Received November 26, 2021, accepted December 19, 2021, date of publication January 5, 2022,
date of current version January 13, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3140209

New Feature Selection Algorithm Based on Feature Stability and Correlation

LUAI AL-SHALABI
Faculty of Computer Studies, Arab Open University, Al-Ardia 92400, Kuwait
e-mail: lshalabi@aou.edu.kw

ABSTRACT The analysis of a large amount of data with high dimensionality of rows and columns increases the load of machine learning algorithms. Such data are likely to have noise and consequently, obstruct the performance of machine learning algorithms. Feature selection (FS) is one of the most essential machine learning techniques that can solve the above-mentioned problem. It tries to identify and eliminate irrelevant information as much as possible and only maintain a minimum subset of appropriate features. It plays an important role in improving the accuracy of machine-learning algorithms. It also reduces computational complexity, run time, storage, and cost. In this paper, a new feature selection algorithm based on feature stability and correlation is proposed to select the effective minimum subset of appropriate features. The efficiency of the proposed algorithm was evaluated by comparing it with other state-of-the-art dimensionality reduction (DR) algorithms using benchmark datasets. The evaluation criteria included the size of the minimum subset, the classification accuracy, the F-measure, and the area under curve (AUC). The results showed that the proposed algorithm is the pioneer in reducing a given dataset with high predictive accuracy.

INDEX TERMS Classification algorithms, correlation, feature selection, stability.

I. INTRODUCTION

Dimensionality reduction methods are becoming required. Of the methods available, feature extraction and feature selection are of big interest. Feature extraction methods such as PCA are trying to get useful features from existing data to minimize the dimensionality of a dataset. The methods are also used to prevent irrelevancy and redundancy. Feature selection is a much-demanded process for machine learning, data analysis, data science, data mining, and big data. This process improves the machine learning goals and minimizes the effort and cost of working with the data. It is simply defined as the process of minimizing the dimensionality of a dataset by reducing the number of attributes without affecting the accuracy of recognition. The reduced dataset could have the same accuracy, improved accuracy, or slightly less accuracy. Removing some attributes through the process of feature selection could improve the accuracy by ignoring irrelevant attributes, redundancy attributes, high rate of data variation, and/or low rate of data variation. Some attributes may cost the user or the customer much money and removing them will return benefits to the user or customer. This is a trade-off between the accuracy and reduction of a dataset. For sensitive datasets such as medical datasets, the priority is accuracy. We may sacrifice dimensionality reduction for accuracy. The dataset could be reduced to the point where further reduction negatively affected the accuracy. For a non-sensitive dataset, it is reasonable to consider both the accuracy and dimensionality reduction. The dataset could be reduced, and this reduction was acceptable even if its accuracy was slightly less than that of the original dataset. In this case, we may sacrifice a small difference in accuracy for dimensionality reduction (number of attributes). In the literature, there are many feature selection methods; none of them is suitable and performs best for every dataset. One feature selection method performed best for a specific dataset, but not for the others. Thus, no one can say that one method is good and the other is bad. As many feature selection methods exist, a wise decision should be made to choose a suitable one with respect to the accuracy, the number of attributes given, and/or run time. This is not an easy decision.

In the past few decades, many feature selection methods have been proposed. Guyon and Elisseeff classified feature selection methods into three classes: filter, wrapper, and embedded methods [1]. They mentioned that in the filter method, each attribute (feature) will have a calculated score, and all attributes with scores that exceed a determinant threshold value are selected. The selected attributes were the most

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Callico.

informative. Das *et al.* [2] stated that feature selection techniques can be divided into filter, wrapper, and embedded methods. Filter methods mostly focus on the properties of data examples without taking into consideration the underlying learning scheme [3]. On the other hand, wrapper methods utilize a classifier or a learning model to evaluate the efficiency of several reducts and implement a search method to find the best one. In contrast to wrapper methods, embedded methods consider feature selection in the training process to reduce the time complexity for re-classifying different reducts.

Wrapper methods are more efficient than filter methods as they consider both feature subset and classifier. However, wrapper methods are more complex because a classifier is required to re-learn each reduct [4].

Lazar *et al.* [5] explained several existing filter methods in detail. Filter methods are considered simple and fast and can deal with thousands of attributes. In the wrapper method, a new subset of features is created from the original dataset. The subset was trained, and the accuracy was generated using a classifier. Based on the accuracy, a decision is made to add or remove the attribute from the subset. This method is considered very expensive because the training process will be continually run until the decision is taken to stop the process. Kohavi and John [6] gave extensive explanations about wrapper methods. Embedded methods use hybrid learning methods to obtain an appropriately reduced dataset. They combine the abilities of both wrapper and filter methods.

In this study, a hybrid ScC model was proposed for feature selection as an example of filter methods. Firstly, Stability and correlation are used to discard unwanted examples by removing examples that have high stability and low correlation value as it will be explained later in this article. Then, the coefficient is adopted on the partial reduct generated from the previous step to search and select a best reduct. The proposed ScC model focusses on selecting the most significant features while maximizing the classification accuracy. Nine benchmark datasets are employed to evaluate the efficiency of ScC. The results are measured statistically using different metrics: accuracy, AUC, and F-measure plus the reduction rate.

The main contribution of this study is to merge stability with correlation to carry out feature selection and classification.

In short, the main contributions of this article are two-folds:

- a hybrid ScC model that can reduce the dimension of a dataset to the minimal and maximize classification accuracy.
- a comprehensive statistical evaluation of ScC using benchmark datasets.

The remainder of this paper is organized as follows. Section II describes related works. Section III describes the preprocessing engine including dimensionality reduction methods used in this study. Section IV describes the proposed method. Section V describes the results and the evaluation process. This article concludes with a conclusion.

II. LITERATURE REVIEW

Many different articles have been published, and they have focused on the comparison between different feature selection methods. One of the best ways to assess the performance of feature selection methods is classification. Many published articles have conducted this assessment method, which performs well. A new correlation feature selection (CFS) method was proposed by Hall [7]. The author showed that this method can be applied to regression and classification problems. Phyu and Oo [8] discussed the importance of feature selection methods. They concluded that the selected subset of features was independent and sufficiently relevant for the learning process. Liu *et al.* Ghosh *et al.* [9] built a filter method for managing an evolutionary method. Their method was compared with other evolutionary methods directed by single-filter methods. They tested their method and compared it with other methods regarding the classification accuracy. A new filter method was developed by Hoque *et al.* [10], which summed the scores of numerous filter methods and then compared them to the single filter methods. They used small and large datasets with respect to the number of attributes. Comparisons were made between the methods regarding the classification accuracy metric. A comparison was made by Wah *et al.* [11] to compare wrapper and filter feature selection methods on large datasets with respect to the correctness of the selected features by comparing the classification accuracy of real datasets. Liu [12] and Peng *et al.* [13] compared different filter methods by measuring their classification accuracy regarding the number of features generated by each method.

Similarly, new filter methods have been proposed by many researchers. Pourpanah *et al.* [14] proposed a FAM-BSO feature selection model for data classification by combining the Fuzzy ARTMAP model, which is working as an incremental learning neural network, with BSO, which acts as a feature selection method. Fleuret [15] and Ke et al [16] were, in their works, conducted comparisons with respect to the classification accuracy and run time metrics separately. Zhu *et al.* [17] and Mohtashami and Eftekhari [18] presented new wrapper method. A comparison was conducted with respect to the classification accuracy metric, which was applied to several datasets. Another comparison study between the wrapper and filter methods was conducted by Xue *et al.* [19]. This study used classification accuracy and run time as metrics for comparison. Bolón-Canedo *et al.* [20] studied three types of feature selection methods (filter, wrapper, and embedded) by comparing the accuracies of different classifiers. Based on two gene expression datasets, Cherrington *et al.* [21] investigated and analyzed different filter methods based on ranking procedures. They concentrated on how threshold determination can affect results of different filter methods based on ranked scores. A comparison study between the three methods of feature selection was conducted by Chaudhary *et al.* [22]. They compared the performance of the gain ratio, correlation, and information gain methods with the naïve Bayes classifier.

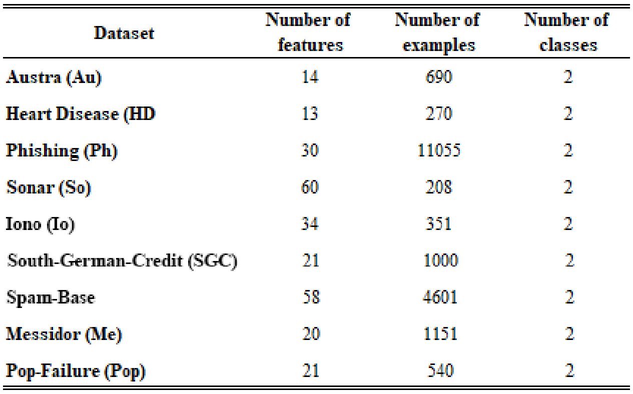
To generate an optimal feature subset, Haq *et al.* [23] combined features selected by multiple feature selection techniques. Darshan and Jaidhar [24] analyzed different filter methods applied to malware detection datasets. They compared the results based on calculated classification accuracy. Comprehensive studies were conducted using text classification datasets to compare the filter and wrapper methods, as in [25]. Bolón-Canedo *et al.* [26] and Inza *et al.* [27] studied the classification accuracy of different filter methods that were applied to microarray datasets. Most existing feature selection algorithms are not compliant with classifying big data and high-dimensional datasets. Du *et al.* [28] proposed a filter based unsupervised feature selection algorithm by extracting the global level manifold structure using LLE on all features and the feature level manifold structure using LLE on each single feature. Then they computed the feature-wise non-negative local linear reconstruction weight to capture the feature relationship. Bommert *et al.* [29] investigated 22 filter methods and 16 high-dimensional datasets with respect to accuracy and runtime. To do this, they used the R machine learning package, which is a convenient tool to conduct feature selection using filter methods. They concluded that there is no filter method that continuously outperforms the other methods. Al-Shalabi [30] proposed a dimensionality reduction algorithm for noisy datasets by reducing the number of incomplete data using rough set theory. Uddin *et al.* [31] illustrated the importance of feature reduction and showed a cost-effective and improved classification performance of the feature extraction approaches over the performance using the entire original dataset. Sinayobye *et al.* [32] proposed a hybrid classification model which has correlation-based filter feature selection algorithm and machine learning as classifiers. Hu *et al.* [33] designed a sequential forward selection method based on a separability (SFSS) algorithm to select attributes. The authors showed that SFSS is fast and robust, with a higher classification accuracy, compression ratio, and extremely low computational time. In the work of Bashir *et al.* [34], a model order reduction of a stable doubly fed induction generator based variable-speed wind turbine model was performed with the aid of the proposed stability-preserving balanced realization algorithm based on discrete frequency weights and a limited frequency interval. In contrast to conventional reduction methods, the proposed method produces steady and precise outcomes. Methods based on information theory such as QPFS, MLQPFS, DQPFS, and CMIM are also important to be mentioned. QPFS is a feature weighting algorithm that has been used in many applications for feature selection. Soheili and Moghadam [35] proposed scalable algorithm called DQPFS which is based on the novel Apache Spark cluster computing model. The excecution of their algorithm show significant results for big data feature selection. Soheili and Moghadam [36] proposed a new algorithm called MLQPFS which selects subset of features with minimal redundancy among the selected features while having the maximized relevancy between the selected features and class labels. Ramírez-Gallego *et al.* [37] proposed a distributed implementation of a generic feature selection framework that includes a broad group of recognized information theory-based methods. Their results show that the proposed framework is capable of quickly dealing with ultrahigh-dimensional datasets as well as those with a big number of samples. Bermego *et al.* [38] proposed an adaptation of the well-known CMIM feature selection algorithm. The algorithm is efficient of approximating the conditional multivariate mutual information of each candidate attribute with respect to the entire set of features.

III. PREPROCESSING ENGINE

A. DATASETS

Nine datasets from various domains were used in the analysis. The number of records, features, and classes are listed in Table 1.

TABLE 1. The datasets.



Dataset	Number of features	Number of examples	Number of classes
Austra (Au)	14	690	2
Heart Disease (HD	13	270	2
Phishing (Ph)	30	11055	2
Sonar (So)	60	208	2
Iono (Io)	34	351	2
South-German-Credit (SGC)	21	1000	2
Spam-Base	58	4601	2
Messidor (Me)	20	1151	2
Pop-Failure (Pop)	21	540	2

B. DIMENTIONALITY REDUCTION METHODS

Feature extraction transforming primary features into a new set that a particular Machine Learning algorithm requires. Feature selection usually identifies the most important and significant features of a dataset of interest. Both improve the learning process efficiency and minimizes the execution time of the learning process. For the sake of contrast, three state-of-the-art dimensionality reduction methods were considered for comparison with the proposed method in terms of the size of the reduced dataset, accuracy, AUC, and F-measure. They are PCA, rough set (RS), and weight guided (WG), whose properties are briefly summarized next.

1) PCA

Mishra *et al.* [39] reported that PCA is an important algorithm that is used to extract important knowledge from a dataset. The PCA algorithm uses a vector space transform for feature reduction in large datasets. For the original dataset, a mathematical projection was used to produce a few important variables or what are called principal components. They also showed that the PCA algorithm represents an important relationship between the different dimensions of data and that it produces eigenvectors. Each eigenvector has and eigenvalue points and the one with the maximum points in the middle of the data. PCA generates the weights needed to produce a new feature that best explains the disparity in the given dataset. The new nominated feature with the defining weights is called the first principal component [40]. More details about the PCA algorithm can be found in [41]. Song *et al.* [42] proposed a method to apply PCA to feature selection. The method well addressed the feature selection issue. Their numerical analysis shows that PCA has the potential to perform feature selection and can select several important individuals from all the feature components.

2) ROUGH SET

As shown by Velusamy and Manavalan [43], rough set theory is used to develop knowledge discovery algorithms using the discernibility matrix and partition properties. The mentioned discernibility matrix is used for feature selection by identifying both the dispensable and indispensable features. Rough set theory calculates many reducts. It then calculates the core that represents the common significant features in all the reducts by using the intersection between them. Feature selection based on rough set theory removes the dispensable attributes from the original dataset (D) in a controlled manner so that the generated reduct (R) provides the same quality of classification (shown as γ) as the original dataset. Velayutham and Thangavel [44] defined a reduct as a subset of the conditional attribute set (C) such that $\gamma R(D) = \gamma C$ (D). All generated subsets are defined as follows:

$$R_{\text{all}} = \{X|X \subseteq C, \gamma_X(D) = \gamma C(D), \tag{1}$$

$$\gamma_{X\text{-}\{a\}}(D) \neq \gamma_X(D), \quad \forall a \in X \tag{2}$$

For many tasks, the best reduct is the one with the minimum number of attributes that must satisfy $R_{\min} \subseteq Rall$, where

$$R_{\min} = \{X|X \subseteq R_{all}, \forall Y \in R_{all}, |X| \leq |Y|\}, \tag{3}$$

3) WEIGHT GUIDED

The weight-guided feature selection algorithm uses attribute weights as an input to determine the order of features added to the set of features. Features with the highest weight have the priority to be added to the set of features. Once an addition to the set of features does not improve the performance or if all features are added, the algorithm is stopped [45].

C. THE CLASSIFICATION ALGORITHMS (CA)

Four classification algorithms were employed to determine the classification performance of the minimal datasets generated by the feature-selected methods explained earlier. The implementation of the classification algorithms considered in this study was provided by the Rapidminer tool using 10-fold cross-validation method for the learning process. The four classification algorithms are explained as follows:

1) Logistic regression (LR) is used to determine whether a feature supports an explicit outcome. It is a supervised machine learning algorithm that generally analyzes a dataset and responds to a type of yes/no question. Peng *et al.* [46] showed that logistic regression has many types, including binary, ordinal, binomial, and multinomial.
2) The fast large margin (FLM) algorithm is an important algorithm in machine learning and is based on a linear support vector. The algorithm is used with iceberg datasets with multiple records and features. The results obtained by this algorithm are like those obtained using either logistic regression or SVM [47].
3) The random forest (RF) algorithm proposed by Breiman [43] refers to a family of methods for creating a group of tree-based classifiers. It is used for regression,classification, and other problems. It creates a number of decision trees during the training process and chooses the common class of individual trees for classification or the mean prediction for regression [49], [50].
4) Gradient boosted trees (GBT) improve the accuracy of successively generated trees. It is a nonlinear regression algorithm that reduces the speed while increasing accuracy. Natekin and Knoll [51] stated that gradient-boosted trees use two layers; thus, they are called 'shallow learning.' They also demonstrated that they are appropriate for the structure.

IV. THE PROPOSED FEATURE SELECTION METHOD

In this section, the two main metrics that are considered as the skeleton of the proposed algorithm, namely stability (S) and correlation (r) will be discussed. The best values of stability and correlation r, which increase the performance accuracy, are shown. Finally, the proposed step-by-step algorithm is presented.

A. STABILITY

It measures how constant or stable the feature (the dataset column) is. If the values of the feature do not vary sufficiently and the feature is practically constant, with a high percentage of values being the same, then it is not a stable feature. The stability of the features in the reduct increases the confidence of the reduct if it does not exceed a specific threshold value. The stability with a specific threshold value represents the robustness of the feature. This indicates the power of the feature that represents the high relevance to the application and its dataset. The stability of the feature is also important to provide higher classification accuracy. Stability is calculated by dividing the number of tuples with the most frequent non-missing values by the total number of tuples with non-missing values. Stability, in this work, is used as a valuable metric to indicate the most appropriate features. To build a better classifier, a robust minimal set of features with high stability is desirable. Such a minimal set will help accelerate the classification process by reducing computational overwork. Khaire and Dhanalakshmiwe [52] presented a summary of feature selection algorithms and their instability. They also provide solutions for handling instability. Yang *et al.* [53] studied the instability of selected features and the difficulty of making a prediction. Kalousis *et al.* [54] examined the stability of feature selection algorithms in high-dimensional spaces, as well as numerous types of data. Perthame *et al.* [55] studied the stability of feature selection algorithms in high-dimensional correlated data. Dernoncourt *et al.* [56] investigated the stability of feature selection algorithms in high dimensions using small sample data.

B. CORRELATION r

The linear correlation coefficient, which is denoted by r is used to measure the strength of the linear relationship between the two variables as well as their direction. This linear coefficient is also called the Pearson correlation coefficient [57]. As an example, we may use the correlation method to determine the relationship between student-age and student-mark variables.

The correlation value could be any value between 1 and -1 inclusively. A correlation value of 1 represents the maximum positive relationship between the two variables, whereas a correlation value of -1 represents the maximum negative relationship between the two variables. The other values lie between -1 and 1. A correlation value of 0 indicates that there is no correlation between the two variables. In short, any correlation value satisfies $(-1 < r < +1)$. The following formula for computing the correlation r is used in this research, based on the standard deviation of the two variables:

$$r = (1/(n-1))^{*}((\sum\nolimits_{x}\sum\nolimits_{y}(x-\bar{x})(y-\bar{\boldsymbol{y}})/(S_xS_y)) \tag{4}$$

$$S_x = \surd((\sum(x-\bar{x})^2)/(n-1)) \tag{5}$$

where n is the number of pairs of data used, Σ is Sigma that represents the summation, $\bar{x}$ is the mean of all x-values, $\bar{\boldsymbol{y}}$ is the mean of all y-values, $\mathrm{S_x}$ is the standard deviation of variable x, and $\mathrm{S_y}$ is the standard deviation of variable y. In this study, the correlation is the other valuable metric for selecting the most relevant features.

C. THE PROPOSED ALGORITHM AND THE PREDEFINED VALUES

The proposed algorithm consists of two main stages. In the two stages, both metrics explained earlier make a difference. In stage 1, the combination between the stability and the correlation will be used; in stage 2, the correlation will only be considered, and it will be applied to the result of stage 1. Many experiments have been conducted to find the best value of stability, which increases the performance accuracy of the chosen set of features (reduct). The most probable value for S after the multiple tests was 35%. The best feature is when the stability value is less than or equal to 35%. In addition, the pairs of stability and correlation were combined to maximize performance accuracy. In the combination process, the most probable values for S and r after multiple tests were 35% and 10% respectively. The best feature is if it has a stability value greater than or equal to 35%, but the correlation value is greater than or equal to 10%. In stage 2, and for all features generated from stage 1, the most probable value for $r$after multiple tests was 5%. Another feature selection process was based on the value of the correlation generated in this stage (5%). The best feature is that it has a correlation greater than 5%.

This research introduces the ScC algorithm which is a novel feature selection algorithm in which the objectives are intended to increase the amount of reduction as well as the performance accuracy. To do so, ScC merged the stability and correlation aspects. To the best of my knowledge, no method has been proposed that aims to use stability and correlation together. Therefore, the proposed algorithm is novel.

Given a dataset $DS = (O, F, V, f)$, where O is a finite set of objects, F is a finite set of features, $V = \cup a \in F$, V_n is a domain of attribute a, and $f: O \times F \rightarrow V$ is a total function such that $f(x, a) \in V_n$ for every $a \in F$, $x \in O$. Let $\gamma(R)$ is an estimate of the performance accuracy of the reduct generated by ScC. In the dataset (DS), the minimal subset R, where $R \subseteq F$ such that $\gamma(R)$ is maximized is called the reduct of DS and is denoted by RED(DS).

If R is very small, $\gamma(R)$ may decrease. This is because the smaller the number of features (which could not be sufficiently representative) to approximate the underlying classification problem, the lower the capacity of the algorithm.

ScC is a new method that can be used to select more representative features from multidimensional features. It considers the stability and correlation of the features. It consists of the two stages mentioned earlier, where Stage 2 relies on the results of Stage 1. Stage 1 is represented by Algorithm 1, whereas Stage 2 is represented by Algorithm 2. Both algorithms are called by Algorithm 3, which represents the proposed feature selection algorithm, which also ranks the different models based on their accuracies to determine their best. To specify the algorithm in detail, *DS* was used to represent the dataset, *CA* was used to represent the classification attribute in the dataset, *SF1* was used to represent the selected features based on the stability value and correlation together (Stage 1), S was used to represent the stability of the feature, r was used to represent the correlation of the feature, and n was used to represent all non-missing values in each feature. *SF* is used to represent the selected features based on the correlation of the selected features from Stage 1 (Stage 2), *CAcc* was used to represent the classification accuracy, *CM* was used to represent the classification model, α was used to represent the most probable value for S, β was used to represent the most probable value for r in Stage 1, and λ is used to represent the most probable value for r in Stage 2.

After multiple tests, the following values were chosen: $\alpha = 0.35$, $\beta = 0.1$, and $\lambda = 0.05$. The scheme of the proposed model is shown in Figure 1 and it is described as follows.

Step 1. Start with the original dataset (*DS*) and empty datasets (*SF1* and *SF*).

Step 2. For each non-classification feature in the original dataset, the stability value was calculated with respect to all non-missing values.

$$S = mode(C_i)/n \tag{6}$$

Step 3. For each non-classification feature in the original dataset *DS*, the correlation *r* was calculated for each non-missing value with respect to the classification attribute corresponding value.

Step 4. For each non-classification feature in the original dataset *DS*, if the stability value is less than α or if the stability value is greater than or equal to α and the correlation is greater than or equal to β, then this feature is added to the *SF1* dataset.

Step 5. For each non-classification feature in the *SF1* dataset, the correlation was calculated for each non-missing value. If the correlation is greater than λ, then this feature is added to the *SF dataset.*

Step 6. *SF* is the dataset with the most suitable features for classification.

Step 7. To evaluate the final reduct, a loop of different classification methods was established and the method with the highest accuracy value would be chosen.

The proposed algorithms were built as shown below:

Algorithm 1 Stage1-Selection Based on Stability and Correlation

```
Input: Original Dataset DS, Classification Attribute CA
Output: Selected Features SF1
  $SF1 \leftarrow \emptyset$    //Selected features, initially empty.
  $\alpha = 0.35$   //The most probable value for S after multiple tests.
  $\beta = 0.1$   //The most probable value for r after multiple tests.
  //Loop until no more features can be selected.
  //Calculate stability S for each column $C_i$ with
  //respect to all non-missing values, n.
  for each column (feature) in DS ($C_i \in DS$)
      $S_{Ci} \leftarrow$ mode($C_i$)/n
      Calculate the correlation r for
      each value in column $C_i$ with respect to CA
      if $S_{Ci} < \alpha$ or ($S_{Ci} \geq \alpha$ and $r_{Ci} \geq \beta$) then
          $SF1 \leftarrow SF1 \cup C_i$
      end if
  end for
  return SF1
```

Algorithm 2 Stage2-Selection Based on Correlation ***r***

```
Input: Dataset SF1, Classification Attribute CA
Output: Selected Features SF
  $SF \leftarrow \emptyset$    //Selected features, initially empty.
  $\lambda = 0.05$  //The most probable value for r after multiple tests.
  //Loop until no more features can be selected.
  //Use correlation r generated in
  //Algorithm 1.
  for each column (feature) in SF1($C_i \in SF1$)
      if $r_{Ci} > \lambda$ then
          $SF \leftarrow SF \cup C_i$
      end if
  end for
  return SF
```

Algorithm 3 The Proposed Algorithm (ScC)

```
Input: Original Dataset DS, Classification Attribute CA
Output: Classification Accuracy CAcc
  run Algorithm 1
  run Algorithm 2
//Loop for the number of classification models $CM_{i=1->m}$
  for each $CM_{i=1->m}$
      run $CM_i$
      $CM_i \leftarrow$ Acc
  end for
  Sort $CM_{i=1->m}$ in ascending order
  Choose CM[1] as the best model for the given DS
  based on Acc.
```

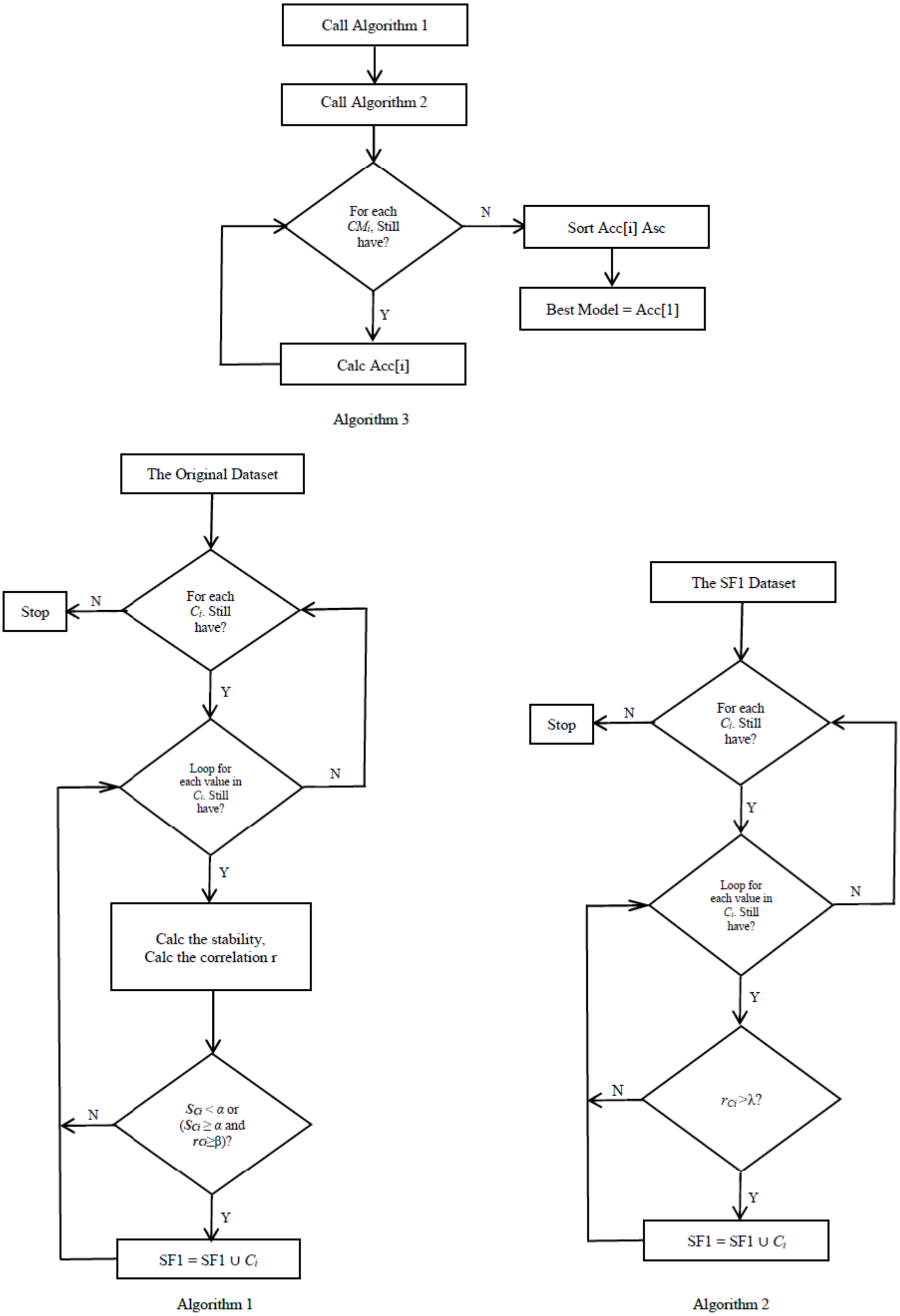
V. EXPERIMENTAL RESULTS

This section reports the experimental results achieved by the proposed feature selection algorithm as well as other state-of-the-art dimensionality reduction methods using the number of datasets mentioned earlier in this paper. The first goal of the experiments was to evaluate the effectiveness of the ScC. The effectiveness of the proposed feature selection algorithm was evaluated using four different classification algorithms: logistic regression, fast large margin, random forest, and gradient boosted trees. The performance of the proposed algorithm was investigated with respect to its reduction rate, classification accuracy, AUC, and the F-measure. The second goal of the experiment was to compare the ScC performance with the results reported by alternative algorithms. A comparison between ScC and the other three dimensionality reduction algorithms explained in Section III was made. For each algorithm, a set of experiments was performed using the nine different benchmark datasets given in Table 1. The adopted classification algorithms and sets of experiments are described in the following subsections.

By viewing these results, one important conclusion can be drawn. This revealed that ScC proved to be very effective for selecting the minimal set of relevant features (reduct). The results also showed that ScC outperformed the other three algorithms.

The two key factors to consider when developing a feature selection algorithm are the performance accuracy and the dataset reduction rate. Both aspects are in opposition because increasing the dataset reduction rate decreases the accuracy rate and vice versa [56]. This is the challenge addressed in this study to prove the efficiency of the proposed algorithm. The accuracy is computed using:

$$\text{Accuracy} = ((\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{TN} + \text{FN})) \tag{7}$$


Call Algorithm 1
Call Algorithm 2
For each CM_i, Still have?
N
Sort Acc[i] Asc
Best Model = Acc[1]
Y
Calc Acc[i]
The Original Dataset
Stop
N
For each C_i. Still have?
Y
Loop for each value in C_i. Still have?
N
Y
Calc the stability, Calc the correlation r
$S_{Ci} < \alpha$ or ($S_{Ci} \geq \alpha$ and $r_{Ci} \geq \beta$)?
N
Y
SF1 = SF1 ∪ C_i
The SF1 Dataset
Stop
N
For each C_i. Still have?
Y
Loop for each value in C_i. Still have?
N
Y
$r_{Ci} > \lambda$?
N
Y
SF1 = SF1 ∪ C_i

Algorithm 3

Algorithm 1

Algorithm 2

FIGURE 1. The scheme of the proposed model.

where TP is the true positives, TN is the true negatives, FP is the false positive, and FN is the false negatives. The proposed algorithm generally obtained better accuracy, AUC, F-measure, and reduction results than the alternative algorithms.

It is important to emphasize that although a better reduction rate can be obtained by algorithms, the accuracy rate is lower, such as in the austra dataset where the reduction rate of the PCA (85.7%) is higher than the reduction rate of ScC (64.3%), but the accuracy rate of ScC using logistic regression (86.78%) was higher than that of PCA (68.73%). These results represent the high quality of the reduct generated by ScC. Many similar examples are presented in the previous tables. From the results above, it can be concluded that the performance of ScC is competitive with the other methods used.

Based on the given results, it is concluded that there is no best classifier or best dimensionality reduction algorithm for all datasets. The importance of a classification system is based on both classification and feature selection algorithms.

Dimensionality reduction (such as feature selection algorithms) minimize the dimension of the original dataset and thus reduce its complexity, but not all classifiers give better results for the given reduced dataset compared to the original dataset. This is the critical point that leads users to decide on suitable dimensionality reduction and classification algorithms for their tasks. This also supports the trade-off between choosing the performance or complexity of the dataset. Sometimes, a classification algorithm and a dimensionality reduction algorithm both support high performance and low complexity. Such algorithms are the ones that everyone is looking for. Dimensionality reduction is not an easy process. Sometimes, one reduced dataset has a higher performance than another less or more reduced dataset based on number of features in the reduced dataset. This is also important to consider, as shown in Table 3.

The analysis of the results conveyed in the tables showed the following attentions which shows that the ScC method was able to give an accuracy rate better than the rate of original dataset using some classifiers:

- The classification accuracy obtained by the logistic regression classifier (as an example) for the original austra dataset was 85.31% with 14 features. Under the same experimental conditions with only five features selected using the ScC (with reduction rate equals to 64.3%) and by using logistic regression classifier, the prediction rate was 86.78%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 1.47% in the recognition rate. It should be noted that ScC obtained the best accuracy for all classifiers used.
- The classification accuracy obtained by the logistic regression classifier (as an example) for the original heart disease dataset was 79.42% with 13 features. Under the same experimental conditions with only five features selected using the ScC (with reduction rate equals to 61.5%) and by using logistic regression classifier, the prediction rate was 90.92%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 11.5% in the recognition rate. ScC achieved the best accuracy for all the classifiers used.
- The classification accuracy obtained by the random forest classifier for the original phishing dataset was 92.53% with 30 features. Under the same experimental conditions with only five features selected using the ScC (with reduction rate equals to 83.3%) and by using random forest classifier, the prediction rate was 92.56%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 0.03% in the recognition rate. Other classifiers obtained better accuracy for the reduced dataset generated by PCA (gradient boosted trees) and RS (logistic regression, and fast large margin) with small percentage values but with a very low reduction percentage.
- The classification accuracy obtained by the fast large margin classifier for the original sonar dataset was 72.88% with 60 features. Under the same experimental conditions with only twenty-one features selected using the ScC (with reduction rate equals to 65%) and by using fast large margin classifier, the prediction rate was 73.3%. Thus, the reduction of the dimension in the feature space resulted in an improvement of approximately 0.45% in the recognition rate. Other classifiers obtained better accuracy for the reduced dataset generated by PCA (logistic regression, random forest, and gradient boosted trees), but they had a lower reduction percentage.
- The classification accuracy obtained by the gradient-boosted tree classifier for the original iono dataset was 90% with 34 features. Under the same experimental conditions with only eight features selected using the ScC (with reduction rate equals to 76.5%) and by using gradient-boosted tree classifier, the prediction rate was 93%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 3% in the recognition rate. ScC achieved the best accuracy for all the other classifiers.
- The classification accuracy obtained by the fast large margin classifier for the original south-german-credit dataset was 70.18% with 21 features. Under the same experimental conditions with only one feature selected using the ScC (with reduction rate equals to 95.5%) and by using fast large margin classifier, the prediction rate was also 70.18%.
- The classification accuracy obtained by the logistic regression tree classifier for the original spam-base dataset was 61.64% with 58 features. Under the same experimental conditions with only six features selected using the ScC (with reduction rate equals to 89.7%) and by using logistic regression classifier, the prediction rate was 88.51%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 26.87% in

TABLE 2. The length of the reduct of each dataset for each dimensionality reduction method used.

		PCA		RS		WGFS		ScC	
Dataset	**Original**	**PCA**	**Deduction %**	**RS**	**Deduction %**	**WGFS**	**Deduction %**	**ScC**	**Deduction %**
Austra	14	2	85.7	3	78.6	2	85.7	5	64.3
Heart Disease	13	3	76.9	3	76.9	6	53.9	5	61.5
Phishing	30	21	30	23	23.3	1	96.7	5	83.3
Sonar	60	17	71.7	22	63.3	10	83.3	21	65
Iono	34	3	91.2	16	52.9	3	91.2	8	76.5
South-German-Credit	21	1	95.5	2	85.7	3	85.7	1	95.5
Spam-Base	58	2	96.6	17	70.7	12	79.3	6	89.7
Messidor	20	2	90	4	80	4	90	3	85
Pop-Failure	21	1	95.2	2	90.5	5	76.2	2	90.5

TABLE 3. The austra dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.8531	0.6873	0.8579	0.6953	**0.8678**
Fast large margin	0.5556	0.6346	0.6396	0.6396	**0.7819**
Random forest	0.8782	0.5736	0.8579	0.7010	**0.8733**
Gradient boosted trees	0.8326	0.6726	0.8477	0.6701	**0.8586**

TABLE 4. The heart disease dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.7942	0.7017	0.6758	0.7650	**0.9092**
Fast large margin	0.8567	0.6675	0.6542	0.6900	**0.8700**
Random forest	0.7675	0.7017	0.6883	0.7167	**0.8442**
Gradient boosted trees	0.8050	0.6892	0.7033	0.7517	**0.8208**

TABLE 5. The phishing dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.9307	0.9266	**0.9313**	0.5570	0.9047
Fast large margin	0.9307	0.9253	**0.9310**	0.5570	0.9126
Random forest	0.9253	0.8208	0.9237	0.5570	**0.9256**
Gradient boosted trees	0.9405	**0.9358**	0.9332	0.5570	0.9246

TABLE 6. The sonar dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.7288	**0.7803**	0.5258	0.6621	0.7000
Fast large margin	0.7288	0.7121	0.5409	0.5500	**0.7333**
Random forest	0.6833	**0.7333**	0.7136	0.5758	0.7167
Gradient boosted trees	0.7470	**0.7788**	0.6833	0.6773	0.6773

TABLE 7. The iono dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.8500	0.7017	0.7800	0.6500	**0.8600**
Fast large margin	0.8400	0.6675	0.7700	0.6500	**0.8400**
Random forest	0.9500	0.7017	**0.9400**	0.8500	0.9100
Gradient boosted trees	0.9000	0.6892	0.9200	0.8300	**0.9300**

TABLE 8. The south-german-credit dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.7343	**0.7018**	0.6644	0.6958	**0.7018**
Fast large margin	0.7018	**0.7018**	0.6574	0.6888	**0.7018**
Random forest	0.7204	**0.7018**	**0.7018**	**0.7018**	**0.7018**
Gradient boosted trees	0.7474	**0.7018**	**0.7018**	0.7028	**0.7018**

TABLE 9. The spam-base dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.6164	0.7161	0.5548	0.6768	**0.8851**
Fast large margin	0.7146	0.7032	0.4247	0.7688	**0.8227**
Random forest	0.6621	0.6978	0.7329	0.7123	**0.8813**
Gradient boosted trees	0.6271	**0.7042**	0.6058	0.6167	0.6903

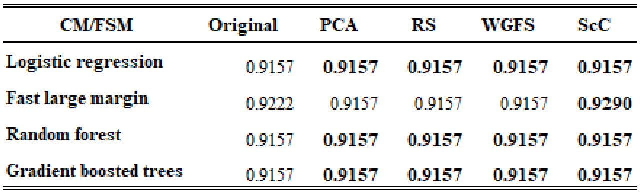
TABLE 10. The messidor dataset.

CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.6231	0.5898	**0.6708**	0.5394	0.5303
Fast large margin	0.7561	0.5898	**0.6522**	0.6442	0.5410
Random forest	0.5303	0.5305	0.5793	0.5714	**0.5915**
Gradient boosted trees	0.5305	0.5945	**0.7121**	0.6707	0.6191

the recognition rate. ScC achieved the best accuracy for all the other classifiers.

- The classification accuracy obtained by the random forest classifier for the original messidor dataset was 53.03% with 20 features. Under the same experimental conditions with only three features selected using the ScC (with reduction rate equals to 85%) and by using

TABLE 11. The pop-failure dataset.



CM/FSM	Original	PCA	RS	WGFS	ScC
Logistic regression	0.9157	**0.9157**	**0.9157**	**0.9157**	**0.9157**
Fast large margin	0.9222	0.9157	0.9157	0.9157	**0.9290**
Random forest	0.9157	**0.9157**	**0.9157**	**0.9157**	**0.9157**
Gradient boosted trees	0.9157	**0.9157**	**0.9157**	**0.9157**	**0.9157**

random forest classifier, the prediction rate was 59.15%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 6.12% in the recognition rate. RS achieved the best accuracy for all the other classifiers.

- The classification accuracy obtained by the fast large margin classifier for the original pop-failure dataset was 92.22% with 21 features. Under the same experimental conditions with only two features selected using the ScC (with reduction rate equals to 90.5%) and by using fast large margin classifier, the prediction rate was 92.9%. Thus, the reduction of the dimension in the feature space resulted in an improvement of 0.68% in the recognition rate. ScC achieved the best accuracy for all the other classifiers.

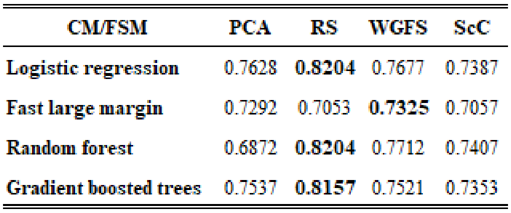
Similar results were obtained in most of the experiments conducted in this study, as shown in Tables 3 to 11, and are displayed in bold color. Figure 2 shows the accuracy rate of the proposed method of reduction with comparison with other methods using the four explained classifiers.

Both the accuracy rate and the reduction rate are important, and both play a role in improving the performance of reduction algorithms. The following formula is used as a measure of performance [58], and it is used to calculate the performance of the proposed and other feature selection algorithms for the sake of comparison:

$$Performance = 2^{*}((Reduction\ rate^{*}Accuracy\ rate)/(Reduction\ rate + Accuracy\ rate)) \quad (8)$$

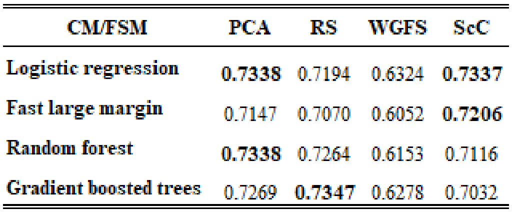
The performance of the hybrid approach, which consists of both the accuracy rate and the reduction rate that resulted earlier, was calculated for each dataset. The results are shown in Tables 12 to 20.

TABLE 12. The performance of the hybrid approach and the other DR methods using the austra dataset.



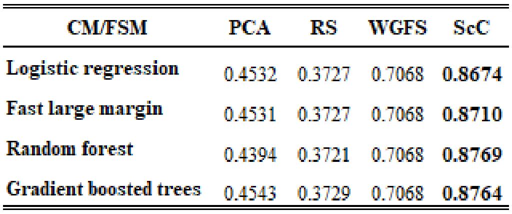
CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.7628	**0.8204**	0.7677	0.7387
Fast large margin	0.7292	0.7053	**0.7325**	0.7057
Random forest	0.6872	**0.8204**	0.7712	0.7407
Gradient boosted trees	0.7537	**0.8157**	0.7521	0.7353

TABLE 13. The performance of the hybrid approach and the other DR methods using the heart disease dataset.



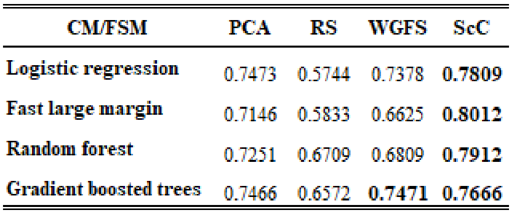
CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	**0.7338**	0.7194	0.6324	**0.7337**
Fast large margin	0.7147	0.7070	0.6052	**0.7206**
Random forest	**0.7338**	0.7264	0.6153	0.7116
Gradient boosted trees	0.7269	**0.7347**	0.6278	0.7032

TABLE 14. The performance of the hybrid approach and the other DR methods using the phishing dataset.



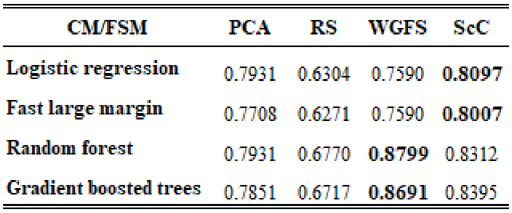
CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.4532	0.3727	0.7068	**0.8674**
Fast large margin	0.4531	0.3727	0.7068	**0.8710**
Random forest	0.4394	0.3721	0.7068	**0.8769**
Gradient boosted trees	0.4543	0.3729	0.7068	**0.8764**

TABLE 15. The performance of the hybrid approach and the other DR methods using the sonar dataset.



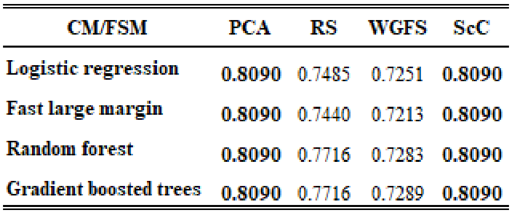
CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.7473	0.5744	0.7378	**0.7809**
Fast large margin	0.7146	0.5833	0.6625	**0.8012**
Random forest	0.7251	0.6709	0.6809	**0.7912**
Gradient boosted trees	0.7466	0.6572	**0.7471**	**0.7666**

TABLE 16. The performance of the hybrid approach the other DR methods using the iono dataset.



CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.7931	0.6304	0.7590	**0.8097**
Fast large margin	0.7708	0.6271	0.7590	**0.8007**
Random forest	0.7931	0.6770	**0.8799**	0.8312
Gradient boosted trees	0.7851	0.6717	**0.8691**	0.8395

TABLE 17. The performance of the hybrid approach and the other DR methods using the south-german-credit dataset.



CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	**0.8090**	0.7485	0.7251	**0.8090**
Fast large margin	**0.8090**	0.7440	0.7213	**0.8090**
Random forest	**0.8090**	0.7716	0.7283	**0.8090**
Gradient boosted trees	**0.8090**	0.7716	0.7289	**0.8090**

The results shown in Tables 12 to 20 shows that ScC produced the optimal performance for all datasets except austra, messidor, and pop-failure (with small difference) using

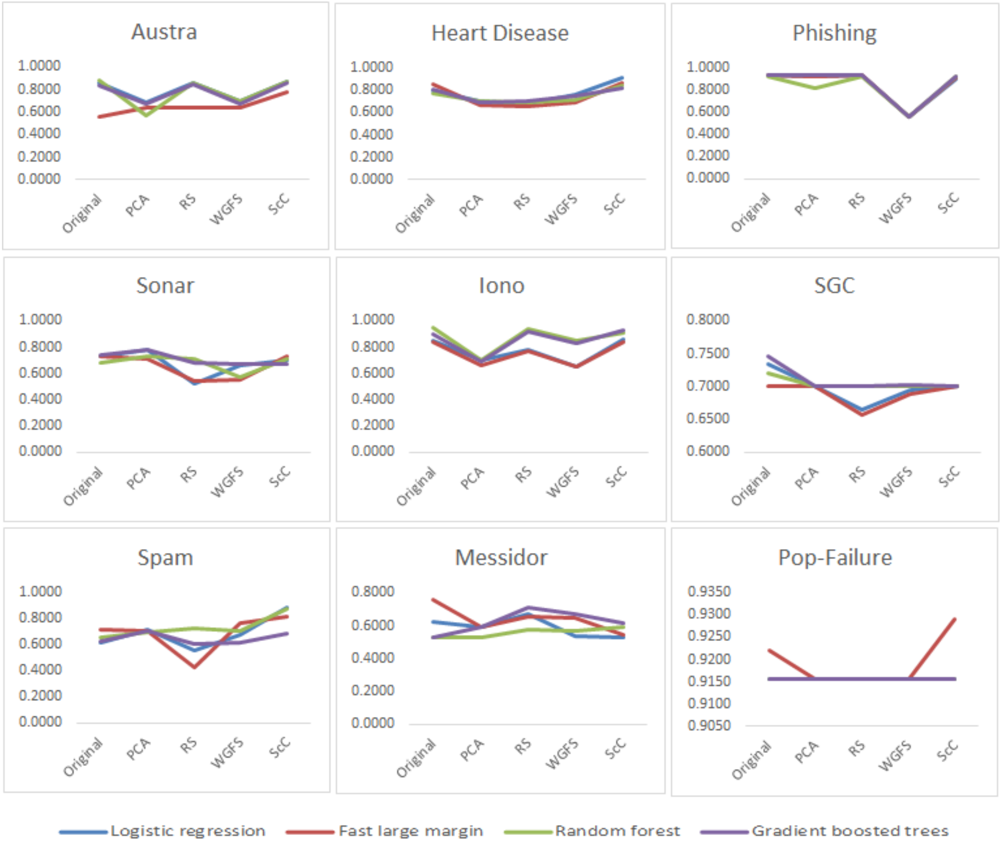


FIGURE 2. The accuracy rate of the hybrid approach using all the datasets.

TABLE 18. The performance of the hybrid approach and the other DR methods using the spam-base dataset.

CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.8225	0.6217	0.7303	**0.8910**
Fast large margin	0.8139	0.5306	0.7807	**0.8582**
Random forest	0.8103	0.7197	0.7505	**0.8891**
Gradient boosted trees	**0.8146**	0.6525	0.6938	0.7802

TABLE 19. The performance of the hybrid approach and the other DR methods using the messidor dataset.

CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	0.7126	**0.7297**	0.6745	0.6531
Fast large margin	0.7126	0.7186	**0.7509**	0.6612
Random forest	0.6675	0.6720	**0.6990**	0.6976
Gradient boosted trees	0.7161	0.7535	**0.7686**	0.7164

TABLE 20. The performance of the hybrid approach and the other DR methods using the pop-failure dataset.

CM/FSM	PCA	RS	WGFS	ScC
Logistic regression	**0.9335**	0.9103	0.8318	0.9103
Fast large margin	**0.9335**	0.9103	0.8318	0.9169
Random forest	**0.9335**	0.9103	0.8318	0.9103
Gradient boosted trees	**0.9335**	0.9103	0.8318	0.9103

logistic regression and fast large margin classifiers. Based on this, an important conclusion was reached, which highlighted that the performance of the ScC hybrid approach is competitive with the state-of-the-art methods used in this work. ScC shows that it is an innovator in generating important and resonable reducts with high accuracy.

Figure 3 shows the performance of the proposed hybrid approach of reduction with comparison with other methods using the four discussed classifiers.

In addition to accuracy and selected features (reduct), AUC and F-measure are another performance indicators used for performance comparison. AUC is the measure of the ability of a classifier to distinguish between classes. Higher value of the AUC represents better performance of the model at distinguishing between the positive and negative classes. F-Measure provides a way to combine both precision and recall into a single measure that captures both properties.

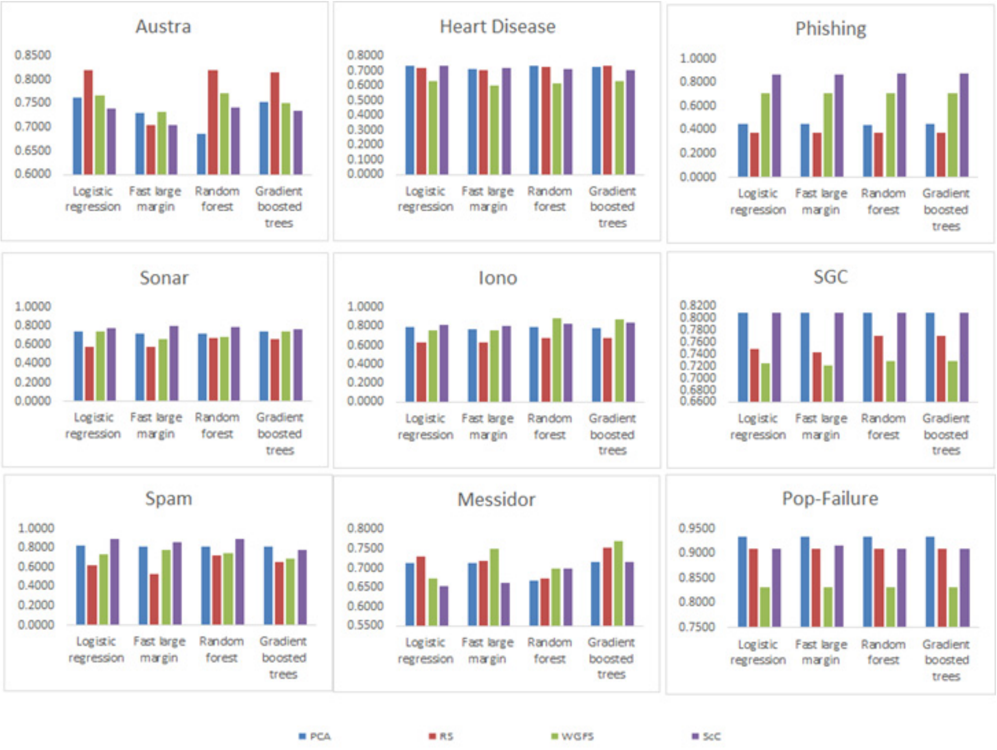


FIGURE 3. The performance of the hybrid approach using all the datasets.

Precision measures the number of positive class predictions that belong to the positive class. Recall measures the number of positive class predictions made out of all positive examples in the dataset. The F-measure is computed using:

$$\text{F-measure} = 2((\text{precision} * \text{recall})/(\text{precision} + \text{recall})) \tag{9}$$

where precision is computed using:

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{10}$$

and recall is computed using:

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FP}) \tag{11}$$

F-measure and AUC rates were computed using the logistic regression classifier. F-measure rates are shown in Table 21. Table 21 shows that the proposed model gives high rate for all dataset with distinguish results for six of them. Table 22 shows that the AUC test gives high rates for all datasets with distinguish results for seven of them.

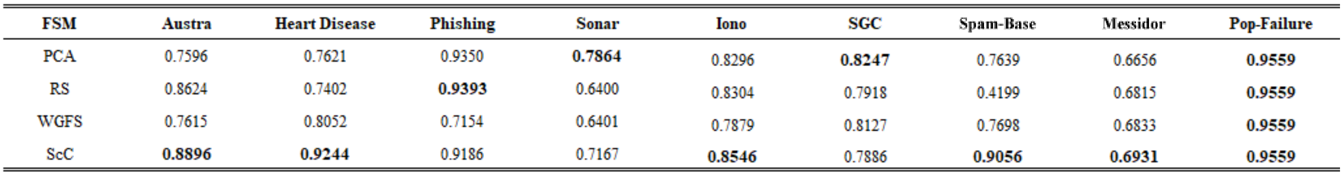
The advantages of the proposed method over the other state-of-the-art- methods used in this article can be summarized as follows:

- The accuracy of ScC is higher than that of the other methods for most of the classifiers used. This conclusion reflects the high quality of the reduct generated by the proposed method.
- The performance of ScC which considers both the accuracy, and the reduction rate is mostly higher than the performance of the other methods.
- F-measure rate for ScC is higher than that rate given by most of the other methods.
- AUC rate for ScC is higher than that rate given by most of the other methods.

The time complexity analysis of ScC is conducted using the big-O notation [59]. The analysis is divided into two parts: Algorithm 1 and Algorithm 2. As stated in Section III, Algorithm 1 consists of the computation of stability and correlation. Stability uses the mode function that exists in the loop of features. Let N is the number of input features (the outer loop), and T_N is the time complexity for the outer loop in Algorithm 1. Also, let T_M is the time complexity for the mode function of M values (which is $O(M)$) and T_{A1} is the time complexity for Algorithm 1. In the worst-case scenario, $T_{A1} = O(N) * O(M)$ which yield to $O(NM)$. If $N = M$ (which is very far from the reality where N is very small number comparing with M) then $T_{A1} = O(N^2)$.

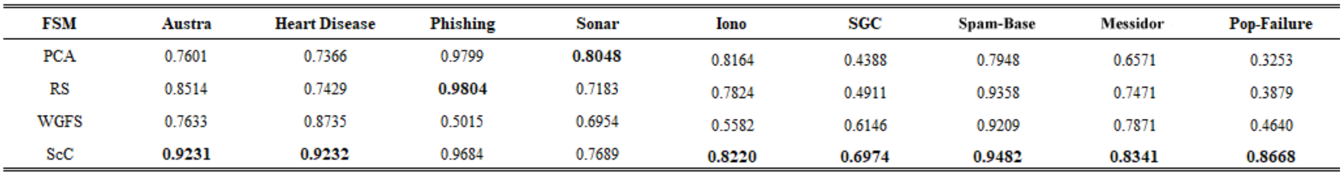
For Algorithm 2, Let N is the number of input features (the outer loop), and T_N is the time complexity for the outer loop in Algorithm 2. Also, let T_M is the time complexity for the inner loop of M values (which is $O(M)$) and T_{A2} is the time complexity for Algorithm 2. In the worst-case scenario, Algorithm 2 requires $O(NM)$ time-complexity. Again, if $N = M$ (which is very far from the reality where N is very small number comparing with M) then $T_{A2} = O(N^2)$. As such,

TABLE 21. The F-measure rates for the nine datasets.



FSM	Austra	Heart Disease	Phishing	Sonar	Iono	SGC	Spam-Base	Messidor	Pop-Failure
PCA	0.7596	0.7621	0.9350	**0.7864**	0.8296	**0.8247**	0.7639	0.6656	**0.9559**
RS	0.8624	0.7402	**0.9393**	0.6400	0.8304	0.7918	0.4199	0.6815	**0.9559**
WGFS	0.7615	0.8052	0.7154	0.6401	0.7879	0.8127	0.7698	0.6833	**0.9559**
ScC	**0.8896**	**0.9244**	0.9186	0.7167	**0.8546**	0.7886	**0.9056**	**0.6931**	**0.9559**

TABLE 22. The AUC rates for the nine datasets.



FSM	Austra	Heart Disease	Phishing	Sonar	Iono	SGC	Spam-Base	Messidor	Pop-Failure
PCA	0.7601	0.7366	0.9799	**0.8048**	0.8164	0.4388	0.7948	0.6571	0.3253
RS	0.8514	0.7429	**0.9804**	0.7183	0.7824	0.4911	0.9358	0.7471	0.3879
WGFS	0.7633	0.8735	0.5015	0.6954	0.5582	0.6146	0.9209	0.7871	0.4640
ScC	**0.9231**	**0.9232**	0.9684	0.7689	**0.8220**	**0.6974**	**0.9482**	**0.8341**	**0.8668**

the worst-case time complexity for ScC is $T_{A1} + T_{A2}$ which yield to O(NM) + O(NM), which is O(NM) when all features extend to infinity.

VI. CONCLUSION

ScC, which is a novel feature selection algorithm based on stability and correlation, was introduced. The aim was to produce an algorithm that generates a minimal set of high-confidence features that can produce an optimized accuracy. Attention was paid to evaluate the extent to which the dimension of the minimal dataset can affect classification performance. The proposed algorithm is evaluated using benchmark datasets. The ScC performance was compared to the other three state-of-the-art dimensionality reduction methods considered in this study, namely, PCA, WGFS, and RS.

The contributions of this article can be summarized as follows: (1) Provide the research community with a deep organized study on the feature selection problem as one of the important problems in different disciplines, including machine learning, AI, data mining, data science, and many others. (2) A new feature selection evolutionary algorithm is introduced to produce a minimal set of significant and distinctive features based on the stability of the features and their correlations. The reduced dataset helps classification algorithms to work faster and more efficiently. (3) Use an effective hybrid approach for improving the classification performance based on the reduction rate and the accuracy rate of four selected classifiers: logistic regression, fast large margin, deep learning, random forest, and gradient boosted trees. (4) Extensive evaluation of the proposed algorithm (ScC) using a benchmark dataset. Likewise, an extensive evaluation of the proposed hybrid approach using the same datasets was performed.

The main findings of this study can be shortened as follows. ScC is highly competitive and promising in terms of the reduction rate. The accuracy of the minimal datasets generated by the proposed feature selection algorithm also outperforms the accuracy of the data sets generated by other methods. Meanwhile, the performance of the proposed hybrid approach of ScC and the accuracy rate outperformed others (PCA and accuracy rate, RS and accuracy rate, and WGFS and accuracy rate). ScC can be improved by enhancing its efficiency through the process of reducing its computational cost. Finally, one conclusion was drawn from the results, which confirmed that the proposed feature selection algorithm and the proposed hybrid approach play a key role in improving the classification performance.

Selecting mandatory features requires the stability of the features. The low variation in feature values frequently produces moderately constant features (unstable features), which usually do not add value to the mandatory feature set. The instability of such features generally reduces the sureness and confidence of the selected features, which leads to low classification accuracy.

Future research will focus on the following:

1) Extend the proposed algorithm to be able to work with big data.
2) Extended the analysis of the proposed algorithm by comparing it with various filtering methods such as QPFS.
3) Extend the analysis to include the efficiency of the proposed algorithm in terms of the run time and compare it with the efficiency of other methods of dimensionality reduction.

ACKNOWLEDGMENT

The author wishes to thank his wife, daughters, and friends for their continuous support.

REFERENCES

[1] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003.

[2] A. K. Das, S. Sengupta, and S. Bhattacharyya, "A group incremental feature selection for classification using rough set theory based genetic algorithm," *Appl. Soft Comput.*, vol. 65, pp. 400–411, Apr. 2018.

[3] B. Ma and Y. Xia, "A tribe competition-based genetic algorithm for feature selection in pattern classification," *Appl. Soft Comput.*, vol. 58, pp. 328–338, Sep. 2017.

[4] Y. Zhang, D. Gong, Y. Hu, and W. Zhang, "Feature selection algorithm based on bare bones particle swarm optimization," *Neurocomputing*, vol. 148, pp. 150–157, Jan. 2015.

[5] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, and A. Nowe, "A survey on filter techniques for feature selection in gene expression microarray analysis," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 9, no. 4, pp. 1106–1119, Jul. 2012.

[6] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997. [Online]. Available: http://www.robotics.stanford.edu/~gjohn

[7] M. A. Hall, "Feature selection for discrete and numeric class machine learning," Dept. Comput. Sci., Univ. Waikato, Hamilton, New Zealand, Tech. Rep. 99/04, 1999.

[8] T. Z. Phyu and N. N. Oo, "Performance comparison of feature selection methods," in *Proc. 3rd Int. Conf. Control, Mechatronics Autom. (ICCMA)*, vol. 42, Feb. 2016, pp. 1–4, doi: 10.1051/matecconf/20164206002.

[9] M. Ghosh, S. Adhikary, K. K. Ghosh, and A. Sardar, "Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods," *Med. Biol. Eng. Comput.*, vol. 57, no. 1, pp. 159–176, 2019.

[10] N. Hoque, M. Singh, and D. K. Bhattacharyya, "EFS-MI: An ensemble feature selection method for classification," *Complex Intell. Syst.*, vol. 4, no. 2, pp. 105–118, Jun. 2018.

[11] Y. B. Wah, N. Ibrahim, H. A. Hamid, S. Abdul-Rahman, and S. Fong, "Feature selection methods: Case of filter and wrapper approaches for maximizing classification accuracy," *Pertanika J. Sci. Technol.*, vol. 26, no. 1, pp. 329–340, 2018.

[12] Y. Liu, "A comparative study on feature selection methods for drug discovery," *J. Chem. Inf. Comput. Sci.*, vol. 44, no. 5, pp. 1823–1828, Sep. 2004.

[13] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.

[14] F. Pourpanah, Y. Shi, C. P. Lim, Q. Hao, and C. J. Tan, "Feature selection based on brain storm optimization for data classification," *Appl. Soft Comput.*, vol. 80, pp. 761–775, Jul. 2019.

[15] F. Fleuret, "Fast binary feature selection with conditional mutual information," *J. Mach. Learn. Res.*, vol. 5, pp. 1531–1555, Nov. 2004.

[16] W. Ke, C. Wu, Y. Wu, and N. N. Xiong, "A new filter feature selection based on criteria fusion for gene microarray data," *IEEE Access*, vol. 6, pp. 61065–61076, 2018.

[17] Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper–filter feature selection algorithm using a memetic framework," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 1, pp. 70–76, Feb. 2007.

[18] M. Mohtashami and M. Eftekhari, "A hybrid filter-based feature selection method via hesitant fuzzy and rough sets concepts," *Iran. J. Fuzzy Syst.*, vol. 16, no. 2, pp. 165–182, 2019.

[19] B. Xue, M. Zhang, and W. N. Browne, "A comprehensive comparison on evolutionary feature selection approaches to classification," *Int. J. Comput. Intell. Appl.*, vol. 14, no. 2, Jun. 2015, Art. no. 1550008.

[20] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "A review of feature selection methods on synthetic data," *Knowl. Inf. Syst.*, vol. 34, no. 3, pp. 483–519, 2013.

[21] M. Cherrington, F. Thabtah, J. Lu, and Q. Xu, "Feature selection: Filter methods performance challenges," in *Proc. Int. Conf. Comput. Inf. Sci. (ICCIS)*, 2019, pp. 1–4, doi: 10.1109/ICCISci.2019.8716478.

[22] A. Choudhary, S. Kolhe, and H. Rajkamal, "Performance evaluation of feature selection methods for mobile devices," *Int. J. Eng. Res. Appl.*, vol. 3, no. 6, pp. 587–594, 2013.

[23] A. U. Haq, A. Zeb, Z. Lei, and D. Zhang, "Forecasting daily stock trend using multi-filter feature selection and deep learning," *Expert Syst. Appl.*, vol. 168, Feb. 2021, Art. no. 114444.

[24] S. S. Darshan and C. Jaidhar, "Performance evaluation of filter-based feature selection techniques in classifying portable executable files," *Proc. Comput. Sci.*, vol. 125, pp. 346–356, Oct. 2018.

[25] Y. Aphinyanaphongs, L. D. Fu, Z. Li, E. R. Peskin, E. Efstathiadis, C. F. Aliferis, and A. Statnikov, "A comprehensive empirical comparison of modern supervised classification and feature selection methods for text categorization," *J. Assoc. Inf. Sci. Technol.*, vol. 65, no. 10, pp. 1964–1987, Oct. 2014.

[26] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," *Inf. Sci.*, vol. 282, pp. 111–135, Oct. 2014.

[27] I. Inza, P. Larrañaga, R. Blanco, and A. J. Cerrolaza, "Filter versus wrapper gene selection approaches in DNA microarray domains," *Artif. Intell. Med.*, vol. 31, no. 2, pp. 91–103, 2004.

[28] L. Du, X. Lv, C. Ren, and Y. Chen, "A filter-based unsupervised feature selection method via improved local structure preserving," in *Proc. 5th Int. Conf. Big Data Inf. Anal. (BigDIA)*, 2019, pp. 162–169, doi: 10.1109/BigDIA.2019.8802793.

[29] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, and M. Lang, "Benchmark for filter methods for feature selection in high-dimensional classification data," *Comput. Statist. Data Anal.*, vol. 143, Mar. 2020, Art. no. 106839.

[30] L. Al-Shalabi, "Rough set-based reduction of incomplete medical datasets by reducing the number of missing values," *Int. Arab J. Inf. Technol.*, vol. 16, no. 2, pp. 203–210, 2019.

[31] P. Uddin, A. Mamun, and A. Hossain, "PCA-based feature reduction for hyperspectral remote sensing image classification," *IETE Tech. Rev.*, vol. 38, no. 4, pp. 377–396, 2020, doi: 10.1080/02564602.2020.1740615.

[32] J. O. Sinayobye, K. S. Kaawaase, F. N. Kiwanuka, and R. Musabe, "Hybrid model of correlation based filter feature selection and machine learning classifiers applied on smart meter data set," in *Proc. IEEE/ACM Symp. Softw. Eng. Afr. (SEiA)*, May 2019, pp. 1–10, doi: 10.1109/SEiA.2019.00009.

[33] M. Hu, E. C. C. Tsang, Y. Guo, and W. Xu, "Fast and robust attribute reduction based on the separability in fuzzy decision systems," *IEEE Trans. Cybern.*, early access, Jan. 5, 2021, doi: 10.1109/TCYB.2020.3040803.

[34] S. Bashir, M. Imran, S. Batool, M. Imran, M. I. Ahmad, F. M. Malik, M. Salman, A. Wakeel, and U. Ali, "Frequency limited & weighted model reduction algorithm with error bound: Application to discrete-time doubly fed induction generator based wind turbines for power system," *IEEE Access*, vol. 9, pp. 9505–9534, 2021, doi: 10.1109/ACCESS.2021.3049575.

[35] M. Soheili and A. M. Eftekhari-Moghadam, "DQPFS: Distributed quadratic programming based feature selection for big data," *J. Parallel Distrib. Comput.*, vol. 138, pp. 1–14, Apr. 2020.

[36] M. Soheili and A. E. Moghadam, "Feature selection in multi-label classification through MLQPFS," in *Proc. 4th Int. Conf. Control, Instrum., Autom. (ICCIA)*, 2016, pp. 430–434, doi: 10.1109/ICCIAutom.2016.7483201.

[37] S. Ramirez-Gallego, H. Mourino-Talin, D. Martinez-Rego, V. Bolon-Canedo, J. M. Benitez, A. Alonso-Betanzos, and F. Herrera, "An information theory-based feature selection framework for big data under apache spark," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1441–1453, Sep. 2018, doi: 10.1109/TSMC.2017.2670926.

[38] P. Bermejo, J. A. Gámez, and J. M. Puerta, "Adapting the CMIM algorithm for multilabel feature selection. A comparison with existing methods," *Expert Syst.*, vol. 35, no. 1, Feb. 2018, Art. no. e12230, doi: 10.1111/exsy.12230.

[39] S. Mishra, U. Sarkar, S. Taraphder, S. Datta, D. Swain, R. Saikhom, S. Panda, and M. Laishram, "Multivariate statistical data analysis-principal component analysis (PCA)," *Int. J. Livestock Res.*, vol. 7, no. 5, pp. 60–78, 2017, doi: 10.5455/ijlr.20170415115235.

[40] R. Bro and A. K. Smilde, "Principal component analysis," *Anal. Methods*, vol. 6, no. 9, pp. 2812–2831, 2014.

[41] Z. Gniazdowski, "New interpretation of principal components analysis," 2017, *arXiv:1711.10420*.

[42] F. Song, Z. Guo, and D. Mei, "Feature selection using principal component analysis," in *Proc. Int. Conf. Syst. Sci., Eng. Design Manuf. Inf.*, 2010, pp. 27–30, doi: 10.1109/ICSEM.2010.14.

[43] K. Velusamy and R. Manavalan, "Performance analysis of unsupervised classification based on optimization," *Int. J. Comput. Appl.*, vol. 42, no. 19, pp. 22–27, Mar. 2012.

[44] C. Velayutham and K. Thangavel, "Unsupervised quick reduct algorithm rough set theory," *J. Electron. Sci. Technol.*, vol. 9, no. 3, pp. 193–201, 2011.

[45] M. Ringsquandl, S. Lamparter, S. Brandt, T. Hubauer, and R. Lepratti, "Semantic-guided feature selection for industrial automation systems," in *Proc. ISWC*, 2015, pp. 225–240, doi: 10.1007/978-3-319-25010-6_13.

[46] C.-Y. J. Peng, K. L. Lee, and G. M. Ingersoll, "An introduction to logistic regression analysis and reporting," *J. Educ. Res.*, vol. 96, no. 1, pp. 3–14, Oct. 2002, doi: 10.1080/00220670209598786.

[47] G. Wisniewski and F. Yvon, ''Fast large-margin learning for statistical machine translation,'' *Int. J. Comput. Linguistics Appl.*, vol. 4, no. 2, pp. 45–62, 2013.

[48] L. Breiman, ''Random forests,'' *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[49] L. Al Shalabi, Z. Shaaban, and B. Kasasbeh, ''Data mining: A preprocessing engine,'' *J. Comput. Sci.*, vol. 2, no. 9, pp. 735–739, 2006.

[50] J. R. Quinlan, ''Unknown attribute values in induction,'' in *Proc. 6th Int. Workshop Mach. Learn.*, 1989, pp. 164–168.

[51] A. Natekin and A. Knoll, ''Gradient boosting machines, a tutorial,'' *Front. Neurorobot.*, vol. 7, no. 21, pp. 1–21, Dec. 2013, doi: 10.3389/fnbot.2013.00021.

[52] U. M. Khaire and R. Dhanalakshmi, ''Stability of feature selection algorithm: A review,'' *J. King Saud Univ.-Comput. Inf. Sci.*, to be published, doi: 10.1016/j.jksuci.2019.06.012.

[53] P. Yang, B. B. Zhou, J. Y. H. Yang, and A. Y. Zomaya, ''Stability of feature selection algorithms and ensemble feature selection methods in bioinformatics,'' in *Biological Knowledge Discovery Handbook*, vol. 14, M. Elloumi and A. Y. Zomaya, Eds. Hoboken, NJ, USA: Wiley, 2013, pp. 333–352.

[54] A. Kalousis, J. Prados, and M. Hilario, ''Stability of feature selection algorithms: A study on high-dimensional spaces,'' *Knowl. Inf. Syst.*, vol. 12, no. 1, pp. 95–116, Dec. 2006.

[55] É. Perthame, C. Friguet, and D. Causeur, ''Stability of feature selection in classification issues for high-dimensional correlated data,'' *Statist. Comput.*, vol. 26, no. 4, pp. 783–796, May 2015.

[56] D. Dernoncourt, B. Hanczar, and J.-D. Zucker, ''Analysis of feature selection stability on high dimension and small sample data,'' *Comput. Statist. Data Anal.*, vol. 71, pp. 681–693, Mar. 2014, doi: 10.1016/j.csda.2013.07.012.

[57] J. Biesiada and W. Duch, ''Feature selection for high-dimensional data—A Pearson redundancy based filter,'' *Adv. Soft Comput.*, vol. 45, pp. 242–249, Oct. 2007.

[58] H. J. Escalante, M. Marin-Castro, A. Morales-Reyes, M. Graff, A. Rosales-Pérez, M. Montes-y-Gómez, C. A. Reyes, and J. A. Gonzalez, ''MOPG: A multi-objective evolutionary algorithm for prototype generation,'' *Pattern Anal. Appl.*, vol. 20, no. 1, pp. 33–47, Feb. 2017.

[59] T. H. Cormen, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.

LUAI AL-SHALABI received the Ph.D. degree in computer science with a focus on data mining, in 2000. He is currently an Associate Professor of data mining with Arab Open University, Kuwait Branch. He has over 25 publications in reputable local and international conferences and journals, mostly on data mining and its applications. His research interests include data mining, data science, knowledge discovery, and machine learning.