of the filter methods and the wrapper approaches applied in the current work are put forward. Section 3 sets out a description of the utilized convolutional neural network. Section 4 is devoted to the presentation of the proposed method. The case study that depicts implementation of the method for a particular database is introduced in Section 5. In Section 6, we characterize the examined data sets and delineate the settings of the CNN. In this part of the article, we provide a thorough analysis of the obtained results. Section 7 concludes the work.

2. Feature selection

The proposed method utilizes filter feature selection methods in its operation; therefore, in the first part of the current section, we devote some attention to these particular techniques. Since our method belongs to the class of wrapper approaches, the concept of WA is explained in the second part of this section.

2.1. Filter methods.

In this work, two known and widely used FS filter methods are selected for research that do not reduce the number of features. These are Pearson's correlation coefficient and the ReliefF algorithm. The third filter method, the decision tree, is chosen due to its property of reducing the number features.

2.1.1. Pearson's linear correlation coefficient.

PC is a statistic measure used to establish a linear correlation between two variables. For the i-th feature, $i = 1, \dots, I$, of L input records and the available data targets t_l, the correlation between the i-th input and the output can be defined as follows (Benesty *et al.*, 2009):

$$r_i = \frac{\sum_{l=1}^{L} \left(x_{li} - \overline{x}_i\right)\left(t_l - \overline{t}\right)}{\sqrt{\sum_{l=1}^{L} \left(x_{li} - \overline{x}_i\right)^2 \sum_{l=1}^{L} \left(t_l - \overline{t}\right)^2}}, \tag{1}$$

where $\overline{x}_i = (1/L) \cdot \sum_{l=1}^{L} x_{li}$ is the mean over the i-th feature, while $\overline{t} = (1/L) \cdot \sum_{l=1}^{L} t_l$ denotes the mean over the targets of all $\mathbf{x}_l$ records for $l = 1, \dots, L$. If $\mathbf{x}_l$ and t_l are linearly dependent, $r_i = \pm 1$; $r_i = 0$ if the variables are not correlated. On the basis of decreasing values of $|r_i|$, the ranking of feature importance can be established.

2.1.2. ReliefF algorithm.

RF (Robnik-Šikonja and Kononenko, 2003) is a filter based approach developed to determine the ranking of features. It finds the weights of the attributes that reflect the relevance of the predictors. For a randomly selected record $\mathbf{x}_l$, ReliefF searches for $k = 1, \dots, K$ of its nearest neighbors from class c (nearest hits $\mathbf{h}_k$) and k nearest neighbors from each of $j = 1, \dots, J$ classes (nearest misses $\mathbf{m}_k^j$) for $j \neq c$. The weights of the features are determined according to the following formula:

$$w_i^{\text{new}} = w_i^{\text{old}} - \frac{1}{L \cdot K}\Big[\sum_{k=1}^{K} \Delta(x_{li}, h_{ki})^2 + \sum_{j=1,\, j\neq c}^{J} \frac{P_j}{1-P_c} \sum_{k=1}^{K} \Delta\big(x_{li}, m_{ki}^{(j)}\big)^2\Big], \tag{2}$$

where h_{ki} and $m_{ki}^{(j)}$ are the i-th elements of $\mathbf{h}_k$ and $\mathbf{m}_k^j$, respectively; P_j and P_c denote the occurrence probabilities of class j and class c; Δ determines the discrepancy between the i-th feature of the vectors; $\Delta = \{0, 1\}$ and $\Delta \in [0, 1]$ hold for discrete and continuous attributes, respectively. A higher feature's relevance results from a greater weight value.

2.1.3. Decision tree. A DT is a hierarchical structure used in decision processes. If the targets for given records are discrete, the DT is a classification tree. The DT takes the form of a graph composed of nodes, branches and leaves. In classification trees, nodes represent a data split based on a given attribute, branches denote possible courses of action available at a node while leaves are the counterpart of decisions. While growing the tree, it is possible to establish the importance of data features. In CART (Breiman *et al.*, 1984), such an importance is determined by the decrease in node impurity weighted by the node probability:

$$\Delta\mathcal{I} = P_n \mathcal{I}(\mathbf{X}_n) - \big(P_{(n,l)}\mathcal{I}(\mathbf{X}_{(n,l)}) + P_{(n,r)}\mathcal{I}(\mathbf{X}_{(n,r)})\big), \tag{3}$$

where $\mathcal{I}$ is the Gini index or entropy indicator, $\mathbf{X}_n$ denotes the set of records reaching the n-th node; $\mathbf{X}_{(n,l)}$ and $\mathbf{X}_{(n,r)}$ stand for the records reaching the left and right child nodes of the n-th parent, respectively. In (3), $P_n = |\mathbf{X}_n|/L$, $P_{(n,l)} = |\mathbf{X}_{(n,l)}|/L_n$ and $P_{(n,r)} = |\mathbf{X}_{(n,r)}|/L_n$ are the probabilities of the nodes n, (n, l) and (n, r), respectively. By maximizing the impurity $\Delta\mathcal{I}$ over all possible splitting attributes, the features' significance is obtained.

2.2. Wrapper approaches. Feature selection performed according to a wrapper approach requires an application of a model (e.g., a classifier). The main idea of the WA is as follows. A feature subset selection algorithm is employed to choose the subset of attributes. Based on these attributes a new data set is established for which the model's performance is evaluated. Such a performance can be assessed by certain statistical indicators, e.g., accuracy (Kohavi and John, 1997).

Among many alternatives, sequential feature selectors constitute an important family of search algorithms. Basically, they add or remove one feature at a time, and count the performance of a model until a feature subset of a desired size S ($S < I$) is achieved. In this work, sequential forward selection (SFS) and sequential backward selection (SBS) are used as reference methods; therefore, they are shortly discussed herein. Since the idea of both SFS and SBS is fairly similar, only the first is presented below.

We are given a set F of available input features: $F = \{f_1, f_2, \dots, f_I\}$. The aim of SFS is to select a subset of features $\varPhi_S = \{\phi_1, \dots, \phi_s, \dots, \phi_S\}$ where $\phi_s \in F$ and $S = 0, 1, 2, \dots, I$, provided that $S < I$. Initially, for $S = 0$, $\varPhi_0 = \emptyset$. Now, let

$$R = F \setminus \varPhi_S \tag{4}$$

be the reduced set of features. Let also $Q(\cdot)$ stand for the quality measure of a classifier (e.g., accuracy). The feature that is associated with the best classifier performance can be established as follows:

$$\hat{\phi} = \arg \max_{r=1,\dots,|R|} Q\left(\varPhi_S \cup \phi_r\right), \tag{5}$$

where $\phi_r \in R$. Once $\hat{\phi}$ is found, it is included in the next, enlarged feature subset:

$$\varPhi_{S+1} = \varPhi_S \cup \hat{\phi}. \tag{6}$$

Setting $S \leftarrow S + 1$ reduces R from which ϕ_r are checked for selection in (5). The process of including further features (4)–(6) is repeated until some assumed stopping criterion is met.

The SBS approach is conducted in a similar manner but it is initialized as follows: $\varPhi_S = F$, $S = I$ and $R = \varPhi_S$. The quality Q is computed on the set $\varPhi_S \setminus \phi_r$ and, instead of inclusion as in SFS, the exclusion of the feature takes place: $\varPhi_{S-1} = \varPhi_S \setminus \hat{\phi}$. Afterwards, $S \leftarrow S - 1$ and the process of excluding next features is repeated.

3. Convolutional neural network

The convolutional neural network (CNN), proposed by LeCun *et al.* (1998), is a type of feed-forward neural network whose architecture is arranged in layers of a repeatedly occurring operation. After a single input layer, often called the input volume due to the characteristics of incoming signals, multiple convolutional, pooling and activation layers appear. Neurons are usually activated by rectified linear units which compute the following operation: ReLU$(x) = \max(0, x)$. Some normalization and dropping operations may also be introduced. The CNN's final stages are formed by a combination of fully connected layers and an output layer that provides some classification result.

The CNN is designed to process data that take the form of multiple arrays, e.g., images containing pixel values in the three color channels (LeCun *et al.*, 2015).

However, in spite of common CNN application in image recognition tasks (Scherer *et al*., 2010; Krizhevsky *et al*., 2017; Koziarski and Cyganek, 2018; Wang *et al*., 2020), this network finds its usage when the input signals emerge as labeled feature patterns (Azizjon *et al*., 2020) and time series records (Abdeljaber *et al*., 2018; Kiranyaz *et al*., 2021). Such a model is known as a one dimensional CNN (1D–CNN).

From a structural point of view, three distinct layer types are utilized in the 1D–CNN:

- convolutional (CONV), where a one-dimensional convolution operation is applied to the input signals;
- pooling, where information down-sampling occurs and similar features are merged into one;
- dense, that fully connects the last processing layer with the final output.

The last type is also known as the multilayer perceptron (MLP) layer due to its similarity to an MLP's structure. A 1D–CNN is adjusted by means of the following hyper-parameters: the number of CONV and MLP neurons and layers, the filter size in each CONV layer, the pooling factor and the choice of pooling and activation functions.

Operationally, during the forward propagation, each k-th neuron of the l-th layer in a 1D–CNN accepts the following signal:

$$\mathbf{x}_e^{(f)} = \sum_{g=1}^{G} \mathcal{C}\big(\mathbf{y}_g^{(f-1)}, \mathbf{w}_{eg}^{(f-1)}\big) + b_e^{(f)}, \qquad (7)$$

where

- $\mathbf{y}_g^{(f-1)}$ denotes the output signal of the g-th neuron in layer $f-1$;
- $\mathbf{w}_{eg}^{(f-1)}$ stands for the filter array (kernel) from the g-th neuron in the $(f-1)$-th layer to the e-th neuron in layer f;
- $b_e^{(f)}$ indicates the bias coefficient of the e-th neuron in the f-th layer;
- G is the number of neurons in layer $f-1$.

In (7), $\mathcal{C}$ is a convolution generally defined as:

$$\mathcal{C} = (y \star w)(x) = \sum_{i=-\infty}^{\infty} y(i) \cdot w(x-i), \qquad (8)$$

where x and i refer to the indices of the elements occurring in the signal y and the filter w. The inner f-th activation computed on $\mathbf{x}_e^{(f)}$ is then determined as follows:

$$\mathbf{y}_e^{(f)} = \mathrm{ReLU}\big(\mathbf{x}_k^{(l)}\big). \qquad (9)$$

Next, the signal (9) undergoes down-sampling based on a selected pooling operation. The 1D–CNN's training process is based on backpropagating the error from the MLP layer to the input layer, and is realized by computing the gradients of an error function with respect to both the input signal and the filter.

1D–CNNs have become the state-of-the-art application tools in engineering fields such as automatic speech recognition (Abdel-Hamid *et al*., 2014b), electrocardiogram monitoring (Kiranyaz *et al*., 2015b) or bearing fault detection (Eren, 2017).

The specification (structure and hyper-parameters) of a 1D–CNN used in this study is presented in Section 6.

4. Proposed method

In this section, we introduce a new approach that allows us to isolate the most relevant subset of attributes out of the entire set of available variables. The idea is universal; therefore, it can be tested on multiple databases. Generally, in the proposed method, we fuse the outcomes of three different FS methods, which are expressed in terms of the ranking of features' significance. Based on the ordered feature indices and the appropriate selection criterion, a weighted combined ranking that considers the performance of the classifier is propounded. It establishes a suboptimal subset of attributes ordered from the most to least significant ones.

Due to the common use of the accuracy in evaluating the FS performance (Cannas *et al*., 2013; Rodrigues *et al*., 2014; Wuniri *et al*., 2019), this measure is also applied in the current study. It is determined as

$$Acc = \frac{1}{L} \sum_{l=1}^{L} \delta\left[y(\mathbf{x}_l) = t_l\right], \qquad (10)$$

where $y(\mathbf{x}_l)$ is the network's output obtained for $\mathbf{x}_l$. In (10), $\delta\left[\cdot\right] = 1$ when $y(\mathbf{x}_l) = t_l$, and 0 otherwise. The accuracy (10) is determined so as to ensure the highest generalization ability.

In order to select some feature as important, two definitions need to be introduced.

Definition 1. Let $\mathbf{P}$, $\mathbf{R}$ and $\mathbf{T}$ stand for the subsets of indices of the attributes ordered in terms of their significance obtained by the PC, RF and DT methods, respectively. Also, let $\mathbf{P}_j$, $\mathbf{R}_j$ and $\mathbf{T}_j$ denote the subsets of the first j elements of $\mathbf{P}$, $\mathbf{R}$ and $\mathbf{T}$, respectively. Then the *set of common feature indices* is defined as follows (Kusy *et al*., 2020):

$$\mathbf{C}_j = [(\mathbf{P}_j \cup \mathbf{R}_j) \cap \mathbf{T}] \cup \mathbf{T}_j, \qquad (11)$$

where $j = 1, \dots, |\mathbf{T}|$.

The set $\mathbf{T}$ plays a role of a filter that allows the elements of $\mathbf{P}$ and $\mathbf{R}$ to be included in $\mathbf{C}_j$.

Definition 2. Given the set of common feature indices $\mathbf{C}_j$ defined in (11). Let (i) C_{jk} indicate some feature that is the k-th element of $\mathbf{C}_j$, (ii) $\mathbf{X}\left\{C_{jk}\right\}$ denote some natural number that is a direct reference to the index of C_{jk} in $\mathbf{X}$, where $\mathbf{X}$ is any of predefined sets of feature indices. The *weighted combined ranking score*, which is determined for the feature C_{jk} selected in $\mathbf{P}_j$, $\mathbf{R}_j$ and $\mathbf{T}_j$ simultaneously, is defined as follows:

$$R_{C_{jk}} = w_k^{\mathbf{C}_j} \sum_{s=1}^{3} \left(|\mathbf{T}| - \mathbf{X}_s\left\{C_{jk}\right\} + 1\right) \tag{12}$$

for $\mathbf{X}_s\left\{C_{jk}\right\} \leq |\mathbf{T}|$ and

$$w_k^{\mathbf{C}_j} = \frac{\left|A^{\mathbf{C}_j} - \hat{A}_k^{\mathbf{C}_j}\right|}{A^{\mathbf{C}_j}}, \tag{13}$$

where $A^{\mathbf{C}_j}$ is the accuracy computed for the data composed of the features' indices in $\mathbf{C}_j$ and $\hat{A}_k^{\mathbf{C}_j}$ is the accuracy determined for the same data but when the values of the k-th feature are permuted across all records.

If $|\mathbf{C}_j| = 1$, there is no need to permute the values of a single feature over all input examples; then $w_k^{\mathbf{C}_j} = 1$. In (12), $\mathbf{X}_1 = \mathbf{P}$, $\mathbf{X}_2 = \mathbf{R}$, $\mathbf{X}_3 = \mathbf{T}$ and $k = 1, \ldots, |\mathbf{C}_j|$. For any $\mathbf{X}_s\left\{C_{jk}\right\} > |\mathbf{T}|$, the s-th summand is not considered in computing $R_{C_{jk}}$. Adding 1 ensures the assignment of the score from the set $\{1, \ldots, |\mathbf{T}|\}$ for each s. The final weighted combined ranking set $\boldsymbol{\Gamma}_j$ for common feature indices stored in $\mathbf{C}_j$ is established as $\boldsymbol{\Gamma}_j = \mathbf{C}_j(\boldsymbol{\Theta}_j)$ where $\boldsymbol{\Theta}_j$ is a collection of k-indices sorted according to descending order $R_{C_{jk}}$'s.

The next step is to consider whether or not all features included in $\boldsymbol{\Gamma}_j$ constitute an optimal set of attributes. In other words, there could be a subset of $\boldsymbol{\Psi}_j^\gamma \subset \boldsymbol{\Gamma}_j$, $\gamma = 1, \ldots, |\boldsymbol{\Gamma}_j|$ for which a given classifier achieves a higher accuracy. For this purpose, it is desirable to successively increase $\boldsymbol{\Psi}_j^\gamma$ and determine classifier performance on data composed of $\boldsymbol{\Psi}_j^\gamma$. The aforementioned can be simply achieved as follows: Let $\boldsymbol{\Psi}_j^0 = \emptyset$. Now, the determination of $\boldsymbol{\Psi}_j^\gamma = \boldsymbol{\Psi}_j^{\gamma-1} \cup \Gamma_{j\gamma}$, for each γ, allows for computing the accuracy $A_{j\gamma}$ on the data set consisting of the increasing subset of features in $\boldsymbol{\Psi}_j^\gamma$. In this way, we can sooner detect a higher classifier performance. Finally, finding both $\gamma^\star$ and $j^\star$ parameters, optimal in terms of the highest accuracy, yields

$$\boldsymbol{\Psi}_{j^\star}^{\gamma^\star} = \arg \max_{\substack{j=1,\ldots,|\mathbf{T}|,\\ \gamma=1,\ldots,|\boldsymbol{\Gamma}_j|}} A_{j\gamma}. \tag{14}$$

The method is summarized in the form of the pseudocode in Algorithm 1. This algorithm is composed of two parts. In the first (Steps 1–12), we compute the final weighted combined ranking set $\boldsymbol{\Gamma}_j$. The second part of the algorithm, embraced by Steps 13–19, leads to determining an optimal subset of features extracted from $\boldsymbol{\Gamma}_j$.

Algorithm 1. Weighted wrapper approach to feature selection.

```
Require: Input data $\langle \mathbf{x}_l, t_l \rangle$ for $l = 1, \ldots, L$
 1: Determine ordered sets of indices $\mathbf{P}$, $\mathbf{R}$ and $\mathbf{T}$ using
    filter methods: PC, RF and the DT
 2: for $j = 1$ to $|\mathbf{T}|$ do
 3:     Select first $j$ features' indices: $\mathbf{P}_j$, $\mathbf{R}_j$ and $\mathbf{T}_j$
 4:     Provide the set of common features' indices $\mathbf{C}_j$
        according to (11)
 5:     Determine accuracy $A^{\mathbf{C}_j}$
 6:     for $k = 1$ to $|\mathbf{C}_j|$ do
 7:         Determine accuracy $\hat{A}_k^{\mathbf{C}_j}$
 8:         Calculate weight $w_k^{\mathbf{C}_j}$ according to (13)
 9:         Yield weighted combined ranking score $R_{C_{jk}}$
            according to (12)
10:     end for
11:     Assign to vector $\boldsymbol{\Theta}_j$ $k$-indices from $R_{C_{jk}}$'s sorted
        in descending order
12:     $\boldsymbol{\Gamma}_j = \mathbf{C}_j(\boldsymbol{\Theta}_j)$
13:     $\boldsymbol{\Psi}_j^0 = \emptyset$
14:     for $\gamma = 1$ to $|\boldsymbol{\Gamma}_j|$ do
15:         $\boldsymbol{\Psi}_j^\gamma = \boldsymbol{\Psi}_j^{\gamma-1} \cup \Gamma_{j\gamma}$
16:         Determine accuracy $A_{j\gamma}$
17:     end for
18: end for
19: return $\boldsymbol{\Psi}_{j^\star}^{\gamma^\star}$ according to (14)
```

The proposed approach that provides an optimal (in terms of classification accuracy) subset of ordered feature indices $\boldsymbol{\Gamma}_j$ is henceforth called a $\boldsymbol{\Gamma}$ method or simply $\boldsymbol{\Gamma}$.

5. Case study

In this section, we apply the $\boldsymbol{\Gamma}$ method to the exemplary diagnostic breast cancer (DBC) database (Dua and Graff, 2017). DBC consists of 30 features, and currently holds the smallest number of attributes among all examined data sets. This will result in a simplification of the analysis.

Consider the 1D–CNN classifier and the sets of attribute indices ordered with respect to their significance provided by the PC, RF and DT methods for the DBC data set (Table 1). As shown, only 12 features are specified by the DT as significant since the tree chooses only 12 attributes as nodes and rejects the remaining 18 variables.

Let us regard $j = 1$ first feature indices provided by each FS method: $\mathbf{P}_1 = \{28\}$, $\mathbf{R}_1 = \{22\}$ and $\mathbf{T}_1 = \{21\}$. Then, according to (11), cf. Step 4 of the algorithm

$$\begin{aligned}\mathbf{C}_1 &= [(\mathbf{P}_1 \cup \mathbf{R}_1) \cap \mathbf{T}] \cup \mathbf{T}_1 \\ &= [(28 \cup 22) \cap \mathbf{T}] \cup 21 = \{21, 22, 28\}.\end{aligned}$$

The 1D–CNN's accuracy for the data set composed of the

Table 1. Indices of the attributes of the DBC database ordered with respect to their significance provided in the sets $\mathbf{P}$, $\mathbf{R}$ and $\mathbf{T}$. The column denoted as N presents the ranking position of the variable starting from the most significant.

N	$\mathbf{P}$	$\mathbf{R}$	$\mathbf{T}$	N	$\mathbf{P}$	$\mathbf{R}$
1	28	22	21	13	11	13
2	23	25	28	14	13	1
3	8	2	22	15	14	17
4	21	21	8	16	22	14
5	3	27	2	17	25	19
6	24	23	11	18	29	3
7	1	28	24	19	2	4
8	4	29	25	20	18	12
9	7	8	5	21	5	18
10	27	7	6	22	9	30
11	6	24	14	23	30	26
12	26	11	15	24	16	16
				25	17	5
				26	20	20
				27	19	15
				28	12	6
				29	10	10
				30	15	9

features from the set $\mathbf{C}_1$ is equal to $A^{\mathbf{C}_1} = 0.9666$.[1] Now, for $k = 1, \dots, |\mathbf{C}_1|$, each k-th feature is permuted across all records. For $k = 1$ (Feature 21), the 1D–CNN yields $\hat{A}_1^{\mathbf{C}_1} = 0.9121$ and therefore (Step 8 of the algorithm)

$$\begin{aligned} w_1^{\mathbf{C}_1} &= \frac{|A^{\mathbf{C}_1} - \hat{A}_1^{\mathbf{C}_1}|}{A^{\mathbf{C}_1}} \\ &= \frac{|0.9666 - 0.9121|}{0.9666} = 0.0564. \end{aligned}$$

In the case when $k = 2$ (Feature 22 is only permuted over all the records), we obtain $\hat{A}_2^{\mathbf{C}_1} = 0.9455$ and $w_2^{\mathbf{C}_1} = 0.0211$. If $k = 3$ (the 28th attribute is solely permuted across the patterns), one obtains $\hat{A}_3^{\mathbf{C}_1} = 0.9367$ and $w_3^{\mathbf{C}_1} = 0.0309$. Having $\mathbf{w}^{\mathbf{C}_1} = \{0.0564, 0.0211, 0.0309\}$ and $|\mathbf{T}| = 12$, the weighted combined ranking score (12) for each feature in $\mathbf{C}_1$ can be obtained (Step 9 of the algorithm). In particular, for Feature 21, it takes the value of

$$\begin{aligned} R_{C_{11}} &= w_1^{\mathbf{C}_1} \cdot [12 - \mathbf{P}\{C_{11}\} + 1 \\ &\quad + 12 - \mathbf{R}\{C_{11}\} + 1 + 12 - \mathbf{T}\{C_{11}\} + 1] \\ &= 0.0564 \cdot [12 - \mathbf{P}\{21\} + 1 \\ &\quad + 12 - \mathbf{R}\{21\} + 1 + 12 - \mathbf{T}\{21\} + 1] \\ &= 0.0564 \cdot (12 - 4 + 1 \\ &\quad + 12 - 4 + 1 + 12 - 1 + 1) \\ &= 1.6920. \end{aligned}$$

Similarly, in the case of Features 22 and 28, we get $R_{C_{12}} = 0.4642$ and $R_{C_{13}} = 0.8961$, respectively. Thus, after sorting the weighted ranking scores $R_{C_{11}}$, $R_{C_{12}}$ and $R_{C_{13}}$ in descending order, the sequence of $k = 3$ indices equal to $\mathbf{\Theta}_1 = \{1, 3, 2\}$ imposes $\mathbf{\Gamma}_1 = \mathbf{C}_1(\mathbf{\Theta}_1) = \{21, 28, 22\}$, as indicated in Step 12 of the algorithm. Computing $\mathbf{\Gamma}_1$ for $j = 1$ finishes the first part of the algorithm. Table 2 presents the attribute indices stored in $\mathbf{\Gamma}_j$ for $j = 1, \dots, 12$. We can see that some indices, which appear in the top rows of the table (i.e., are the most significant), change their ranking places. This results from the introduced weights.

Steps 13–19 of Algorithm 1 realize the choice of the feature subset $\mathbf{\Psi}_{j^\star}^{\gamma^\star}$ which is optimal in terms of $A_{j\gamma}, \forall_{j,\gamma}$. For the DBC data set, the highest accuracy is obtained for $\mathbf{\Gamma}_5$ where only 4 features are involved in representing the input records; therefore $\mathbf{\Psi}_5^4 = \{21, 22, 25, 2\}$, which is shown in boldface in Table 2.

In Fig. 1, we demonstrate the impact of j and the number of the elements in the ranking sets $\mathbf{\Gamma}_j$ on the accuracy of a 1D–CNN in a three-dimensional visualization (top drawing) and a plane projection (bottom drawing). The plot presenting the best network's performance is marked with a dashed line. Each dependency is created by successive inclusion of a single less significant feature into both subsets.

6. Experimental results

This section shortly highlights the input data sets, the decision tree parameters and the 1D–CNN configuration used in the experiments. However, the main emphasis of this part of the article is placed on the analysis of the results obtained after the application of the $\mathbf{\Gamma}$ method and wrapper approaches SFS and SBS in the task of reducing the analyzed data sets. All the results are expressed in terms of accuracy (10) computed with the use of a 10-fold cross validation procedure. The simulations are repeated 10 times, and all results are averaged.

6.1. Data sets. For the purpose of the experimental analysis, six machine learning repository (UCI–MLR) data sets the of the University of California, Irvine, are used:

- Sports articles—a collection of articles identified by a crowd sourcing website (Hajj *et al.*, 2019): 1000 cases, 57 attributes, 2 classes;

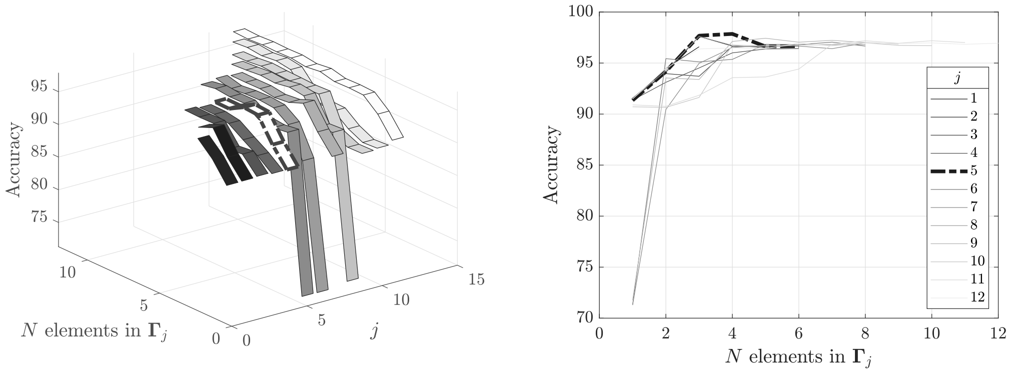
- Diagnostic breast cancer (DBC)—a set of characteristics of the cell nuclei in the image (Dua and Graff, 2017): 569 cases, 30 attributes, 2 classes.

- QSAR—a database of molecular descriptors used to classify chemicals (Mansouri *et al.*, 2013): 1055

[1] The accuracy values provided in this example are real and computed as the results of the experiments for the DBC data set.

Table 2. Weighted combined ranking sets for the indices of particular common features stored in $\mathbf{\Gamma}_j$ for the DBC data set. The subset of indices $\mathbf{\Psi}_5^4$ for which the maximum accuracy of a 1D–CNN is determined is marked with boldface.

	N	$\mathbf{\Gamma}_1$	$\mathbf{\Gamma}_2$	$\mathbf{\Gamma}_3$	$\mathbf{\Gamma}_4$	$\mathbf{\Gamma}_5$	$\mathbf{\Gamma}_6$	$\mathbf{\Gamma}_7$	$\mathbf{\Gamma}_8$	$\mathbf{\Gamma}_9$	$\mathbf{\Gamma}_{10}$	$\mathbf{\Gamma}_{11}$	$\mathbf{\Gamma}_{12}$
Indices	1	21	21	21	21	**21**	25	25	21	22	28	28	21
	2	28	22	2	8	**22**	21	8	22	8	25	25	28
	3	22	25	22	28	**25**	28	24	25	25	2	8	2
	4		28	28	2	**2**	8	2	28	24	21	2	25
	5			25	22	28	22	22	2	21	6	22	11
	6			8	25	8	24	11	8	28	8	11	22
	7						2	28	24	5	22	24	8
	8						11	21	11	11	24	21	6
	9									2	5	6	24
	10										11	5	14
	11											14	5
	12												15



(a) (b)

Fig. 1. Changes in a 1D–CNN's performance in the classification of the DBC data set after application of the $\mathbf{\Gamma}$ method for feature selection. Plot (a) reveals the influence of both parameter j and the number of the elements in the ranking sets $\mathbf{\Gamma}_j$ on the network's accuracy, plot (b) is its 2-dimensional projection stressing the value of j for which the highest accuracy value is attained.

cases, 41 attributes, 2 classes;

- Spam base—a collection of spam e-mails (Dua and Graff, 2017): 4601 cases, 57 attributes, 2 classes;
- Statlog—a set of pixels values in a satellite image (Dua and Graff, 2017): 4435 cases, 36 attributes, 6 classes;
- Optical digits—a data set of normalized bitmaps of handwritten digits (Dua and Graff, 2017): 3823 cases, 62 attributes, 10 classes.

The characteristics of benchmark data sets (number of records, features and classes) are summarized in the second column of Table 3.

All $i = 1, \ldots, I$ data features are normalized to $[-1, 1]$ interval according to the formula

$$x'_{li} = 2\frac{x_{li} - x_{li}^{\min}}{x_{li}^{\max} - x_{li}^{\min}} - 1, \tag{15}$$

where x_{li}, $x_{li}^{\min}$ and $x_{li}^{\max}$ represent given, minimum and maximum feature values, respectively.

6.2. Neural network structure.

The 1D–CNN applied in this research has an experimentally selected structure of a single CONV layer and three fully connected layers. The final network's architecture is adjusted as follows:

- I-length input layer with zero-center normalization:

$$x_{li}^{\text{norm}} = \frac{x'_{li} - \mu_i}{\sigma_i}, \tag{16}$$

where μ_i and σ_i are respectively the mean and the standard deviation computed for the i-th feature;

- A convolution layer that consists of filters selected from the set $\{5, 10, 15, 20, 25\}$, where the filter sizes and the stride are equal to 4 and 1, respectively; the padding is set so as the output and the input signal have the same size; the ReLU function is used as activation;
- MLP layer with $4J$ neurons;
- MLP layer with $2J$ neurons;
- MLP layer with J neurons;
- the output layer that computes the final response based on the softmax transfer function.

The size of the output layer depends on the number of classes in the data set. Also, the number of neurons in all MLP layers is influenced by J. Due to the use of a single CONV layer, no pooling operation is applied.

6.3. Decision tree parameters. For the attribute selection by means of the decision tree, the CART algorithm (Breiman *et al.*, 1984) is applied. The following parameters are adjusted experimentally for tree growing: feature splitting criterion (Gini index, entropy), maximum tree depth: $\{3, 4, 5\}$, minimum number of leaf node observations: $[1, \max(2, \lfloor L/2 \rfloor)]$. No tree pruning procedure is applied.

6.4. Results and a discussion. According to the main idea behind the $\mathbf{\Gamma}$ method, for all six data sets, the value of j and the corresponding $\mathbf{\Gamma}_j$ subsets are determined for which the 1D–CNN's accuracy attains the highest value.

Figure 2 includes a graphical summary of all the experiments conducted on six databases. Each illustration presents the winning plot for $\mathbf{\Gamma}$, i.e., for such a value of j that contributes to the highest accuracy of a 1D–CNN. As reference, we delineate the characteristics for SFS and SBS methods. If we consider the proposed method, we can see that for each data set, the highest network's accuracy is always provided when the number of features is smaller than the dimensionality of the input set. This, in turn, can be significant in the context of a prediction for an unknown sample, since fewer attributes are required for the computation of the 1D–CNN's output signal. Next, one can also observe that it is always possible to find at least one point in the plot (i.e., a subset of features) where the accuracy of $\mathbf{\Gamma}$ method is higher than that for SFS and SBS. In fact, such an outcome is always obtained "earlier"—for a smaller number of the attributes. Finally, it is true that in five out of six cases, for a smaller number of features, the accuracy results for SFS and SBS are higher than those obtained by the proposed method, but these are still small enough values; therefore, they may be considered insignificant.

In Table 3, we present the highest accuracy values of a 1D–CNN obtained in all classification tasks when data features are established by SFS, SBS and the combined ranking subsets $\mathbf{\Gamma}_j$. To make the analysis comprehensive, the number of features found for a given outcome is added below. Based on the results, the following major remarks can be made:

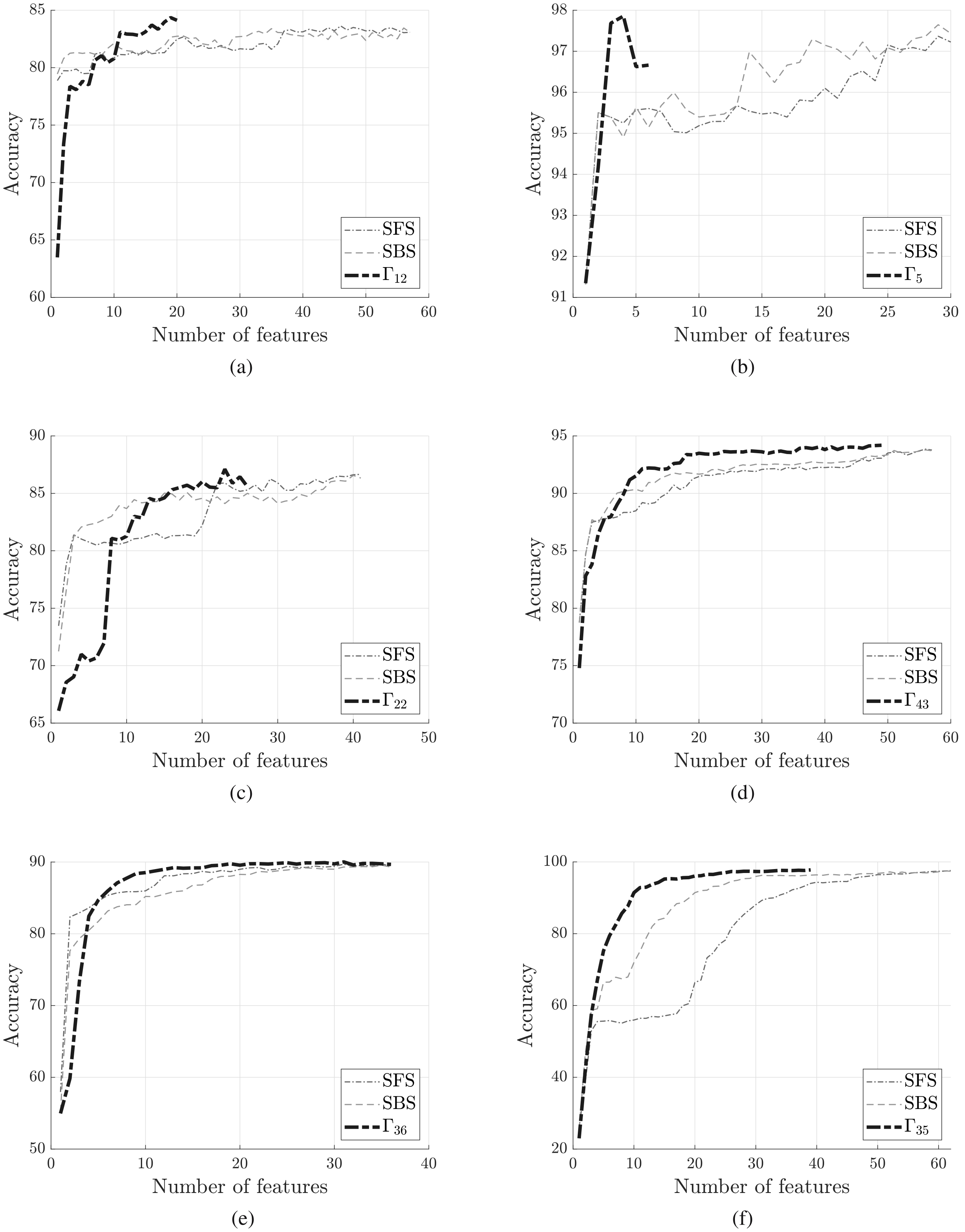
1. The 1D–CNN's accuracy obtained for the data sets with the attributes reduced by the $\mathbf{\Gamma}$ method is always higher than the accuracy computed in the original input space. This is not the case for the remaining methods.
2. The application of the $\mathbf{\Gamma}$ method makes a 1D–CNN attain the highest accuracy in all classification tasks.
3. The use of the $\mathbf{\Gamma}$ method results in the highest decrease in the input features for all data sets. The optimal subsets of features $\mathbf{\Psi}_{j^\star}^{\gamma^\star}$ for sports articles, DBC, QSAR, spam base, Statlog and optical digits data sets are represented by $\mathbf{\Psi}_{12}^{19}$, $\mathbf{\Psi}_{5}^{4}$, $\mathbf{\Psi}_{22}^{23}$, $\mathbf{\Psi}_{43}^{49}$, $\mathbf{\Psi}_{36}^{31}$ and $\mathbf{\Psi}_{35}^{39}$, respectively.

Table 3 also presents a statistical comparison evaluated by means of pairwise T-tests between the accuracies computed when the feature subsets are provided by the $\mathbf{\Gamma}$ method against the SFS, SBS and when all attributes are included. The tests are determined for 0.95 confidence intervals. The outcomes are revealed to be statistically significant in all but two classification cases which is indicated with the '$+'$ label placed next to the standard deviation value.

The very concise comments specified above confirm that, despite the availability of well-known FS methods, it is worth searching for other, new solutions which generate ordered attribute rankings. This can be realized twofold: either by merging state-of-the-art approaches, or by developing new strategies. The effectiveness of novel ideas does not have to be necessarily the highest. However, they must find a certain balance between being highly accurate and capable of reducing data dimensionality at the same time. This goal is achieved in this study.

7. Conclusions

The essence of the feature selection methods lies in the choice of the attribute subset that is the most representative for the entire database. Constraining the set of features to the smallest possible collection, while maintaining a high classification accuracy, is required by all FS methods. Numerous FS methods, both filters and wrappers, are known for not only producing a ranking,



(a) (b)

(c) (d)

(e) (f)

Fig. 2. Averaged accuracy values attained by a 1D–CNN for a particular number of input features in the classification of the data sets considered: sports articles (a), DBC (b), QSAR (c), spam base (d), Statlog (e) and optical digits (f). Each dependency illustrates, in succession, SFS, SBS and $\mathbf{\Gamma}$. For the proposed wrapper approach, the accuracy is plotted for the set of attributes stored in $\mathbf{\Gamma}_j$ where the value of j indicates the highest 1D–CNN performance.

Table 3. Highest accuracy (in %) achieved by a 1D–CNN in the classification of the examined databases with the attributes given by SFS, SBS and the proposed $\mathbf{\Gamma}$ method. The number of features for the presented result is shown below each outcome. The last column outlines the performance of the network on each data set with all features. The accuracies are averaged over 10 simulation runs; standard deviations are added alongside.

Data set	Input size (classes)	SFS	SBS	$\mathbf{\Gamma}$	All
Sports articles	1000 × 57 (2)	$83.63 \pm 0.35^{+}$ 46	$83.46 \pm 0.27^{+}$ 56	$\mathbf{84.36} \pm 0.33$ **19**	$83.00 \pm 0.43^{+}$ 57
DBC	569 × 30 (2)	$97.37 \pm 0.26^{+}$ 29	$97.65 \pm 0.15^{+}$ 29	$\mathbf{97.86} \pm 0.15$ **4**	$97.15 \pm 0.28^{+}$ 30
QSAR	1055 × 41 (2)	86.66 ± 0.62 41	$86.56 \pm 0.12^{+}$ 40	$\mathbf{87.11} \pm 0.38$ **23**	$86.58 \pm 0.55^{+}$ 41
Spam base	4601 × 57 (2)	$93.85 \pm 0.15^{+}$ 56	$93.78 \pm 0.12^{+}$ 56	$\mathbf{94.19} \pm 0.12$ **49**	$93.95 \pm 0.16^{+}$ 57
Statlog	4435 × 36 (6)	89.76 ± 0.42 35	$89.56 \pm 0.12^{+}$ 35	$\mathbf{90.02} \pm 0.29$ **31**	$89.50 \pm 0.36^{+}$ 36
Optical digits	3823 × 62 (10)	$97.57 \pm 0.08^{+}$ 62	$97.50 \pm 0.18^{+}$ 61	$\mathbf{97.77} \pm 0.08$ **39**	$97.58 \pm 0.23^{+}$ 62

but also selecting a subset of features. For example, the PC, RF and DT methods select attributes on the basis of various relationships between features and a given class. Their final outcomes are therefore reflected in different attribute rankings. However, since these state-of-the-art filters have the value of universality, the proper fusion of the rankings they provide can result in other forms of significance.

For this particular reason, we proposed a new feature selection solution, i.e., the $\mathbf{\Gamma}$ method. It belongs to the class of wrapper approaches and relies on assigning the weight coefficients to particular features based on the combined ranking criterion that involves sub-ranking generated by PC, RF and the DT. The weights were established on the basis of the accuracy determined by a one-dimensional convolutional neural network throughout successive variable permutations across all input records. On the one hand, the use of the DT allowed us to constrain the sizes of feature collections; on the other hand, the application of PC and RF yielded high classification correctness.

In this work, the experimental analysis consisted in computing the 1D–CNN's accuracy on six, high dimensional repository databases for which the features were reduced by the proposed method and state-of-the-art sequential attribute selection procedures. As a point of reference, the classification results obtained by the network in the original input space were taken into account. The effectiveness of the feature ranking methods was estimated by comparing not only the highest correctness of classification, but also the number of the attributes where the best performance was observed. The presented results have demonstrated that the $\mathbf{\Gamma}$ method achieved top results in all cases in the context of the accuracy, as well as the smallest number of the input features.

Future work will focus on incorporating feature selection into reduction of the data records so that the entire input space is to be decreased in size. Both approaches will work in parallel, and this will allow us to establish representative and pruned collections of available samples.

Acknowledgment

This work is partially financed by the Polish Ministry of Science and Higher Education under the program *Regional Initiative of Excellence* in 2019–2022, project no. 027/RID/2018/19, and by the Rzeszów University of Technology, within the subsidy for the maintenance and development of research potential (UPB).

References

Abdel-Hamid, O., Mohamed, A.-R., Jiang, H., Deng, L., Penn, G. and Yu, D. (2014a). Convolutional neural networks for speech recognition, *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **22**(10): 1533–1545.

Abdel-Hamid, O., Mohamed, A.-R., Jiang, H., Deng, L., Penn, G. and Yu, D. (2014b). Convolutional neural networks for speech recognition, *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **22**(10): 1533–1545.

Abdeljaber, O., Avci, O., Kiranyaz, M.S., Boashash, B., Sodano, H. and Inman, D.J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data, *Neurocomputing* **275**: 1308–1317.

Abdeljaber, O., Avci, O., Kiranyaz, S., Gabbouj, M. and Inman, D.J. (2017). Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, *Journal of Sound and Vibration* **388**: 154–170.

Awada, W., Khoshgoftaar, T.M., Dittman, D., Wald, R. and Napolitano, A. (2012). A review of the stability of feature selection techniques for bioinformatics data, *IEEE 13th International Conference on Information Reuse & Integration (IRI), Las Vegas, USA*, pp. 356–363.

Azizjon, M., Jumabek, A. and Kim, W. (2020). 1D CNN based network intrusion detection with normalization on imbalanced data, *International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, Japan*, pp. 218–224.

Benesty, J., Chen, J., Huang, Y. and Cohen, I. (2009). Pearson correlation coefficient, *in* J. Benesty and W. Kellermann (Eds.), *Noise Reduction in Speech Processing*, Springer Topics in Signal Processing, Springer, Berlin, pp. 1–4.

Bolón-Canedo, V., Sánchez-Maroño, N. and Alonso-Betanzos, A. (2013). A review of feature selection methods on synthetic data, *Knowledge and Information Systems* **34**(3): 483–519.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees*, CRC Press, Boca Raton.

Broughton, R., Coope, I., Renaud, P. and Tappenden, R. (2010). Determinant and exchange algorithms for observation subset selection, *IEEE Transactions on Image Processing* **19**(9): 2437–2443.

Cannas, L.M., Dessì, N. and Pes, B. (2013). Assessing similarity of feature selection techniques in high-dimensional domains, *Pattern Recognition Letters* **34**(12): 1446–1453.

Devijver, P. and Kittler, I. (1982). *Pattern Recognition: A Statistical Approach*, Prentice-Hall, Englewood Cliffs.

Dua, D. and Graff, C. (2017). *UCI Machine Learning Repository,* `http://archive.ics.uci.edu/ml`.

El Aboudi, N. and Benhlima, L. (2016). Review on wrapper feature selection approaches, *International Conference on Engineering & MIS (ICEMIS), Agadir, Morocco*, pp. 1–5.

Eren, L. (2017). Bearing fault detection by one-dimensional convolutional neural networks, *Mathematical Problems in Engineering* **2017**: 1–9.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection, *Journal of Machine Learning Research* **3**: 1157–1182.

Hajj, N., Rizk, Y. and Awad, M. (2019). A subjectivity classification framework for sports articles using cortical algorithms for feature selection, *Neural Computing and Applications* **31**: 8069–8085.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M. and Inman, D.J. (2021). 1D convolutional neural networks and applications: A survey, *Mechanical Systems and Signal Processing* **151**: 107398.

Kiranyaz, S., Ince, T. and Gabbouj, M. (2015a). Real-time patient-specific ECG classification by 1-D convolutional neural networks, *IEEE Transactions on Biomedical Engineering* **63**(3): 664–675.

Kiranyaz, S., Ince, T. and Gabbouj, M. (2015b). Real-time patient-specific ECG classification by 1-D convolutional neural networks, *IEEE Transactions on Biomedical Engineering* **63**(3): 664–675.

Kohavi, R. and John, G.H. (1997). Wrappers for feature subset selection, *Artificial Intelligence* **97**(1): 273–324.

Koziarski, M. and Cyganek, B. (2018). Impact of low resolution on image recognition with deep neural networks: An experimental study, *International Journal of Applied Mathematics and Computer Science* **28**(4): 735–744, DOI: 10.2478/amcs-2018-0056.

Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2017). Imagenet classification with deep convolutional neural networks, *Communications of the ACM* **60**(6): 84–90.

Kusy, M., Zajdel, R., Kluska, J. and Zabinski, T. (2020). Fusion of feature selection methods for improving model accuracy in the milling process data classification problem, *International Joint Conference on Neural Networks (IJCNN), Glasgow, UK*, pp. 1–8.

LeCun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning, *Nature* **521**(7553): 436–444.

LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE* **86**(11): 2278–2324.

Li, Y., Hsu, D.F. and Chung, S.M. (2013). Combination of multiple feature selection methods for text categorization by using combinatorial fusion analysis and rank-score characteristic, *International Journal on Artificial Intelligence Tools* **22**(02): 1350001.

Lu, J., Zhao, T. and Zhang, Y. (2008). Feature selection based-on genetic algorithm for image annotation, *Knowledge-Based Systems* **21**(8): 887–891.

Mansouri, K., Ringsted, T., Ballabio, D., Todeschini, R. and Consonni, V. (2013). Quantitative structure–activity relationship models for ready biodegradability of chemicals, *Journal of Chemical Information and Modeling* **53**(4): 867–878.

Narendra, P.M. and Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection, *IEEE Transactions on Computers* **26**(09): 917–922.

Pes, B. (2020). Ensemble feature selection for high-dimensional data: A stability analysis across multiple domains, *Neural Computing and Applications* **32**(10): 5951–5973.

Robnik-Šikonja, M. and Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF, *Machine Learning* **53**(1–2): 23–69.

Rodrigues, D., Pereira, L.A., Nakamura, R.Y., Costa, K.A., Yang, X.-S., Souza, A.N. and Papa, J.P. (2014). A wrapper approach for feature selection based on bat algorithm and optimum-path forest, *Expert Systems with Applications* **41**(5): 2250–2258.

Rokach, L., Chizi, B. and Maimon, O. (2006). Feature selection by combining multiple methods, *in* M. Last *et al.* (Eds), *Advances in Web Intelligence and Data Mining*, Springer, Berlin/Heidelberg, pp. 295–304.

Russell, S. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs.

Scherer, D., Müller, A. and Behnke, S. (2010). Evaluation of pooling operations in convolutional architectures for object recognition, *International Conference on Artificial Neural Networks, Thessaloniki, Greece*, pp. 92–101.

Vergara, J.R. and Estévez, P.A. (2014). A review of feature selection methods based on mutual information, *Neural Computing and Applications* **24**(1): 175–186.

Wang, Y., Zhang, D. and Dai, G. (2020). Classification of high resolution satellite images using improved U-Net, *International Journal of Applied Mathematics and Computer Science* **30**(3): 399–413, DOI: 10.34768/amcs-2020-0030.

Whitney, A.W. (1971). A direct method of nonparametric measurement selection, *IEEE Transactions on Computers* **100**(9): 1100–1103.

Wuniri, Q., Huangfu, W., Liu, Y., Lin, X., Liu, L. and Yu, Z. (2019). A generic-driven wrapper embedded with feature-type-aware hybrid Bayesian classifier for breast cancer classification, *IEEE Access* **7**: 119931–119942.

Zajdel, R., Kusy, M., Kluska, J. and Zabinski, T. (2020). Weighted feature selection method for improving decisions in milling process diagnosis, *in* L. Rutkowski *et al.* (Eds), *Artificial Intelligence and Soft Computing,* Lecture Notes in Computer Science, Vol. 12415, Part I, Springer, Cham, pp. 280–291.

Maciej Kusy received his MSc degree in electrical engineering from the Rzeszów University of Technology, Poland, in 2000, his PhD degree in biocybernetics and biomedical engineering from the Warsaw University of Technology, Poland, in 2008, and his DSc degree in information and communication technology from the Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland, in 2019. He is an associate professor with the Faculty of Electrical and Computer Engineering, Rzeszów University of Technology. His current research interests include computational and artificial intelligence machine learning as well as data mining.

Roman Zajdel holds a PhD degree (1999) and a DSc degree (2019) in computer science from the Wrocław University of Technology. He is an associate professor at the Institute of Control and Computer Engineering, Rzeszów University of Technology, Poland. His current research interests concentrate on reinforcement learning, fuzzy logic and machine learning.

Received: 24 February 2021
Revised: 19 July 2021
Re-revised: 22 September 2021
Accepted: 4 October 2021