Comparison of hybrid binary GWO-PSO algorithm with feature selection methods by using machine learning classifiers

Buğra Kaan TİRYAKİ[*]

Karadeniz Technical University Faculty of Science,
Department of Computer Science, Trabzon.

Geliş Tarihi (Received Date): 17.04.2024
Kabul Tarihi (Accepted Date): 04.12.2024

Abstract

In the field of machine learning, feature selection methods used in the pre-processing of data for the classifier have become very popular. Instead of the whole dataset, it is important to create a new sub-dataset by discarding the irrelevant and redundant variables in the dataset to make the data ready for analysis. In this way, both the performance of the learning classifier will increase, and cost and time savings will be achieved. In this study, the performance of the hybrid binary grey wolf optimization - particle swarm optimization (BHGWOPSO) algorithm with machine learning methods is investigated. In addition, a comparison was made between BHGWOPSO and other feature selection methods such as principial component analysis and filter methods in contrast to literature. Thus, it is aimed to show which of the different feature selection methods will work better. For this purpose, five different benchmark datasets with different number of features were selected. Both feature selection methods and machine learning classifiers were compared with each other using the accuracy metric. As a result of the comparisons, it was observed that a different feature selection method and a different classifier had higher accuracy values for each data set.

Keywords: *Binary hybrid optimization, feature selection, learning classifiers, wrapper method*

[*]Buğra Kaan TİRYAKİ, btiryaki@ktu.edu.tr, http://orcid.org/0000-0003-0995-7389

Hibrit ikili GWO-PSO algoritmasının makine öğrenmesi sınıflandırıcıları kullanılarak özellik seçim yöntemleriyle karşılaştırılması

Öz

Makine öğrenmesi alanında sınıflandırıcı için verinin ön işlemesinde kullanılan değişken seçme yöntemleri oldukça popüler bir hale gelmiştir. Tüm veri seti yerine, veri setindeki değişkenlerden ilgisiz ve gereksiz olanların atılarak yeni bir alt veri kümesi oluşturulması veriyi analize hazır hale getirmek için önemlidir. Bu sayede öğrenme sınıflandırıcının hem performansı artacak hem de maliyet ve zaman bakımından tasarruf sağlanabilecektir. Bu çalışmada hibrit ikili gri kurt optimizasyon-parçacık sürü optimizasyon (BHGWOPSO) algoritmasının makine öğrenmesi yöntemleriyle performansı araştırılmıştır. Ayrıca simülasyonlarda literatürden farklı olarak BHGWOPSO ile diğer özellik seçim yöntemlerinden temel bileşen analizi ve filtre yöntemler kullanılarak da karşılaştırma yapılmıştır. Böylelikle farklı özellik seçim yöntemlerinin hangisinin daha iyi çalışacağının gösterilmesi amaçlanmıştır. Bu amaçla farklı özellik sayılarına sahip beş farklı ölçüt veri seti seçilmiştir. Hem özellik seçim yöntemleri hem de makine öğrenmesi sınıflandırıcıları birbirleriyle doğruluk metriği kullanılarak karşılaştırılmıştır. Karşılaştırmalar sonucunda her bir veri seti için farklı bir özellik seçim yöntemin ve farklı bir sınıflandırıcının daha yüksek doğruluk değerine sahip olduğu görülmüştür.

Anahtar kelimeler: *İkili hibrit optimizasyon, özellik seçimi, öğrenme sınıflandırıcıları, sarmal yöntem*

1. Introduction

Today, the collection and storage of data are increasing day by day. Processing this large amount of data collected and drawing meaningful conclusions from it has become very important. It is sometimes impossible to work with all variables in large data sets. Instead, researchers may prefer to make inferences with fewer variables. Therefore, eliminating irrelevant and redundant features from the dataset is important. Irrelevant and redundant variables can increase computation time and negatively affect classification accuracy. Numerous datasets currently in existence comprise thousands, occasionally even tens of thousands of features. Achieving higher success with less data and variables will naturally require less time and cost [1]. In this case, it can be said that the aim of feature selection is to exponentially reduce the size of the hypothesis space [2].

Data mining will rise in tandem with cloud computing and the internet of things. This technique is used in various fields, including biology, finance, geography, astrophysics and various applications, including microarray analysis, recommender systems for financial data with high frequency, text categorization, detection of epileptic seizures, face identification, cancer classification, gene classification, and customer relationship management [1,3]. In these domains, limited training samples and high dimensionality data frequently compromise statistical significance. A pre-processing step called feature selection improves user interpretation by reducing training time and data dimensionality by removing noise and overfitting [2].

Feature selection methods are used to process fewer variables from big data. Especially, feature selection methods for learning classifiers have become very popular. Since choosing an optimal or sub-optimal subset of features is crucial to a machine-learning technique's effectiveness, several strategies have been developed in this area [4]. Since a subset of data is created from the same dataset, variable selection in the dataset is also called feature subset selection (FSS) in some sources.

In general, the steps in a simple machine learning process are shown in Figure 1. The feature selection part that needs to be done before applying the machine learning method is dealt with in this study. This part is also very important in machine learning. In this step, as in the preprocessing of the dataset, it is aimed to use the dataset more effectively by selecting the features of the dataset. For example, accuracy can be increased in the same machine learning classifier by consuming less cost and time.

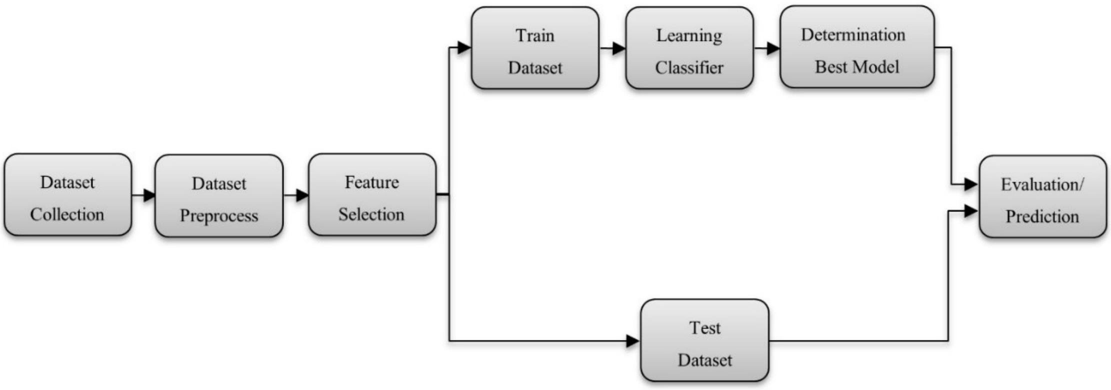

Figure 1. Workflow of a basic machine learning classification

Comparative studies of feature selection exist in the literature. Particle swarm optimization and genetic algorithms were examined by Talbi et al. for feature selection in classification [5]. They used a variety of cancer datasets which called high dimensional microarray data. Xue et al. provided an inclusive guideline on the strengths and weaknesses of feature selection [6]. For feature selection, they retailed comprehensive categories of evolutionary computation. An iterative deterministic local optimization technique that can be combined with the selection of wrapper or filter features was proposed by Zarshenas and Suzuki [4]. They contrasted their feature optimization technique for Naïve Bayes, multilayer perceptrons, and support vector machines. By using an innovative local search approach, Moradi and Gholampour suggested a new hybrid particle swarm optimization for feature subset selection [3]. They showed their new algorithm results on benchmark datasets and compared them with feature selection methods. Nekkaa and Boughaci presented a hybrid search strategy for feature selection that relies on both the stochastic local search and the harmony search algorithm [7]. They also included a support vector machine classifier with optimal parameters in their combination. Comparative experimental results were included in their study. The most recent feature selection techniques created for and used in medical issues such DNA microarray data analysis, biomedical signal processing, and medical imaging were discussed by Remeseiro and Bolon-Canedo [8]. Ghosh et al. developed a method which performs feature selection by gathering information from the candidate solutions generated by genetic algorithm and particle swarm optimization [9]. They combined the exploitation capability of genetic algorithm with the exploration capability of particle swarm optimization. In order to address feature selection issues, Al-Tashi et al. suggested a binary variant of the hybrid grey wolf optimization and particle swarm optimization [10]. They compared their algorithm to other wrapper methods. El-Kenawy and Eid proposed hybrid gray wolf and particle swarm optimization for feature selection [11]. They compared wrapper methods with their algorithm. Also, they evaluated their algorithm performance with wrapper methods. A wrapper-based feature selection approach was presented by Allam and Malaiyappan that selects the best features from dataset attributes by utilizing various tutors [12]. Their goal is to explore the complete solution space without coming upon the local optimal feature set. Also, they compared their proposed method with different wrapper-based feature selection techniques. In order to perform the feature subset selection process, Sameer presented a new modified binary teaching-learning based optimization and showed that support vector machine (SVM) binary identification is accurate when used as a fitness function [13].

Al-Tashi et al. have demonstrated the performance of their proposed binary hybrid grey wolf and particle swarm optimization algorithm (BHGWOPSO) with only k-nearest neighbour (KNN) among machine learning methods [10], and their experimental results with other hybrid methods are mentioned, but filter methods in feature selection are not mentioned. In addition, as suggested in the conclusion and future work section of their article, it is mentioned that the experimental analysis of their proposed method can be compared with other popular machine learning classifiers such as support vector machine (SVM) and artificial neural network (ANN). The motivation of this study comes from the fact that there is no experimental study with two other popular classifiers, such as SVM and ANN, using BHGWOPSO for feature selection. In this study, unlike the literature, a binary hybrid optimization method, BGWOPSO, is compared with benchmark data sets to measure the performance of BGWOPSO with both other feature selection methods and machine learning methods.

2. Material and method

In this section, the feature selection methods and machine learning classifiers that will be used to demonstrate the performance of the BHGWOPSO method will be discussed. These are given in two sub-headings as feature selection methods and machine learning classifiers.

2.1. Feature selection methods

The methods used in feature selection are divided into three categories: principal component analysis (PCA), filter and wrapper methods.

2.1.1. Principal component analysis (PCA)

It is one of the most basic approaches used in feature selection. It is also one of the multivariate statistical methods used to reduce data size. The aim of principal component analysis is to reach from a d-dimensional space to a new k-dimensional space with minimum data loss. PCA is an unsupervised technique that doesn't make use of output data [14].

A data table containing observations defined by several dependent variables that are typically connected is analyzed using PCA. Its goal is to take significant information out of the data table and put it into a new collection of variables known as principal components. By showing the variables and observations as points on maps, PCA also depicts the similarity pattern of the data [15].

2.1.2. Filter methods

The filter method frequently uses correlation between variables and is based on statistical and probabilistic techniques. Filter models choose features without regard to estimators by using broad properties of the training set. As general properties of the training data, consider distance, consistency, fuzzy-set, rough-set, and statistical methodology [16]. This method is faster than the wrapper approach because it does not use a learning classifier. However, it tends to select many subsets. Therefore, a performance criterion (threshold value) is needed to select a subset or to select the highest ranked features. Filter methods based on ranked scores are an efficient way to pick fewer variables instead of selecting variables from the complete dataset. When there are redundant variables in the data, feature selection techniques based on ranked features might identify possibly important inferior characteristics [17].

The filter methods selected in this study are minimum redundancy maximum relevance (MRMR), ReliefF, chi-square tests (Chi2). Supported data types for MRMR and Chi2 are categorical and continuous features, but ReliefF supports data types such as either all categorical or all continuous features.

2.1.3. Wrapper methods

Wrapper techniques utilize machine learning techniques to identify dataset subsets. They aim to improve accuracy and efficiency with the help of machine learning classifiers. In general, wrapper models aim to optimize a predictor. Nonetheless, research indicates that wrapper-based techniques outperform filter-based techniques [17,18].

In wrapper techniques, choosing a feature subset is an NP-hard task. Metaheuristic algorithms are one method of solving NP-hard issues. One type of metaheuristic algorithms that works well for feature selection issues is swarm intelligence techniques. Some of the swarm intelligence-based techniques utilized in feature selection challenges are whale optimization (WO), salp swarm optimization (SSO), gray wolf optimization (GWO), and particle swarm optimization (PSO). Combining these approaches has led to the development of hybrid methodologies in certain investigations. This study uses a hybrid optimization method for feature selection called the binary version of the hybrid GWO-PSO optimization algorithm.

Al-Tashi et al. developed the BHGWOPSO algorithm. It operates in binary space because it is a feature selection. The BHGWOPSO algorithm functions generally as follows. Position update is carried out using the formula below. The PSO and GWO algorithms are combined to model it as follows [10,11].

$$x_d^{t+1} = \begin{cases} 1 & \text{if sigmoid}\left(\frac{\mathrm{x}_1+\mathrm{x}_2+\mathrm{x}_3}{3}\right) \geq \text{rand} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

In the context of binary search, x_d^{t+1} represents the updated position at iteration t in dimension d. The term "rand" denotes a random number drawn from a uniform distribution between 0 and 1. Equation 2 can be used to determine the sigmoid function.

$$sigmoid(a) = \frac{1}{1+e^{-10(x-0.5)}} \tag{2}$$

x_1^d, x_2^d and x_3^d are defined by equations (3), (4) and (5), respectively.

$$x_1^d = \begin{cases} 1 & \text{if } \left(\mathrm{x}_\alpha^\mathrm{d} + \mathrm{bstep}_\alpha^\mathrm{d}\right) \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$$x_2^d = \begin{cases} 1 & \text{if } \left(\mathrm{x}_\beta^\mathrm{d} + \mathrm{bstep}_\beta^\mathrm{d}\right) \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$x_3^d = \begin{cases} 1 & \text{if } \left(\mathrm{x}_\delta^\mathrm{d} + \mathrm{bstep}_\delta^\mathrm{d}\right) \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Equations (6-7) are also used to define step functions.

$$\mathrm{cstep}_{\alpha,\beta,\delta}^\mathrm{d} = \frac{1}{1+e^{-10\left(A_1^d D_{\alpha,\beta,\delta}^d - 0.5\right)}} \tag{6}$$

$$\mathrm{bstep}_{\alpha,\beta,\delta}^\mathrm{d} = \begin{cases} 1 & \text{if } \mathrm{cstep}_{\alpha,\beta,\delta}^\mathrm{d} \geq \mathrm{rand} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Here, α, β and δ parameters are employed in order to simulate the leadership hierarchy of the GWO. The exploration and exploitation of the grey wolf are mathematically modelled as $\vec{D}_\alpha$, $\vec{D}_\beta$, $\vec{D}_\delta$ which are updated as in equation (8).

$$\vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - w * \vec{X}\right|, \quad \vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - w * \vec{X}\right|, \quad \vec{D}_\delta = \left|\vec{C}_\delta \cdot \vec{X}_\alpha - w * \vec{X}\right| \tag{8}$$

Equation (9) can be used to compute a particle's velocity.

$$v_i^{k+1} = w * \left(v_i^k + c_1 r_1\left(x_1 - x_i^k\right) + c_2 r_2\left(x_2 - x_i^k\right) + c_3 r_3\left(x_3 - x_i^k\right)\right) \tag{9}$$

The i^{th} particle's velocity in the $(k+1)^{th}$ iteration rate is denoted by v_i^{k+1}, the i^{th} particle's velocity in the t^{th} iteration is denoted by v_i^k, and the i^{th} particle's position in the t^{th} iteration is indicated by x_i^k [19]. Furthermore, the inertia weight, w, is typically selected from the interval [0,1.2]. Typically, the learning coefficients c_1, c_2 and c_3 are found in the interval [0,2]. The values r_1, r_2 and r_3 are chosen at random from a uniform distribution inside the interval [0,1].

The following is an update of the top three solutions (particle positions).

$$x_i^{k+1} = x_d^{t+1} + v_i^{k+1} \tag{10}$$

The position of the i^{th} particle in $(t+1)^{th}$ iteration is given by x_i^{t+1}.

Ultimately, a fitness function that reduces the number of features and improves classification accuracy is used to solve the optimization problem [10]. Equation 11 uses $\alpha = [0,1]$, $\beta = 1 - \alpha$, and $\rho_R(D)$ to represent the classifier error rate.

$$fitness = \alpha\rho_R(D) + \beta\frac{|SF|}{|WF|} \tag{11}$$

In addition, $|SF|$ represents the chosen subset of features, while $|WF|$ represents all the dataset's features.

2.2. Machine learning classifiers

After feature selection, machine learning algorithms were used to classify the data in the reduced number of features. These are k-nearest neighbor (KNN) [20], support vector machines (SVM) [21,22], Naive Bayes (NB) [23], artificial neural networks (ANN) [22,24] and decision trees (DT) [23,25].

2.2.1. K-nearest neighbors (KNN)

KNN is one of the most widely used machine learning methods in classification. It gives a class label according to the majority by looking at which class k neighbors of the point to be classified are in. The distance calculation and the number of neighbors (k) are important. The distance of the point to be predicted with other data (points) is calculated with distance formulas such as Euclidean, Manhattan, and Minkowski. Although there are studies on determining the number of neighbors, it is generally determined by taking the square root of the amount of data. On the other hand, when the number of data is large and the data size is high, the computation time of this algorithm increases. This situation constitutes its disadvantage [20].

2.2.2. Support vector machines (SVM)

Support vector machines are supervised learning models in machine learning that examine data for regression and classification along with related learning techniques [21]. It is a method that separates datasets planarly by dividing them at maximum distance. In data sets larger than two dimensions, it divides the data sets as hyperplane. If a data point is expressed as a d-dimensional vector, the separation process with a d-1 dimensional hyperplane is called SVM method. The area between the plane or line that best separate two classes is called margin. If some data are in the margin region, it is referred to as soft margin, and if none are in the margin region, it is referred to as hard margin. Naturally, a hard margin will be the desired situation in theory and practice.

Numerous hyperplanes are available for data classification. The hyperplane that shows the greatest margin or separation between the two classes is a fair candidate to be the best one. In order to optimize the distance to the closest data point on each side, the hyperplane is selected [22].

2.2.3. Naive bayes (NB)

Naive Bayes is a technique for creating strong and simple-to-train classifiers that calculate the probability of an event given a collection of conditions by applying Bayes' theorem. It is the derivation of a function of the classification with the help of conditional probabilities. Naive Bayes classifier is a multipurpose classifier and have applications in many different fields. In general, their performance is higher in all cases where the probability of a class is determined by the probabilities of some causal factors. When the likelihood of a class is based on the probabilities of certain causative elements, they often perform better in all circumstances. In other words, it gives better results when the initial probabilities needed in Bayes' theorem can be objectively determined without being subjective, that is, when they can be obtained from the data [23].

2.2.4. Artificial neural network (ANN)

Because not everything can be roughly predicted by a linear or logistic regression, neural networks were developed. The dependent variable is displayed in the output layer, and the independent variables are displayed in the input layer. To predict the dependent variable, one uses the independent variables. A neural network is an algorithm for supervised learning that makes use of a combination of hyper-parameters to approximate the intricate link between input and output. There are variables in an artificial neural network, like the number of hidden layers, hidden units, activation function, and learning rate [22,24].

The number of nodes in the output layer of a classification problem equals the number of classes in the dependent variable. The input variables are converted into a higher-order function using the hidden layer. Transforming the input signal into an output signal is the activation function's goal. They are necessary for neural networks to represent intricate nonlinear processes that more basic models might overlook [24].

2.2.5. Decision trees (DT)

A decision tree is an algorithm that uses a tree-like model of decisions and their possible outcomes, including chance event outcomes, resource costs and utility. Decision trees are one of the most preferred algorithms for classification problems. It was developed to subdivide a set containing a large amount of data into branches using methods such as information gain, gini index, and gain rate. As the name suggests, it consists of roots, branches, and leaves. The leaves represent the values of the class in the problem [23, 25].

3. Results

In this section, firstly, the benchmark data sets to be used in the study are mentioned. Then, feature selection methods are applied to these data. The parameters of the BHGWOPSO algorithm are determined as follows: the number of search agents is 10, the maximum number of iterations is 100, the number of wolves is 10, $c_1 = c_2 = c_3 = 0.5$ and $w = 0.5 + rand()/2$. With the help of the selected features, the classification process was performed by machine learning methods. The success of each method is measured by accuracy metric and compared with each other. Additionally, to prevent overfitting, 5-fold cross-validation is carried out. On a PC with an Intel® Core (TM) i7-4740 CPU and 16 GB of RAM, the classifications are made. Simulations are also carried out using the Matlab® R24b program.

3.1. Benchmark Datasets

Five different benchmark datasets are determined from the UCI Machine Learning Repository [26]. These are Breast Cancer, Wine, Student Success, Glass, and Connectionist Bench datasets. They are listed with their properties in Table 1.

Table 1. Description of the benchmark datasets

Data Set	Instances	Attributes	Classes
Breast Cancer	569	30	2
Wine	178	13	3
Student Success	4424	36	3
Glass	214	9	6
Connectionist Bench	208	60	2

Breast Cancer dataset is called as diagnostic Wisconsin breast cancer database. Its features are calculated from a digital picture of a breast mass that was aspirated with a fine needle. The target variable includes two classes: benign and malignant.

Wine dataset comprises the outcomes of a chemical study conducted on wines sourced from three distinct cultivars and grown in the same region of Italy. It's employed to ascertain the provenance of wines. The target variables consist of three categories.
Student Success dataset is called as predict students' dropout and academic success. Academic, demographic, and socioeconomic characteristics that were known at the time of student enrolment are included in the dataset. The output class consists of three categories: dropout, enrolled and graduate.

Glass dataset has six types of glass which are defined in terms of their oxide content such as Na, Fe, K, etc. This dataset is conducted for criminological investigation. The target attribute has six classes for identification of glass.

Connectionist Bench dataset is used to distinguish sonar signals bouncing off a metal cylinder from those bouncing off cylindrical rock. It is acquired by reflecting sonar waves off a metal cylinder at different angles and in different circumstances. It has a binary classification: rock or mine.

3.2. Classification performance metrics

This section presents classification results for five different benchmark datasets. To compare the performance of machine learning classification methods against feature extraction methods, accuracy performance measure given in equation (12) is used.

$$\text{Accuracy} = (TP + TN)/\,(TP + FP + FN + TN) \tag{12}$$

Here, the number of samples that are actually positive but predicted as positive is called true positive (TP), the number of samples that are actually negative but predicted as positive is called false positive (FP), the number of samples that are actually positive but predicted as negative is called false negative (FN), and the number of samples that are actually negative but predicted as negative is called true negative (TN).

First, the accuracy results for the Breast Cancer dataset are given in Table 2. Bold values indicate the classifier with the highest accuracy according to the relevant feature selection method in Table 2.

Table 2. Comparison of classification accuracy for Breast Cancer dataset

Learning Classifiers	**Overall**	**Feature Selection Methods**				
		PCA	**Filter**			**Wrapper**
			MRMR	**Chi2**	**ReliefF**	**BHGWOPSO**
KNN	97.4	90.7	96	93.5	97.2	96.3
SVM	**97**	90.3	**97**	94.7	**97.4**	**97.2**
NB	93.3	**91**	94.4	92.6	92.6	94.4
ANN	96.0	90.3	94.9	**95.6**	96.5	96
DT	91.6	87.5	91.6	93.5	91.6	93.1

According to Table 2, SVM classifier gives the highest accuracy value except for PCA and Chi2. As a result of feature selection with Filter and Wrapper methods, the percentage of success in all variables is achieved or exceeded with fewer variables. Among the filter methods, the ANN classifier provides the highest accuracy value for Chi2 with 95.6%. According to the PCA method, the highest accuracy value belongs to the NB classifier with 91%. Also, the best result was obtained with the SVM classifier with 97.2% for BHGWOPSO.

Figure 2 gives the graphical illustration of learning classifiers performance results for Breast Cancer dataset.

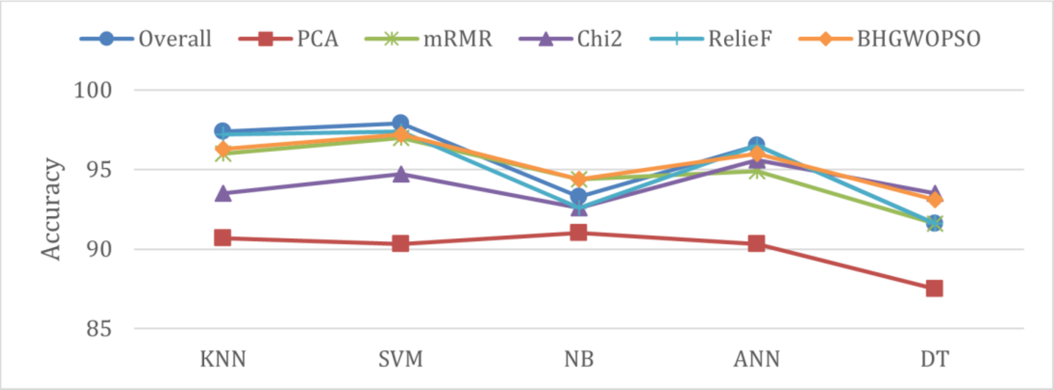

Figure 2. Accuracy changes of classifiers according to feature selection methods on the Breast Cancer dataset

According to Figure 2, the DT method generally has lower accuracy than the other classifiers. On the contrary, the SVM method seems to have higher accuracy than other classification methods. In the feature selection part, RelieF and BHGWOPSO seem to have higher accuracy values in general.

The number of features determined by the feature selection methods for the breast cancer dataset is given in Figure 3.

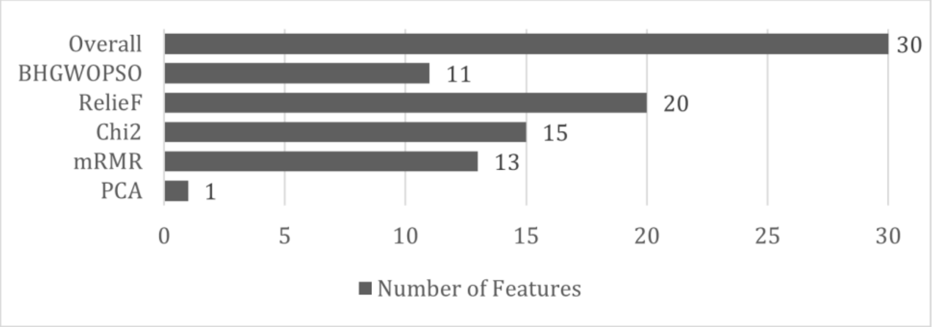

Figure 3. Number of features according to feature selection methods for Breast Cancer dataset

There are 30 features in total for the Breast Cancer dataset. Among the feature selection methods, PCA has low accuracy values despite selecting one feature. Although the other methods extracted almost the same number of features, BHGWOPSO selected the least number of features with 11.

The accuracy results for the Wine dataset are given in Table 3. Bold values indicate the classifier with the highest accuracy according to the relevant feature selection method in Table 3.

Table 3. Comparison of classification accuracy for Wine dataset

Learning Classifiers	**Overall**	**Feature Selection Methods**				
		PCA	**Filter**			**Wrapper**
			MRMR	**Chi2**	**ReliefF**	**BHGWOPSO**
KNN	96.6	65.5	95.5	96	96	**93.8**
SVM	**98.3**	69.5	97.2	96	**98.9**	92.1
NB	97.7	**70.1**	96.6	95.5	**98.9**	92.7
ANN	97.7	67.5	**98.3**	**97.2**	98.3	90.4
DT	87	68.4	87.6	87.6	89.3	**93.8**

According to Table 3, SVM and NB classifiers achieves the highest accuracy of 98.9% with the ReliefF method. The lowest classification percentage belongs to the KNN method with feature selection by the PCA method. Among the filter methods, the ReliefF and MRMR methods gave high accuracy results. In the BHGWOPSO method, the best result is obtained with 93.8% with the KNN and DT classifiers.

Figure 4 gives the graphical illustration of learning classifiers performance results for Wine dataset.

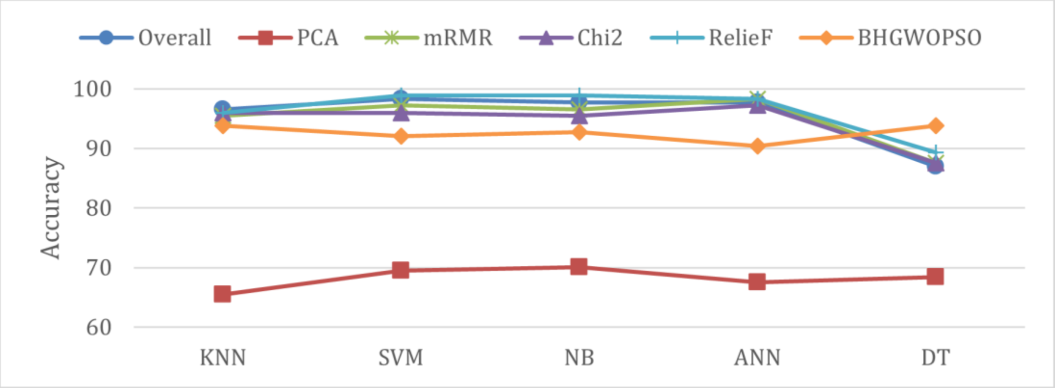

Figure 4. Accuracy changes of classifiers according to feature selection methods on the Wine dataset

According to Figure 4, the highest accuracy value was obtained with the ReliefF method, except for the DT method. In the DT method, the highest accuracy was achieved with BHGWOPSO. In general, SVM, NB and ANN classifiers achieved higher accuracy values. For this dataset, it can be said that the BHGWOPSO feature selection method has a lower performance after PCA, except for the DT classifier.

The number of features determined by the feature selection methods for the Wine dataset is given in Figure 5.

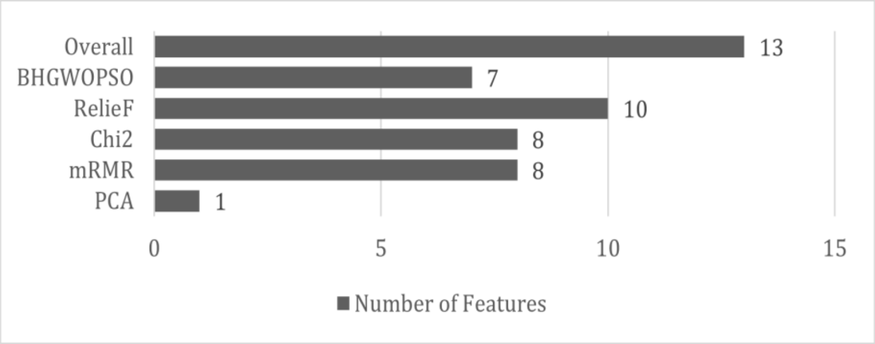

Figure 5. Number of features according to feature selection methods for Wine dataset

There are 13 features in total for the Wine dataset. It is seen from the previous results that although PCA selects one variable in the feature selection methods, it has low accuracy values. Although the other methods extract almost the same number of features, BHGWOPSO selects the least number of features with 7.

The accuracy results for the Student Success dataset are given in Table 4. Bold values indicate the classifier with the highest accuracy according to the relevant feature selection method in Table 4.

Table 4. Comparison of classification accuracy for Student Success dataset

Learning Classifiers	**Overall**	**Feature Selection Methods**				
		PCA	**Filter**			**Wrapper**
			MRMR	**Chi2**	**RelieфF**	**BHGWOPSO**
KNN	69.8	50.4	72.1	73.2	70.8	73.4
SVM	**76.5**	37.1	69.1	**76.5**	**75.9**	76.3
NB	65.8	49.8	67	70	71	69.2
ANN	74.8	53.4	**74.8**	76.2	75.1	**76.9**
DT	74.1	**54.1**	74.3	74.7	73.8	75.3

According to Table 4, the ANN classifier has the highest accuracy of 76.9%. This high accuracy value was achieved with BHGWOPSO. The filter methods worked with almost the same percentage of accuracy. The lowest accuracy value (37.1%) was obtained with the SVM classifier with PCA method.

Figure 6 gives the graphical illustration of learning classifiers performance results for Student Success dataset.

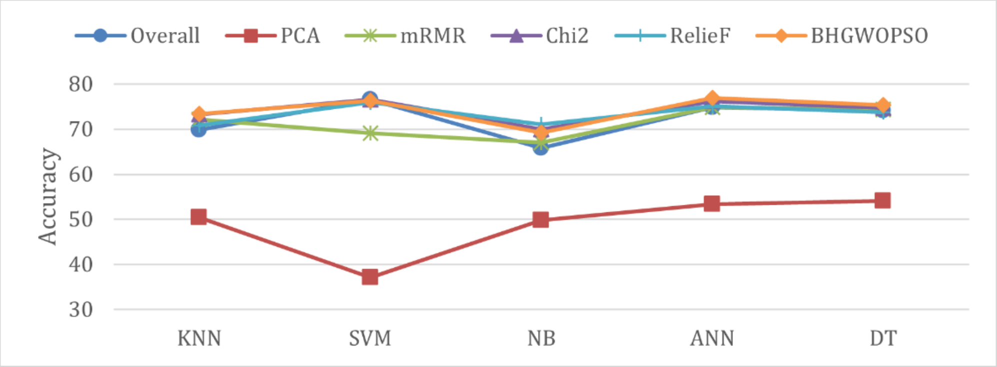

Figure 6. Accuracy changes of classifiers according to feature selection methods on the Student Success dataset

According to Figure 6, the accuracy values of the classifiers show a close variation except for the PCA method. The lowest success belongs to the PCA method. When PCA is used, the DT method has the highest accuracy value compared to the other classifiers. On the other hand, the accuracy values obtained with the ANN classifier are higher than the feature selection methods. In the feature selection part, it is seen that the BHGGWOPSO method has higher accuracy values in general.

The number of features determined by the feature selection methods for the Student Success dataset is given in Figure 7.

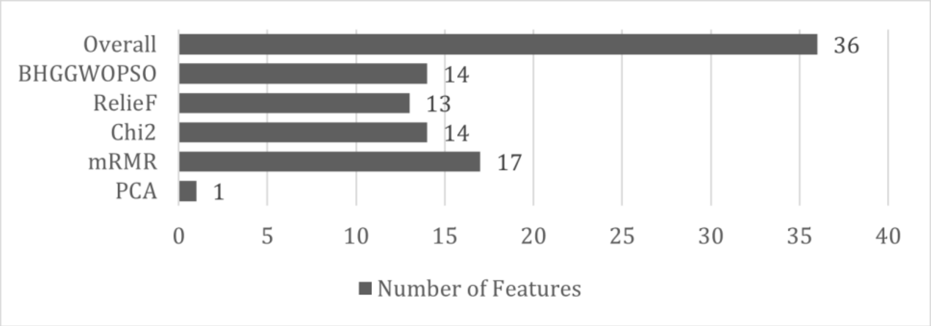

Figure 7. Number of features according to feature selection methods for Student Success dataset

There are 36 features in total for the Student Success dataset. It is seen from the previous results that although PCA selects one variable in the feature selection methods, it has low accuracy values. Although the other methods extract almost the same number of features, RelieF selects the least number of features with 13.

The accuracy results for the Glass dataset are given in Table 5. Bold values indicate the classifier with the highest accuracy according to the relevant feature selection method in Table 5.

Table 5. Comparison of classification accuracy for Glass dataset

Learning Classifiers	**Overall**	**Feature Selection Methods**				
		PCA	**Filter**			**Wrapper**
			MRMR	**Chi2**	**ReliefF**	**BHGWOPSO**
KNN	63.6	64	**67.8**	**75.7**	**72.9**	66.4
SVM	60.7	60.7	59.3	61.2	62.9	62.6
NB	63.6	50.9	67.3	62.1	62.6	**66.8**
ANN	**65.9**	**65.9**	65.4	65.9	66.4	64
DT	62.1	62.1	67.3	66.4	64.5	63.1

According to Table 5, the highest accuracy value of 75.7% with the KNN classifier was obtained with the Chi2 method. The lowest classification percentage belongs to the NB method with feature selection by the PCA method. Filter methods have higher accuracy values than other methods. It can even be said that when it works with the KNN classifier, it works better than other classifiers. The BHGWOPSO method has a higher accuracy value in the NB method compared to other classifiers.

Figure 8 gives the graphical illustration of learning classifiers performance results for Glass dataset.

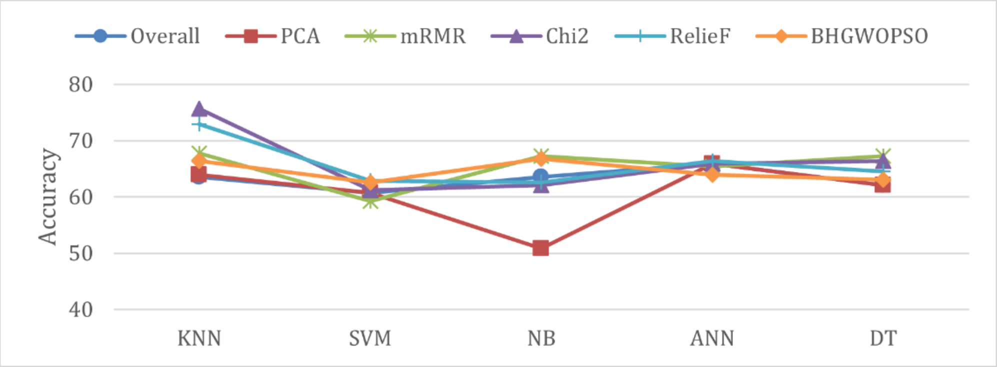

Figure 8. Accuracy changes of classifiers according to feature selection methods on the Glass dataset

According to Figure 8, the highest accuracy value is obtained with the KNN classifier with Chi2 method. The lowest success belongs to the NB classifier with the PCA method. The variability between the feature selection methods is higher for the KNN method and lower for the ANN classifier. The BHGGWOPSO method works better for SVM and NB classifiers.

The number of features determined by the feature selection methods for the Glass dataset is given in Figure 9.

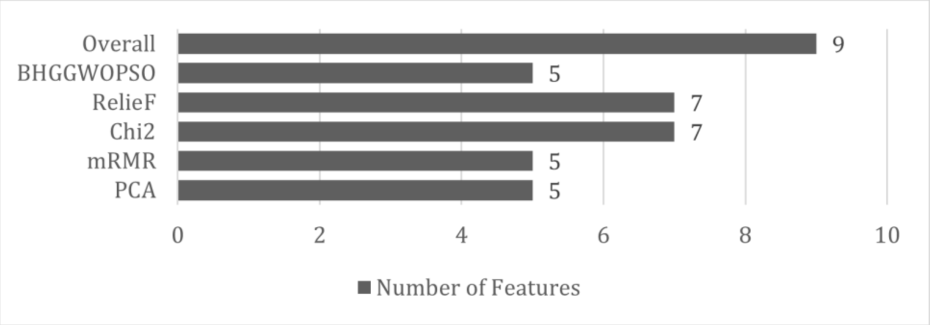

Figure 9. Number of features according to feature selection methods for Glass dataset

There are 9 features in total for the Glass dataset. In the feature selection methods, PCA, mRMR and BHGWOPSO have the least number of features with 5 variables.

Lastly, the accuracy results for the Connectionist Bench dataset are given in Table 6. Bold values indicate the classifier with the highest accuracy according to the relevant feature selection method in Table 6.

Table 6. Comparison of classification accuracy for Connectionist Bench dataset

Learning Classifiers	**Overall**	**Feature Selection Methods**				
		PCA	**Filter**			**Wrapper**
			MRMR	**Chi2**	**ReliefF**	**BHGWOPSO**
KNN	81.2	76	**74.5**	**81.7**	**79.3**	79.3
SVM	77.9	76	73.6	76.9	76.4	74.5
NB	70.2	76	70.2	66.3	64.9	67.8
ANN	**82.7**	**79.8**	68.3	75.5	75	**81.7**
DT	73.6	72.6	70.2	76.9	79.3	76.4

According to Table 6, the highest accuracy of 82.7% was obtained with the ANN classifier when all variables were used. However, when feature selection is applied instead of using all variables, the highest accuracy is obtained with BHGWOPSO with Chi2 with 81.7%. The classifiers for these values are KNN and ANN, respectively. The lowest classification percentage (64.9%) belongs to the NB method with RelieF feature selection. Filter methods work best with the KNN classifier, while BHGWOPSO achieved the highest success with ANN.

Figure 10 gives the graphical illustration of learning classifiers performance results for Connectionist Bench dataset.

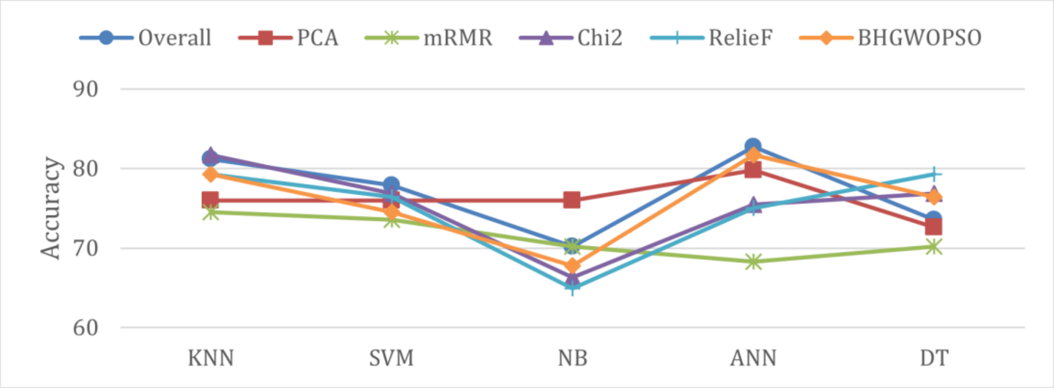

Figure 10. Accuracy changes of classifiers according to feature selection methods on the Connectionist Bench dataset

According to Figure 10, the highest accuracy value is obtained with the ANN classifier using all variables. The lowest accuracy is obtained by the NB classifier with the RelieF method. The difference in accuracy values between the feature selection methods is higher in the ANN method, while the variability is less in the SVM classifier. It can be said that the BHGGWOPSO method works better with the SVM and NB classifiers. It can be said that the success of the PCA method has increased for this dataset.

The number of features determined by the feature selection methods for the Connectionist Bench dataset is given in Figure 11.

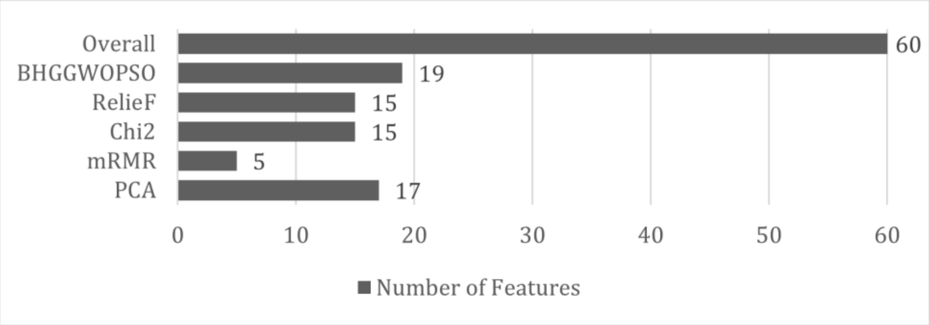

Figure 11. Number of features according to feature selection methods for Connectionist Bench dataset

There are 60 features in total for Connectionist Bench dataset. Among the feature selection methods, mRMR has the least number of features with 5 variables. The other methods have a similar number of features.

4. Discussion and conclusions

In this study, a detailed comparison of BGWOPSO method, which is one of the Wrapper methods, with other feature selection methods is made. In five benchmark datasets, the feature selection methods are applied with machine learning classifiers and compared with their accuracy values. When the BHGWOPSO feature selection method is applied, the SVM classifier for Breast Cancer dataset, DT and KNN for the Wine dataset, ANN for the Student Success dataset, NB for the Glass dataset, and ANN for the Connectionist Bench dataset have the highest accuracy. This is quite normal as no single method is considered to work best for all datasets.

When the simulation results are analyzed, it can be said that the performance of different feature selection methods and different classifiers for each data set is high. In this case, if a machine learning classifier is to be used for any data set, it would be useful to apply different feature selection methods and compare the results. Especially in the last dataset, although the highest accuracy is obtained when all features are used, it is seen from the accuracy values that almost the same success can be achieved when fewer features are preferred due to cost, time, etc. In cases where feature selection is applied, higher classification accuracy can be achieved with fewer variables when redundant and irrelevant variables are not used.

Since it is desired to achieve maximum accuracy with the minimum number of features, the minimum number of features is chosen to give maximum accuracy in all feature selection methods. However, there may be an increase in accuracy values for some learning classifiers as the number of features increases. In addition, although the same number of variables is selected, different accuracy values between classifiers are due to the selection of different variables.

Future studies can investigate how hybrid GWO-PSO algorithms work on different datasets and application domains. There is great potential in combining algorithms with different meta-heuristics to make them more efficient and integrating them with advanced techniques such as deep learning. The success of these algorithms in different sectors such as time series data, health, finance and bioinformatics can make significant contributions in terms of data diversity and model adaptation. Enhancing hybrid algorithms with ensemble methods can improve the performance of classifiers and provide more reliable results. In this context, diversification of model evaluation metrics and comparisons with multiple classifiers can help us better understand how algorithms perform on different data types. Finally, the development of new methodologies to improve the efficiency and accuracy of hybrid GWO-PSO algorithms may offer innovative solutions in the fields of machine learning and optimization.

References

[1] Büyükkeçeci, M., Okur, M. C., A Comprehensive Review of Feature Selection and Feature Selection Stability in Machine Learning. **Gazi University Journal of Science**, **36**, 4, (2022).

[2] Cherrington, M., Thabtah, F., Lu, J., Xu, Q., Feature selection: filter methods performance challenges, **Proceedings, International Conference on Computer and Information Sciences (ICCIS)**, 1-4. (2019).

[3] Moradi, P., Gholampour, M., A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy, **Applied Soft Computing**, **43**, 4, 117-130, (2016).

[4] Zarshenas, A., Suzuki, K., Binary coordinate ascent: An efficient optimization technique for feature subset selection for machine learning, **Knowledge-Based Systems, 110,** 191-201, (2016).

[5] Talbi, E.-G., Jourdan, L., Garcia-Nieto, J., Alba, E., Comparison of population based metaheuristics for feature selection: Application to microarray data classification, **Proceedings, 2008 IEEE/ACS International Conference on Computer Systems and Applications**, 45-52, (2008).

[6] Xue, B., Zhang, M., Browne, W., Yao, X., A survey on evolutionary computation approaches to feature selection, **IEEE Transactions on evolutionary computation, 20**, 4, 606-626, (2015).

[7] Nekkaa, M., Boughaci, D, Hybrid harmony search combined with stochastic local search for feature selection, **Neural Processing Letters, 44**, 199-220, (2016).

[8] Remeseiro, B., Bolon-Canedo, V., A review of feature selection methods in medical applications, **Computers in biology and medicine, 112**, 103375, (2019).

[9] Ghosh, M., Guha, R., Alam, I., Lohariwal, P., Jalan, D., Sarkar, R., Binary genetic swarm optimization: A combination of GA and PSO for feature selection, **Journal of Intelligent Systems, 29**, 1, 1598-1610, (2019).

[10] Al-Tashi, Q., Kadir, S., Rais, H., Mirjalili, S., Alhussian, H., Binary optimization using hybrid grey wolf optimization for feature selection. **Ieee Access, 7**, 39496-39508, (2019).

[11] El-Kenawy, E.-S., Eid, M., Hybrid gray wolf and particle swarm optimization for feature selection, **International Journal of Innovative Computing, Information and Control, 16**, 3, 831-844, (2020).

[12] Allam, M., Malaiyappan, N., Wrapper based feature selection using integrative teaching learning based optimization algorithm, **International Arab Journal of Information Technology, 17**, 6, 885-894, (2020).

[13] Sameer, F., Comparison study on the performance of the multi classifiers with hybrid optimal features selection method for medical data diagnosis, **Multimedia Tools and Applications, 81,** 13, 18073-18090, (2022).

[14] Alpaydin, E., **Introduction to machine learning**. MIT press, (2020).

[15] Abdi, H., & Williams, L., Principal component analysis, **Wiley interdisciplinary reviews: computational statistics, 2**, 4, 433-459, (2010).

[16] Sánchez-Maroño, N., Alonso-Betanzos, A., Tombilla-Sanromán, M., Filter methods for feature selection-a comparative study, **Proceeding, International Conference on Intelligent Data Engineering and Automated Learning**, **2**, 178-187, (2007).

[17] Liu, H., Motoda, H., Setiono, R., Zhao, Z., Feature selection: An ever evolving frontier in data mining, **Proceeding, Feature selection in data mining**, 4-13, (2010).

[18] Liu, H., Zhao, Z., **Manipulating data and dimension reduction methods: Feature selection**, Springer New York, (2012).

[19] Farshi, T. R., Drake, J. H., Özcan, E., A multimodal particle swarm optimization-based approach for image segmentation, **Expert Systems with Applications, 149**, 113233, (2020).

[20] Raschka, S., Liu, Y., Mirjalili, V., Dzhulgakov, D., **Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python**, Packt Publishing Ltd, (2022).

[21] Sun, J., Rahman, M., Wong, Y., Hong, G., Multiclassification of tool wear with support vector machine by manufacturing loss consideration, **International Journal of Machine Tools and Manufacture, 44**, 11, 1179-1187, (2004).

[22] Hastie, T., Tibshirani, R., Friedman, J., Friedman, J., **The elements of statistical learning: data mining, inference, and prediction**, Springer, (2009).

[23] Bonaccorso, G., **Machine learning algorithms**, Packt Publishing Ltd, (2017).

[24] Ayyadevara, V., **Pro machine learning algorithms**, Springer, (2018).

[25] Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., Wu, Xindong, Top 10 algorithms in data mining, **Knowledge and information systems, 14**, 1-37, (2008).

[26] M. Kelly, R. Longjohn, K. Nottingham, The UCI Machine Learning Repository, (2024). https://archive.ics.uci.edu, (26.01.2024).