Multimedia Tools and Applications
https://doi.org/10.1007/s11042-022-14132-z

Wide-ranging approach-based feature selection for classification

Hemanta Kumar Bhuyan[1] · M Saikiran[1] · Murchhana Tripathy[2] · Vinayakumar Ravi[3]

Received: 5 December 2021 / Revised: 20 October 2022 / Accepted: 25 October 2022

Abstract
Feature selection methods have been issued in the context of data classification due to redundant and irrelevant features. The above features slow the overall system performance, and wrong decisions are more likely to be made with extensive data sets. Several methods have been used to solve the feature selection problem for classification, but most are specific to be used only for a particular data set. Thus, this paper proposes wide-ranging approaches to solve maximum feature selection problems for data sets. The proposed algorithm analytically chooses the optimal feature for classification by utilizing mutual information (MI) and linear correlation coefficients (LCC). It considers linearly and nonlinearly dependent data features for the same. The proposed feature selection algorithm suggests various features used to build a substantial feature subset for classification, effectively reducing irrelevant features. Three different datasets are used to evaluate the performance of the proposed algorithm with classifiers which requires a higher degree of features to have better accuracy and a lower computational cost. We considered probability value (p value <0.05) for feature selection in experiments on different data sets, then the number of features is selected (such as 7, 5, and 6 features from mobile, heart, and diabetes data set, respectively). Various accuracy is considered

✉ Hemanta Kumar Bhuyan
hmb.bhuyan@gmail.com

M Saikiran
kiranmunagala2@gmail.com

Murchhana Tripathy
murchhanatripathy@gmail.com

Vinayakumar Ravi
vravi@pmu.edu.sa

[1] Department of Information Technology, Vignan's Foundation for Science, Technology & Research (Deemed to be University), Guntur, AP, India

[2] Information Systems & Technology, T A Pai Management Institute, Manipal Academy of Higher Education, Manipal, India

[3] Center for Artificial Intelligence, Prince Mohammad Bin Fahd University, Khobar 34754, Saudi Arabia

Published online: 17 November 2022

with different classifiers; for example, classifier Nearest_Neighbors made accuracy such as 0.92225, 0.88333, 0.86250 for mobile, heart, and diabetes data sets, respectively. The proposed model is adequate as per the evaluation of several real-world data sets.

Keywords Feature selection · Mutual information · Linear correlation coefficient · Classification · Data mining · Confusion matrix · P-values

1 Introduction

Feature selection has become the primary concern with the rapid growth of high dimensional data in many disciplines, such as text mining, image mining, visual classification, bioinformatics etc. Advanced computer and database technologies play a vital role in information processing, information recovery with discriminative projection for feature selection [42] and predicting adverse drug interaction [47]. The embedding method was developed with Adaptive Similarity Embedding for Unsupervised Multi-View Feature Selection (ASE-UMFS). This technique decreases the high-to-low dimensional data and unifies different views into a combination weight matrix [41]. But sometimes, handling such a large quantity of data is difficult since traditional machine learning methods only work well on small data sets. Feature selection addresses this by deleting insignificant, redundant, and noisy information. It improves the effectiveness of the learning algorithm, reduces assessment costs, and provides a better understanding of data sets.

Usable feature selection algorithms can be divided widely into two different groups [23, 32]: (a) filter methods and (b) wrapper methods. Filter methods are independent of the learning algorithm and are very cheap to be used. They rely only on the characteristics of the variable. But the risk with these methods is that they may select sub-sets of features that would not correlate to the generative model selected. Compared to this, the wrapper methods directly use the induction algorithm to evaluate feature sub-sets. They usually exceed filter methods in terms of predictive performance but are usually more domain-specific. Recently, both feature and sub-feature selection are developed with different approaches in [4, 6, 7]. Similarly, diversity approaches are used for feature selection, like graph-based feature selection for biological data sets [20] and hybrid feature selection model for predicting students' performance [44].

Large datasets such as Census data set, biological data sets, mobile data set etc., pose a significant challenge to the feature selection model. Thus, computational difficulties handling such data sets may lead to imprecise classification. Computational complexity usually increases when a large amount of data needs to be classified. A dataset of significant size will impede classifier development and cause system failure because of insufficient memory. Many large-scale datasets include noisy, redundant, and unusable features, which pose a significant challenge to data mining models.

As contempt responsiveness rises for feature selection, existing solutions have been found with the individual performance of selecting the right features for processing classification. The adaptive selective process is not suitable for identifying features from any data set. Thus, it developed an advanced filter method for feature evaluation, which is very useful for feature selection. This paper proposed the framework for solving the feature selection problem on various large data sets using an Advanced Feature Selection Algorithm based on mutual information and correlation coefficient. It consists of two phases. Before processing, the first phase conducts a preliminary search to identify and remove irrelevant or redundant features.

Thus, it shortens the search range from the feature space to the second phase, where the classifiers are used to select the features. This paper utilized the advanced feature selection approach with different classifiers. Before evaluating any data set, the whole data set is split into the training and testing data sets. We also changed the size of the data sets more than once for better performance. We have removed constant, quasi-constant, and duplicate features from the data set. The feature set is also considered as duplicate and non-duplicate features during evaluation. The proposed approach also uses p-values to evaluate the feature values statistically. Several classifiers, such as Nearest Neighbours, Linear SVM, Gradient Boosting, Decision tree, Random Forest, etc., have been evaluated with comparative accuracy. We applied various testing of the confusion matrix mentioned in the experiment section for the proposed model.

Although various methods are considered to select the features from the different data sets, they developed their model to their requirements. Thus, we were motivated to develop this model with wide range of approaches for any kind of dataset. We can take any kind of data set to select features per our proposed model. We have also proved with theoretical methods of our model.

To summarize, the following list includes the key contributions of this paper.

(1) The proposed feature selection method uses theoretical mutual information analysis (MI) to measure feature dependence on output classes. Classifiers are used for relevant features to evaluate classes with accuracy.
(2) The correlation coefficient approach is applied with theoretical and experimental analysis to get a relationship among features.
(3) The method has a flexible parameter setting. As a result, its results are not dependent on an arbitrary value assigned to a free parameter and thus may be assumed to be objective. Additionally, the proposed method is easy to implement and works in various domains.
(4) We use different methods of confusion matrix to achieve more accuracy through classifiers.
(5) It used threshold values for p-values to filter the features from statistical evaluation

The rest of the paper is organized as follows. Section 2 deals with the related work that was found for this study with a proper background of the paper. Section 3 presents the framework for feature selection-based classification. Section 4 proposes the feature selection algorithms with theoretical and computational complexity analysis. Section 5 analyses the experimental settings. Section 6 contains the details and results of the experiments. Discussion with advantages, limitations, etc. are mentioned in section 7. We finally conclude section 8 with future work.

2 Related work

Feature selection removes redundant and irrelevant features while identifying the most optimized subset of features that describe various class characteristics. Feature selection can be categorized into two general types of methods: (a) filter and (b) wrapper methods [2]. (a) Filter methods have used independent measures such as information, distance, and consistency measures as the criteria for identifying feature relations. In contrast, wrapper algorithms use particular learning algorithms to evaluate the value of the features. Filter methods have less

computational expenses when working with high-dimensional data or large-scale data, but wrapper methods are often much more computationally expensive.

To put it another way, the filtering algorithms evaluated the discriminability of each data feature, filtered out the irrelevant features, and retained the discriminative features, such as LapScore [22]. The spectral regression is used to rate the relative relevance of various features when selecting a feature. Iterative search for the best features without the aid of learning algorithms is the goal of wrapper methods like UFSACO [39], which are intended to be directly part of learners. Regarding feature selection, wrapper methods have to train the learning algorithm repeatedly, making them more time-consuming than filtering and embedded methods. As a general rule, there were existing embedded approaches to learning intrinsic structure through various methods for picking discriminative features. The embedded techniques combine with a machine learning model to form the best single objection function. Many researchers and practitioners in artificial intelligence use sparse learning to choose features for their models [12, 45, 46]. NDFS [30] explicitly enforces a nonnegative restriction on the class indicators learned through spectral clustering for unsupervised feature selection.

A hybrid of wrapper and filter processes can be improved by successfully allowing filter methods to search the function's space with high accurac**y.** In this connection, the new Wrapper-filter selection algorithm (WFSA) has been proposed using a memetic framework, i.e., the genetic algorithm [8, 10, 29] as well as the local search (LS) methods. WFSA focuses on improving the quality and efficiency of classification to find substantial subsets of features. In particular, the filter method refines various features in [5, 9, 11, 26] alternatively by adding or removing features based on the descriptive ranking of features. We emphasize filter methods that can determine the size or classification of each feature. In this regard, we consider the filter methods as filter classification methods and inspect the WFSA method. Experimental work on WFSA, has been conducted on many datasets from the University of California, Irvine (UCI) and numerous microarray data sets. This shows that the results obtained exceed the current methodological odds of literature regarding accuracy, selected dimensions, and classification efficiency. We also evaluate the stability between LS and genetic search to optimize WFSA's lookup performance and reliability.

Furthermore, Wang et, al., suggested improving the demand for large-scale training sets for visual-audio model training using selective class activation mapping (SCAM) and its upgrade (SCAM+) in [43]. Similarly, Guangxiao Ma et al., developed Object-level Semantical Saliency Ranking using Omnidirectional Image in [31]. Chen et al., apply the inter-image nonlocal correspondences for designing a selective fusion network to boost the detection performance. Thus, the overall depth can be coarsely estimated using the newly designed depth-transferring strategy [14]. Abeywickrama et al., developed the "State Of The Affairs (SOTA)" model with collective adaptive systems for the verification of requirements and provided self-adaptive software development [19]. Cloud access security for a feature and feature-based image processing are explained in [1, 13, 33].

In addition to the previously described detection techniques, the KDD Cup 99 dataset was investigated for all the detection techniques. It explained the drawbacks of this dataset, and the results of evaluations using different intrusion detection datasets, such as NSL-KDD [40] and Kyoto 2006+ [38]. In [38], a dimensionality reduction method was proposed where a naive Bayes classifier was used to find the most important features for intrusion detection. Results obtained from the NSL-KDD experiment were encouraging. A Candidate Support Vector based Incremental SVM algorithm (CSV-ISVM in long) was suggested by Chitrakar and Huang [15]. Network intrusion detection was subjected to the algorithm. The IDS with the

CSV-ISVM-based detection engine was tested on the Kyoto 2006+ [38] dataset. The results of their IDS experiments were promising in terms of detection rate and false alarm rate. Real-time network intrusion detection was claimed for the IDS. We compared our model with the other detection systems in this work by examining their dataset.

Based on the above-related feature selection approaches, we considered the filter method-based feature selection for the work. Both theoretical (algorithms and theorems) and experimental analysis have been done as per the proposed model.

3 Framework for feature selection-based classification

We develop a framework for feature selection-based classification for data processing, which is shown in Fig. 1. The framework consists of four stages such as: (a) data collection, (b) data pre-processing, (c) building classification, and (d) producing classification results. The above stages are considered to process all data sets and provide appropriate classification accuracy per the model.

3.1 Data collection

To identify features for classification, the first step is data collection. Two things influence a proposed model's overall design and effectiveness: the type of data sources and the location from where data is collected. Thus, various data sets are gathered from different sources per proposed model. The objective is to make appropriate feature selections and improve classification accuracy. Once the data has been collected, it is divided into training and test data sets. Although the data collected in the test set is only categorized according to prototypes, it is a significant source of information for the model.

3.2 Pre-processing of data

A processing step follows the data collection step, during which the basic features are created, and used for filter methods. In this stage, it considers mutual information and correlation

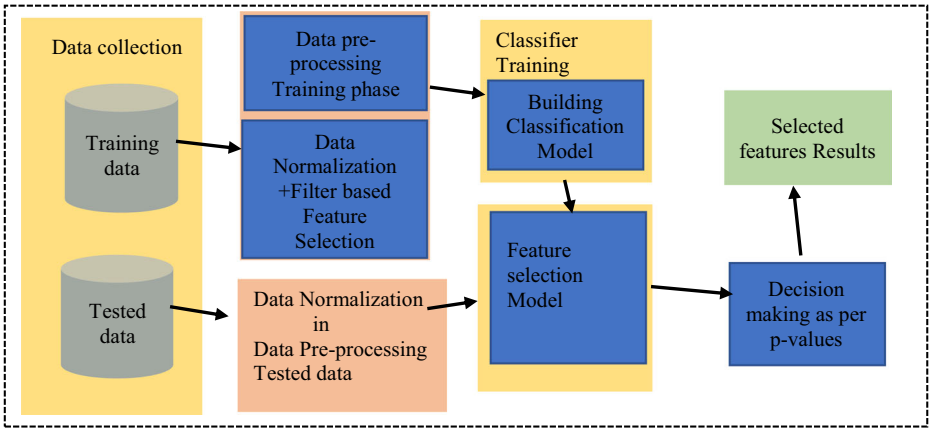

Fig. 1 Framework for feature selection-based classification

coefficient to filter all the features of the considered data set or select features according to their evaluation scores. Three main stages of this phase are mentioned below.

3.2.1 Data conversion

Classifiers need to be trained expect the input data represented as a vector of real numbers. Data conversion transforms each symbolic feature of a dataset into a numerical value first. In this stage, it will learn about feature conversion formats that are not fixed and depend on feature values. It just replaces the categorical values with their numeric values in this stage.

3.2.2 Data normalization

After transferring all symbolic features into numerical values, normalizing the data is an essential step. It will use the transferring and normalization process on test data as well. Normalization involves scaling each feature's value into a proportionate range, which removes any bias favoring features with greater importance. The maximum value for each feature is normalized and falls within a specific range.

3.2.3 Feature selection

Few features do not contribute to an approach, but this does not mean they are not valuable. To make good use of this information, it is critical to determine which features in vital data provide the most significant benefit. The advanced mutual information feature selection (AMIFS) algorithm is developed in the next section to solve the problem of feature selection. Classifier training can only rank features based on their relevance and cannot identify the best number of features required to achieve training. To support this task, this research extends to finding the best number of features needed. Before selecting the features, the technique ranks all features according to their importance to the classification processes. Then considering one feature at a time, the technique improves the classifier. Once the training dataset's highest classification accuracy is achieved, each method's final number of features is decided.

Additionally, we design distinct classes to mimic the systems which have been evaluated on a variety of data types (described in Sections 5). The algorithm for feature selection is applied to the classes as per the proposed model. It lists the total number and indexes of features with respect to the feature selection algorithm used. It shows the selected features in the experimental section.

3.3 Building classification

In this phase, different classifiers are trained using several evaluation criteria such as confusion matrix, accuracy, etc. Classifiers are used to identify one type of record in the dataset, known as the class of the records. With the help of the classifiers, the classification model is built from which all different classes are distinguished.

3.4 Classification analysis

In this part, we considered several classifiers to solve the classification problem. There are two methods to resolve issues with more than two classes: "One-Vs-One (OVO)" and "One-Vs-

All" (OVA). The OVO approach first divides an M-class problem into binary problems. Each binary classifier handles one problem: it is responsible for determining the difference between the data sets of two classes. The OVA method does the opposite; it takes an M-class problem and partitions it into M binary problems. In this case, a binary classifier classifies a single class of data from the rest. It is also clear that with the OVO approach, a larger number of binary classifiers will be required. In this way, it is more difficult to compute. In a study done by Rifkin and Klautau [36], it was found that the OVA technique was preferred over the OVO technique. The OVA technique is used to identify between normal and abnormal data by implementing the LS-SVM method on the proposed model. Next, the classifier is trained with the subset of features that contains the most correlated and essential features. The test data is sent to the previously trained model, which is used to identify the features. If the classifier model confirms that the record is abnormal, it removes those data from the data set during training.

4 Feature selection approaches

We used the Linear Correlation Coefficient (LCC) to measure the mutual dependence of two random variables. In real-world communication, the correlation is nonlinear among variables. When it comes to dependent variables that are not linearly dependent, a linear measure cannot show the relation between them. In order to truly understand the nature of interdependence, we will require a quantitative tool capable of revealing interdependence regardless of whether the variables are linearly or nonlinearly dependent. This paper aims to find an efficient way to extract the most valuable features from a feature space regardless of how correlated they are. We considered feature selection based on Mutual Information and linear correlation coefficient. One of the promising variables in the variable dependence estimation realm is mutual information. In particular, it handles variables that are linearly dependent, as well as nonlinearly dependent. Thus, it is our proposed feature selection algorithm's for correlating them.

4.1 Mutual information (MI) based feature selection

The MI is a symmetric index that reflects the correlation between two random variables. It provides non-negative value and zero value that indicates statistically independent observations. Let us consider two continuous variables $X = \{x_1,x_2,\ldots,x_n\}$ and $Y = \{y_1,y_2,\ldots.y_n\}$ where n is the number of samples. Mutual information can be determined by using the following formula:

$$I(X;Y) = E(X) + E(Y) - E(X,Y) \tag{1}$$

Where E (X) and E (Y) are entropies of X and Y which are defined as

$$E(X) = -\int_x p(x) \log p(x) dx, \tag{2}$$

$$E(Y) = -\int_y p(y) \log p(y) dy, \tag{3}$$

respectively and the joint entropy $E(X, Y)$ is defined as

$$E\ (X,Y) = -\int_x\int_y p(x,y)log\ p(x,y)dxdy \tag{4}$$

The above equations measure the amount of entropy on variables X provided by Y (or reverse) which can quantify the mutual information by using joint probability density function (pdf) as (5) as:

$$I(X;Y) = \int_x\int_y p(x,y)log\frac{p(x,y)}{p(x)p(y)}dxdy \tag{5}$$

Here, p(x,y) is considered as joint probability density function (pdf) and $p(x) = \int p(x,y)\ dx$ and $p(y) = \int p(x,y)\ dy$ are the marginal density functions. If mutual information is derived with mass function and marginal probabilities, the integration notation is replaced by summation notation, as in (6).

$$I(X;Y)\ \sum_{x\in X}\sum_{y\in Y}p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{6}$$

Here, two discrete random variables X and Y are considered with joint probability mass functions p(x, y) and marginal probabilities p(x) and p(y).

Features are relevant or essential if they comprise important information about the class; otherwise, they are nonessential or redundant. A common feature among many classes is that mutual information measures the amount of information that is shared between two random variables. This can be used as a criterion to judge the significance of a feature with respect to a class label. Features with a large mutual information have increased predictive power in this context I(C; f) where C stands for class and f stands for features. If I(C; f) is equal to zero, then, C and f are proven to be independent. Feature f contributes to the classification's redundancy. However, the value of the MI between variables is used as a selection criterion, and any computational errors could produce significant feature degradation. Due to this, it depends on finding the entropies and/or pdfs from the input data instances. It is possible to apply several estimation techniques in order to compute MI. Most researchers use histogram and kernel density estimations to estimate the pdfs [34, 36]. For instance, Peng et al. [34] stated that the histogram approach was fast and accurate but produced many errors. Those researchers presented that kernel density estimation has a high level of estimation accuracy while also having a heavy computational load.

The most significant challenge with histogram techniques is that they work with low-dimensional data, which may limit their application [16]. Histogram and kernel density approaches have been criticized by Rossi et al. [37] for their well-known challenges in dealing with high dimensional data. These two estimations aren't relevant in this case, as this study is working with high-dimensional data. The estimator proposed by Kraskov et al. [27] is applied to address the above issues. Unlike histogram and kernel density estimations, this technique uses the average distance from each data to its k-nearest neighbors to estimate the entropies of the given data. This estimator's novelty is that it can assess MI between two random variables from any data space. In essence, the key idea is to calculate

the entropy without knowing the densities, p(x, y), p(x), and p(y). For more information on how to estimate MI, see [27].

4.2 Advance feature selection algorithms

Two feature selection algorithms have been proposed in this paper based on the principles of [3, 34]. One of the earliest features-based classification evaluation methods used by Battiti is MIFS [3]. it calculates I(C;f_i) and I (f_s,f_i), where f_s and f_i are features and C is a class label. MI is corrected by subtracting a quantity proportional to the MI previously selected. Researchers [2, 28] have conducted multiple studies to advance Battiti's MIFS. As seen in step 4 of Battiti's MIFS, these enhancements were made on the augmentation of the second criterion term, although this approach has limitations. Instead, MIFS [2], MMIFS [22] and MIFS-U [28] do not offer specific suggestions on choosing a value for the parameter β. The selection criterion between the first and second terms is still imbalanced.

In this paper, advanced mutual information-based feature selection has been considered. This section proposes removing the burden of setting an appropriate value for β for all three Battiti's MIFS, Kwak's MIFS-U, and Amiri's MIFS. Equation (7) presents a new feature selection process, intended to maximize I (C; fi) and minimize the average redundancy (MR) results simultaneously. According to this new feature selection approach, a modification to the feature selection criterion would enhance the algorithms.

$$P_{MI} = arg \max_{f_i \in F} \left(I(C; f_i) - \frac{1}{|S|} \sum_{f_s \in S} MR \right) \tag{7}$$

Here, the quantity of feature information by class C is determined by I (C; f_i) and MR is also defined by

$$MR = \frac{I(f_i; f_s)}{I(C; f_i)} \tag{8}$$

Here, $f_i \in F$ and $f_s \in S$. In a particular case, f_i is rejected without computing (7) if I (C; f_i) = 0. For strong dependent of f_i and f_s in I (f_i; f_s), feature fi will make redundancy concerning I (C; fi). Thus, a threshold value for P_{MI} is considered in (7) with the following properties.

(a) When P_{MI} = 0, the current feature f_i has no additional information that the classifier can use. Since f_i is no longer part of S, it is removed.
(b) When P_{MI} > 0, the feature f_i is relevant or important to the classification after choosing the subset S of features. Thus, S is currently augmented by adding the current candidate f_i.
(c) When P_{MI} < 0, It follows that feature f_i is redundant to the output C because it could reduce the MI between the subset S and the output C. When measuring the amount of redundancy between feature f_i and the output class, the term within Eq. (7) that accounts for redundancy is larger than the term that refers to relevance, and as a result is worth nothing**.** Feature f_i has thus been removed from S.

The selection process of features from dataset using MI is considered through algorithm 1.

Algorithm 1. Advanced mutual information-based feature selection.

Input: Feature set F = {f_1,f_2,…f_n},
Output: S – Selected feature subset, S = {s_1,s_2,…s_m}, s_f is selected final feature

1. Initialization: Set S = Φ
2. For i = 1 to n
   Compute I (C;f_i) // C for class, f_i for the feature set, and I for mutual information
3. For s_f = n; select the feature f_i as
   $arg\ max_{f_i}(I(C; f_i))$, i = 1,2, ….$s_f$ // maximum mutual information score between C and f_i
4. Set F ← F \ {f_i}; S ← S ∪ {f_i}; s_f = s_f -1. /* after getting MI score f_i will be removed from F and add in S and test from last feature to 1st feature up to feature set is empty */
5. If F ≠ Φ then
6. Compute I(f_i, f_s) // where s ∈ S
7. Compute P_{MI} in (7) to find f_i where i ∈ {1,2,… s_f };
8. Decreasing s_f by 1
9. F ← F \ {f_i};
10. If (P_{MI} > 0) then
11. S ← S ∪ {f_i}
12. End
13. End
14. Use merge Sort for S according to the value of P_{MI} of each selected feature.
15. Return S

4.3 Correlation coefficient-based feature selection

To determine the flexibility and effectiveness of FMIFS, it considers the auxiliary MI by Linear Correlation Coefficient (LCC) in Algorithm 2. Dependence measures, such as LCC [35], is the most popular approach to assessing the relationship between two random variables. The correlation coefficient for two random variables of the same type is defined when their X and Y values are the same (9). In this way, LCC measures the correlation between random linearly dependent variables very quickly and accurately, but it is unable to detect nonlinear correlations.

$$Corr\ (X;Y) = \frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2\sum_{i=1}^{n}(y_i-\overline{y})^2}} \tag{9}$$

The value of $Corr\ (X;\ Y)$ falls within a specific closed interval that ranges from −1 to 1. In many cases, values equal to −1 or very close to 1 indicate a strong relationship between the two variables. A value close to zero implies a weak relationship and the variables are inversely related. The algorithm 2 is referred to as Flexible Linear Correlation Coefficient based Feature Selection (FLCCFS). The content in Algorithm 2 was designed to identify a feature that maximizes P_{Corr} in (10) and to remove features that are either unnecessary or redundant.

$$P_{Corr} = arg\max_{f_i \in F}\left(Corr(C;f_i) - \frac{1}{|S|}\sum_{f_s \in S}\frac{Corr(f_i;f_s)}{Corr(C;f_i)}\right) \tag{10}$$

Algorithm 2. Advanced Correlation Coefficient Based Feature Selection.

Input: Feature set F = {f_1,f_2,…f_n},
Output: S – Selected feature subset, S = {s_1,s_2,…s_m}, s_f is selected final feature

1. Initialization: Set S = Φ
2. For i = 1 to n
3. Compute Corr (C; f_i)
4. For s_f = n; select the feature f_i as
16. $\arg\max_{f_i}(Corr(C; f_i))$, i = 1,2, ….$s_f$, maximum correlation coefficient score between C and f_i
17. Set F ← F \ {f_i}; S ← S ∪ {f_i}; s_f = s_f -1. /* after getting MI score f_i will be removed from F and add in S and test from last feature to 1st feature up to feature set is empty */
5. If F ≠ Φ then
6. Compute Corr (f_i, f_s)
7. Compute P_{Corr} in (10) to find f_i where i Є {1,2,… s_f };
8. Decreasing s_f by 1;
9. F ← F \ {f_i};
10. If (P_{Corr} > 0) then
11. S ← S ∪ {f_i}
12. End
13. End
14. Use merge Sort for S according to the value of P_{Coff} of each selected feature.
15. Return S

For feature set selection, the selective feature set is always subset of original set but not reverse. This is proved by different theorems as follows.

Theorem 1 Let F is the feature set, and S is the selected feature set as mutual information score. Then prove that for each i, feature $f_i \in F$ and $f_i \in S$, then $S \subset F$ but $F \subset S$ is not possible.

Proof Let original feature set F and selected feature set S (as mutual information score) are two sets considered for the proposed model. Any element is chosen from one set and also available in the second set, that element is common for both the sets. For certain i, $f_i \in F$, and $f_i \in S$. So, we can get $S \subset F$. Here f_i is common for both sets. As per mutual information score, the score may be MI = 0, MI > 0, MI < 0. This score may be happened in few data set. We don't consider MI < 0, and MI = 0 for feature selection due to lack of information or redundancy of data. When we considered MI > 0, then

(a) few features may not be selected due to previous two conditions. Thus, original set must be greater than selected feature set. That is, $S \subset F$.
(b) if all features are selected, there is no meaning of feature selection as per our proposed model. Thus, for selection point of view, S will never be equal to F.

Thus $F \subset S$ is not possible.

Theorem 2 Let F is the feature set, and S is the selected feature set per the correlation coefficient score. Prove that for each i, feature $f_i \in F$ and $f_i \in S$, then $S \subset F$ but $F \subset S$ is not possible.

Proof This can be proved as theorem 1.

Theorem 3 Let F is feature set, and S is the selected feature set per mutual information or correlation coefficient score. Then prove that |F| > |S|, where | | represents the number of elements in a particular set or the cardinality of the set.

Proof We can prove it using contradictory approaches. Assume, $|F| < |S|$. Then, number of elements in selected features set S is greater than F. As per theorem 1 and 2, we know that $S \subset F$ but not reverse. That means, number of elements of S has to be less compared to F. If one or more element is added to S, then we may get $|S| = |F|$ which is not true, because if $|S| = |F|$, there is no meaning of feature selection. Thus, it contradicts our assumption. Since it doesn't satisfy either $|F| < |S|$ or $|S| = |F|$, thus number of elements of F is always greater than S using any kind of approaches for selected features. Thus, $|F| > |S|$. □.

4.4 Analysis of computational complexity

The time complexity of algorithms 1 and 2 is O (n^2). We have shown the complexity analysis of algorithm 1. For algorithm 2, it can be done similarly. The time complexity of different statements of algorithm 1 is as follows: line 1 is of O (1). In line 4, it is a 1 x n and inside a for loop as per line 2, so line 4 gets executed $n \times 1$ x n = n^2 a number of times, and its complexity is O(n^2). The complexity of ine 6 is O (1). Line 7 uses merge sort, and its average time complexity is O (nlogn). Line 8 gets executed n + n = 2n times, and thus, it is O (n). The complexity of line 9 is O (1) and for line 10, 11 and 12 it is O (n^2). Complexity of line number 13, 15, 17, 18, and 20 is O (1). Line 14 and 16 is of O (n). Line 19 uses merge sort, and its average time complexity is O (nlogn). Thus, total time complexity is (8 x O (1) + 2 x O (n) + 2 x O (nlogn) + 4 x O (n^2)) = O (n^2). Thus the time complexity of algorithm 1 is O (n^2). Similarly, the time complexity of algorithm 2 can be found out.

5 Experimental settings

In this section, we are considering several experimental settings based on needed approaches as per the proposed model. The implementation requires analyzing data sets, software tools, and hardware devices for experimental evaluation.

5.1 Datasets

Each dataset possesses unique data size and varying numbers of features. For machine learning and data mining experiments, the datasets are generally collected from the UCI machine learning repository and Kaggle [24, 25], and they provide extensive testing for feature selection methods. We have selected three datasets from Kaggle for our experiment [25], because we want to ensure a fair and rational comparison with other state-of-the-art selection approaches. The datasets are the Mobile price prediction data set, Heart failure clinical record, and Diabetes datasets.

(a) Mobile dataset: We have considered 2000 records with 21 features from the Mobile data set [25]. This data set is used for selling a product. We considered this data set to find selected features to increase the mobile selling or produce more demand. This data set contains a few features with 0 or 1 values which are not so effective for feature selection, so we don't consider a such feature for experiments.

(b) Heart dataset: This data set contains 299 records and 13 features. This data set includes sensitive information because it is health-related data. The data set helps to find appropriate features for heart-related diseases such as heart attack, heart failure, etc.
(c) Diabetes dataset: This data set contains 2000 records and nine features. This data is related to health care. This data set helps to find valuable information related to diabetes.

5.2 Experimental environment

In this part, we implement our proposed model with the help of several software tools, languages, and packages such as Python 3.7, Numpy, Pandas, Matplotlib.pyplot, Seaborn, Sklearn, Google Colab etc. The models for identifying and classifying input data are implemented with Tensorflow. All tests are conducted on a Personal Computer (PC) with software and hardware specifications such as (a) An Intel Core i7 CPU with 16 GB of RAM and (b) Python 3.0.7 on an Ubuntu 16.04 OS for implementation of the diagnosis system.

5.3 Performance evaluation

Several experiments have been conducted to assess the classifiers' performance and effectiveness. The exactness rate, the detection rate, the false positive rate, and the F-measures are applied for this purpose. The exactness, detection rate and false positive rates are defined by

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{11}$$

$$Detection\ Rate = \frac{TP}{TP + FN} \tag{12}$$

$$False\ Positive\ Rate = \frac{FP}{FP + TN} \tag{13}$$

Where True Positive (TP) are the number of actual features in classified records, True Negative (TN), the actual number features in normal classified records classified, False Positive (FP), and False Negative (FN) are the actual number false features in classified records classified as well as standard classified records.

The F-measure is the harmonic mean computed using p (Precision) and recall r [17]. The f-measure used in this paper assigns the same weight to both the accuracy rate (PR) and the callback rate (RR) (14).

$$F{-}Measure = \frac{2(Precision{*}Recall)}{Precision + Recall} \tag{14}$$

Precision (p) is the proportion of positive predicted values. The precise value directly affects the system's performance. A higher precision value means a lower false positive rate and vice versa. The formula for precision is given by (15).

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

Recall (r) is another essential value to measure the system's performance and correctly indicate the proportion of the actual positive numbers identified. The recall is defined as:

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

5.4 Hypothesis testing

We considered two measurements on our data sets such as (a) F-test and (b) p value. F-test is used to evaluate a particular data set's duplicate and non-duplicate data. It evaluates all features of the training data set. p value is considered to avoid the null hypothesis. We considered p value with α test. $\alpha = 0.05$ is considered for significant level. If the experimental result is considered for p value with $\alpha < 0.05$, then null hypothesis will be rejected and features with strong relationship with the class will be selected. Thus, in our experiment, we only considered the p value for $\alpha < 0.05$.

6 Evaluation and result analysis

Our proposed model uses mutual information and linear correlation coefficient under filter methods. All methods have been experimented through several classifiers for getting different outputs. For better performance, we considered several comparison evaluations based on different classifiers, variance threshold value, training and test data set, accuracy score etc. Initially, it considered splitting different data sets into training and test data set as per classifiers and feature selection approaches as Table 1. The following tables briefly describe several evaluations as per methods and classifiers.

6.1 Filter method

In this method, initially, we consider 2000 records with 21 features of mobile data set, and later the data set is split into 40% test data set and 60% training data set i.e., 1200 training data points and 800 test data points. When the original data is split, the quantity of the original data will be ((1200, 20), (800, 20)) by train_filter.shape and test_filter.shape. That means (1200, 20)

Table 1 Collected data set divides into training and testing dataset

S.N.	Feature selection Approaches	Data set	Training data	Test Data
1	Variance Threshold, Regression, SelectKBest	Mobile	60%	40%
2	Variance Threshold, Regression, SelectKBest	Heart	60%	40%
3	Variance Threshold, Regression, SelectKBest	Diabetes	60%	40%
4	Variance Threshold, Mutual Information, SelectKBest	Mobile	80%	20%
5	Variance Threshold, Mutual Information, SelectKBest	Heart	80%	20%
6	Variance Threshold, Mutual Information, SelectKBest	Diabetes	60%	40%

data points are used as training data and (800, 20) data points are used as test data with 20 features for both data sets. We tried to remove the duplicate feature from the 20 features, but no duplicate features are found. Thus, we apply F-Test on non-duplicate trained data and trained data set. Thus, we get both non-duplicate trained and normal trained data as shown in Table 2.

6.2 Use of P-values for experiments

We use p-values (Probability values) for our experiments on all datasets in this paper. Although different authors develop p values, reader can refer to [18, 21] for more clarity. Here, it applies p values on 20 features of the mobile dataset and got p-values with fig size (13, 5) where 5 values are less than 0.05 and 13 values are more significant than 0.05, as shown in Fig. 2. Here, it considers the threshold value (θ) for p-value is 0.05, then p values tested for all mobile price data set features.

We considered p-values <0.05, the feature numbers those are selected are {13, 0, 11, 12, 8, 14, 6}. It is shown in Fig. 2. When we evaluate non-duplicate trained data and trained data set, its accuracy varies for different classifiers as shown in Table 3. In this experiment, we considered 13 classifiers to evaluate the dataset. When a data set is evaluated through classifiers, an individual confusion matrix will be generated where the matrix helps evaluate the eqs. (11–16). Since the data of the confusion matrix are different for different classifiers, the evaluation accuracy will be different as per the classifiers shown in Table 3. We have evaluated all the equations, but the eqs. (11–16) are considered for evaluation performance.

Further, we have taken different classifiers to perform well based on accuracy. Thus, it only mentioned the accuracy of all classifiers of Mobile data set, as in Table 3. The confusion matrix for all data set is given in Table 8.

Further, the filter method is also considered for the heart failure dataset. In this method, we considered 299 records with 13 features and later the data set was split into 60% of training

Table 2 F-test values on Mobile dataset

S.No.	Non-duplicate trained data	Trained data set
1	2.12084124e+01,	2.17946030e-13,
2	1.65466149e+00,	1.75063867e-01,
3	3.75555667e-01,	7.70655179e-01,
4	3.58453310e-01,	7.83033587e-01,
5	2.12280507e-01,	8.87926582e-01,
6	1.62902758e+00,	1.80862329e-01,
7	2.85484468e+00,	3.61147924e-02,
8	2.19018882e+00,	8.75061337e-02,
9	5.19954044e+00,	1.43271628e-03,
10	2.17727196e+00,	8.90036639e-02,
11	7.34719339e-01,	5.31347944e-01,
12	1.55405826e+01,	6.39402088e-10,
13	1.44870472e+01,	2.83874984e-09,
14	2.04718624e+03,	0.00000000e+00,
15	3.08494494e+00,	2.64675743e-02,
16	1.27592982e+00,	2.81190579e-01,
17	6.60927714e-01,	5.76157483e-01,
18	8.66256726e-01,	4.58009493e-01,
19	7.22246341e-01,	5.38739291e-01,
20	1.45133261e-01	9.32814703e-01

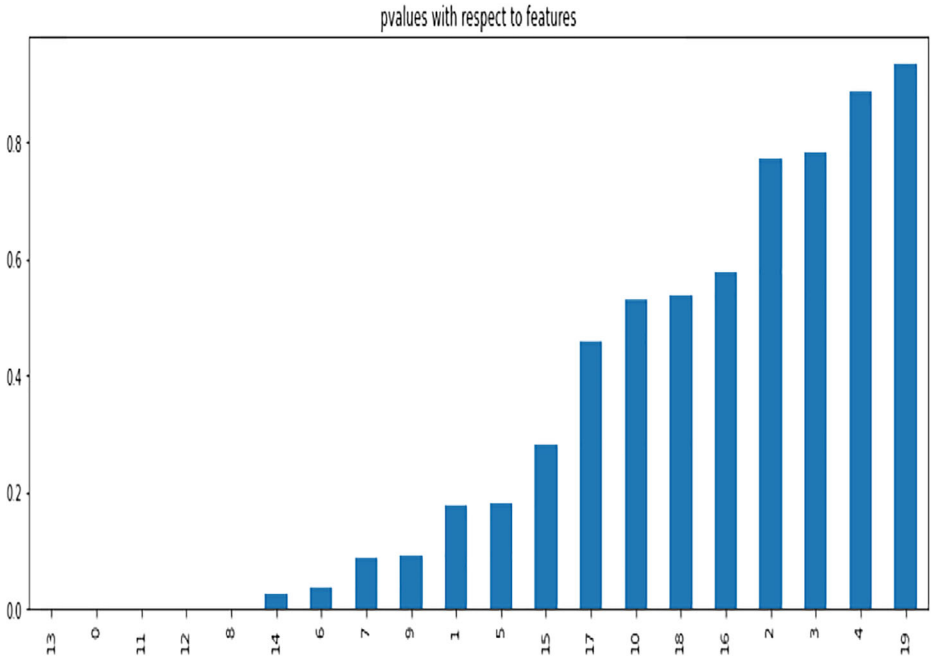

Fig. 2 P-values mobile dataset

data set and 40% test data set i.e., 179 training data points and 120 test data points were used for experiments. When the original data is split, shape of the original data will be ((179, 12) and (120, 12)) by train_filter.shape and test_filter.shape. That means (179, 12) data points are used as training data and (120, 12) data points are used as test data with 12 features for both the data sets. From these two data sets, we tried to remove duplicate feature from 12 features, but no duplicate features are found. Thus, we apply F-Test on non-duplicate trained data and normal trained data set. Then we get the F-test result as shown in Table 4,

Table 3 Filter method -mobile dataset accuracies

S.No.	Classifier name	Accuracy
0	Nearest_Neighbors	0.92225
1	Linear_SVM	0.97250
2	Polynomial_SVM	0.94000
3	RBF_SVM	0.25000
4	Guassian_Process	0.25000
5	Gradient_Boosting	0.93000
6	Decision_tree	0.82000
7	Extra_Trees	0.88000
8	Random_Forest	0.88000
9	Neural_Net	0.62250
10	AdaBoost	0.86500
11	Naïve_Bayes	0.81750
12	QDA	0.96750
13	SGD	0.59750

Table 4 F-test values heart failure dataset

S.No.	Non-duplicate trained data	Trained data set
1	7.93316652e+00,	5.40493334e-03,
2	2.48440592e+00,	1.16765383e-01,
3	1.32437049e+00,	2.51361488e-01,
4	3.24911915e-01,	5.69393405e-01,
5	2.33892111e+01,	2.86268328e-06,
6	5.19140909e-03,	9.42642327e-01,
7	9.43976035e-01,	3.32583250e-01,
8	1.58706123e+01,	9.89626702e-05,
9	9.75182095e+00,	2.09315129e-03,
10	7.54703603e-01,	3.86167540e-01,
11	1.75623082e+00,	1.86803456e-01,
12	6.63172187e+01	6.64810374e-14

Further, we apply P-values on this data set with 12 features with figsize (7, 5) where 5 values are less than 0.05 and 7 values are greater than 0.05 as shown in Fig. 3.

When we considered p-values <0.05, the feature numbers that are selected are {11, 4, 7, 8, 0}. When we evaluated eq. 11 for accuracy on the heart failure data set with different classifiers, its accuracies varied as per different classifiers, as shown in the Table 5.

Further, filter method is also considered for Diabetes dataset. In this method, initially, we considered 2000 records with nine features. Later the data set was split into 60% of training data and 40% of test data i.e., 1200 records were used as training data, and 800 records were used as test data. When the original data set is split, the shape of the original data is ((1200, 8), (800, 8)) by train_filter.shape and test_filter.shape. That means (1200, 8) data points are used

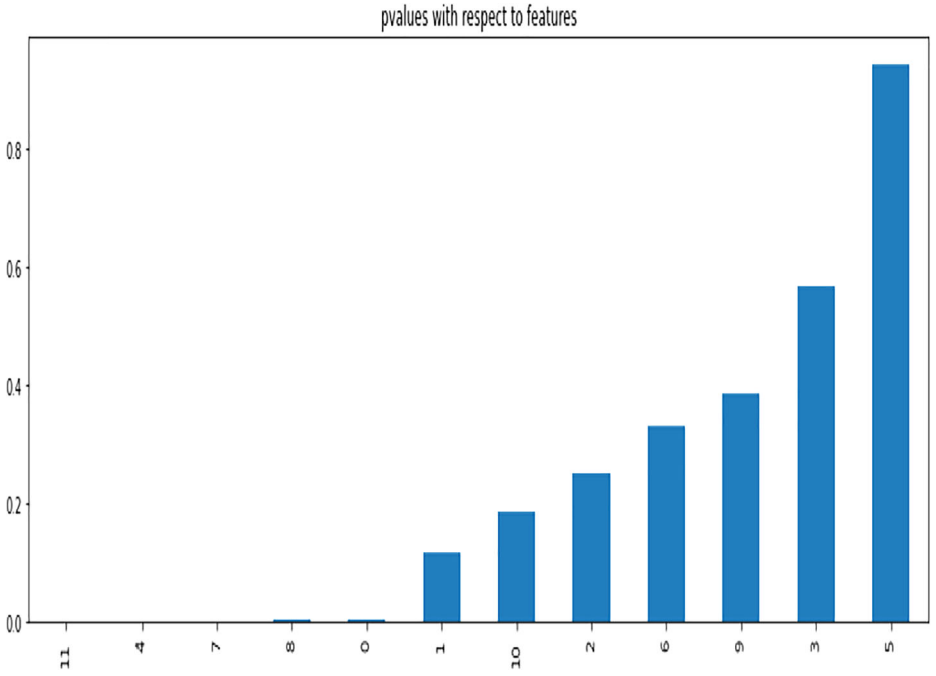

Fig. 3 P-values for heart dataset

Table 5 Filter method for heart dataset accuracies

S.No.	Classifier name	Accuracy
0	Nearest_Neighbors	0.88333
1	Linear_SVM	0.86667
2	Polynomial_SVM	0.86667
3	RBF_SVM	0.68333
4	Guassian_Process	0.88333
5	Gradient_Boosting	0.83333
6	Decision_tree	0.85000
7	Extra_Trees	0.83333
8	Random_Forest	0.83333
9	Neural_Net	0.86667
10	AdaBoost	0.88333
11	Naïve_Bayes	0.85000
12	QDA	0.81667
13	SGD	0.58333

for training data and (800, 8) data points are used for test data with eight features for both data set. We don't consider one feature for the test due to feature contains ID number, which is unique. We tried to remove duplicate features from these two data sets, but no duplicate features were found. Thus, we applied F-Test on non-duplicate trained data and trained data set, which is given in Table 6.

Again, we applied p-values on eight features with figsize (6,2) where 6 values are less than 0.05 and 2 values are greater than 0.05 as shown in Fig. 4.

When we considered p-values <0.05, the number of features is selected as {1, 5, 0, 7, 6, 4}. When we evaluate eq. 11 on Diabetes data set, its accuracies for different classifiers vary, as given in Table 7.

All confusion matrix data is mentioned in Table 8. The Mobile data set considered four kinds of possible confusion matrix that provide single accuracy. If we change the confusion matrix for a particular classifier, it couldn't affect the accuracy of the same classifier. For the sake of the information, we have tested different confusion matrices, but we got the same accuracy for the same classifier. Thus, we don't consider a kind of confusion matrix for other data sets such as Diabetes dataset and Heart_failure dataset, is shown in Table 8.

When we implement algorithm 1 to find mutual information among feature sets and classification of different data sets, we get the outcome of mutual information. But, the last part of algorithm 1 is considered for sorting based on the feature set as per the mutual

Table 6 F-test values for diabetes dataset

S.No.	Non-duplicate trained data	Trained data set
1	68.64561623	3.12431569e-16
2	282.73924149	3.97748131e-57
3	3.14232032	7.65392871e-02
4	3.99864857	4.57622421e-02
5	12.42407141	4.39821776e-04
6	106.42597017	5.78383192e-24
7	32.20559406	1.73814063e-08
8	60.40928604	1.64962253e-14

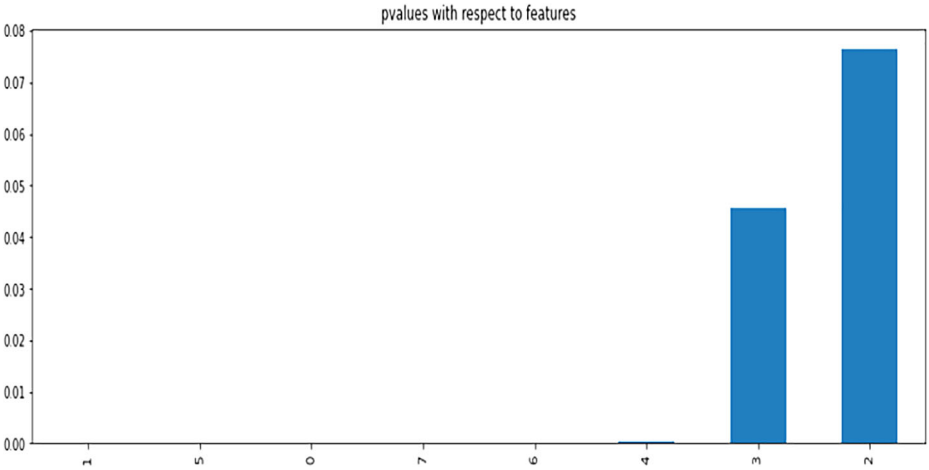

Fig. 4 p-values Diabetes dataset

information value. Thus, we get mutual information results on three different data sets, as mentioned in Tables 9, 10 and 11.

Further, we implemented algorithm 2 to find the correlation coefficient among features of corresponding data sets. The evaluation result for the Mobile data set, the Diabetes Data set, and the Heart Failure data set using correlation coefficient is given in Tables 14, 15, and 16, respectively. Since several features are available in the Mobile dataset and its evaluation result occupy a large space than paper size, the evaluation results of those data set are kept in Appendix A. Particularly, Table 14 is also divide its evaluation results in two parts and kept in the same table. Readers don't confuse by reading on this table. To avoid large space, all evaluation results considered floating round numbers with three digits after the decimal number.

Although, we have considered different approaches for feature selection, but features are selected using the MI and p-values as shown in Table 12. Through these approaches, selected features are orderly selected, but we don't consider the selected features whose MI is zero and p values <0.05 in Table 12. Thus, the common features are selected which are satisfied both

Table 7 Filter method for diabetes dataset accuracies

S.No.	Classifier name	Accuracy
0	Nearest_Neighbors	0.86250
1	Linear_SVM	0.79500
2	Polynomial_SVM	0.78750
3	RBF_SVM	0.97750
4	Guassian_Process	0.98500
5	Gradient_Boosting	0.97750
6	Decision_tree	0.82250
7	Extra_Trees	0.99000
8	Random_Forest	0.84500
9	Neural_Net	0.74750
10	AdaBoost	0.83000
11	Naïve_Bayes	0.77000
12	QDA	0.77500
13	SGD	0.69500

Table 8 Classifiers confusion matrix, accuracy of different data set

S.No.	Classifier name	Mobile dataset	Diabetes dataset	Heart_failure dataset
0	Nearest_Neighbors	[97, 3, 0, 0], [6, 90, 4, 0], [0, 7, 91, 2], [0, 0, 9, 91]	[234, 29], [26, 111]	[1, 40], [6, 13]
1	Linear_SVM	[99, 1, 0, 0], [2, 95, 3, 0], [0, 2, 97, 1], [0, 0, 2, 98]	[230, 33], [49, 88]	[2, 39], [6, 13]
2	Polynomial_SVM	[100, 0, 0, 0], [12, 87, 1, 0], [0, 6, 92, 2], [0, 0,3, 7]	[250, 13], [72, 65]	[41, 0], [8, 11]
3	RBF_SVM	[0,0, 0,100], [0, 0,0, 100], [0,0, 0,100], [0, 0,0, 100]	[263, 0], [9, 128]	[41, 0], [19, 0]
4	Guassian_Process	[0,0,0, 100], [0, 0,0, 100], [0, 0,0, 100], [0,0, 0, 100]	[257, 6], [0, 137]	[1, 40], [6, 13]
5	Gradient_Boosting	[96, 4, 0, 0], [4,94, 2, 0], [0,4,93, 3], [0, 0,11,89]	[256, 7], [2, 135]	[6, 35], [4, 15]
6	Decision_tree	[87,13, 0, 0], [4,84,12,0], [0,15,72,13], [0, 0,17, 83]	[242, 21], [50, 87]	[4, 37], [5, 14]
7	Extra_Trees	[95, 5, 0, 0], [5,90,5,0], [0,12,83,5], [0,0,12,88]	[263, 0], [4, 133]	[2, 39], [5, 14]
8	Random_Forest	[91,9,0,0], [6,91,3,0], [0,15,78,7], [0,0, 10, 90]	[242, 21], [37, 100]	[2, 39] [6, 13]
9	Neural_Net	[54, 44,2,0], [6,74,10,10], [0,41,20,39], [0, 9,9,82]	[229, 34], [64, 73]	[1, 40], [5, 14]
10	AdaBoost	[87,13,0, 0], [4,90,6,0], [0,12,76,12], [0,0,8,92]	[229, 34], [34, 103]	[4, 37], [3, 16]
11	Naïve_Bayes	[87,13, 0, 0], [8,80,12, 0], [0,13,70,17], [0, 0,10,90]	[219, 44], [48, 89]	[1, 40], [8, 11]
12	QDA	[99,1,0,0], [3,94,3,0], [0,3,96,1], [0,0,2,98]	[227, 36], [54, 83]	[1, 40], [9, 10]
13	SGD	[78,21,1, 0], [23,53,12,12], [0,23,20,57], [0,4,4,92]	[263, 0], [137, 0]	[41, 0], [7, 12]

Table 9 Mutual information result on mobile dataset

Ram	0.846238
dual_sim	0.031856
px_height	0.028298
px_width	0.027900
battery_power	0.027715
m_dep	0.025830
three_g	0.019987
mobile_wt	0.011112
int_memory	0.010372
four_g	0.001622
talk_time	0.001463
Wifi	0.000134
touch_screen	0.000000
PC	0.000000
sc_h	0.000000
FC	0.000000
sc_w	0.000000
clock_speed	0.000000
blue	0.000000
n_cores	0.000000

MI > 0 and p values >0.05 as mentioned in Table 12 which is not so effective. But when we considered p-values <0.05, which is more effective than (> 0.05) as per hypothesis test. Thus Table 13 is more effective manner constructed as per MI and p value measurements.

Table 10 Mutual information result on diabetes dataset

DiabetesPedigreeFunction	0.205154
Glucose	0.188032
BMI	0.157945
Insulin	0.108301
Age	0.079297
SkinThickness	0.040284
Pregnancies	0.036353
BloodPressure	0.030140

Table 11 Mutual information result for heart failure dataset

Time	0.244483
serum_creatinine	0.087882
ejection_fraction	0.082135
Age	0.057337
serum_sodium	0.034966
creatinine_phosphokinase	0.029436
Platelets	0.004868
Smoking	0.000000
Sex	0.000000
high_blood_pressure	0.000000
Diabetes	0.000000
Anaemia	0.000000

Table 12 Comparative selected features through MI and p values (> 0.05) of three data set

	Original Data set	Selected features as MI	Selected features as p values(>0.05)	Number of selected features in common from both MI and P values
Mobile data set	[‘battery_power’, ‘blue’, ‘clock_speed’, ‘dual_sim’, ‘fc’, ‘four_g’, ‘int_memory’, ‘m_dep’, ‘mobile_wt’, ‘n_cores’, ‘pc’, ‘px_height’, ‘px_width’, ‘ram’, ‘sc_h’, ‘sc_w’, ‘talk_time’, three_g’, ‘touch_screen’, ‘wifi’] Total=20	[‘ram’, ‘dual_sim’, ‘px_height’, ‘px_width’, ‘battery_power’, ‘m_dep’, ‘three_g’, ‘mobile_wt’, ‘int_memory’, ‘four_g’, ‘talk_time’, ‘wifi’] Total=12	[‘m_dep’,, ‘n_cores’, ‘blue’, ‘four_g’, ‘sc_w’, ‘three_g’, ‘pc’, ‘touch_screen’, ‘talk_time’, ‘clock_speed’, ‘dual_sim’, ‘fc’, wifi’] Total=13	[‘dual_sim’, ‘m_dep’, ‘four_g’, ‘talk_time’, ‘three_g’, ‘wifi’] Total=6
Diabetes dataset	[Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age, Outcome]	Diabetes Pedigree Function, Glucose, BMI, Insulin, Age, Skin Thickness, Pregnancies, Blood Pressure, Total=8	Skin Thickness, Blood Pressure, Total=2	Skin Thickness, Blood Pressure, Total=2
Heart_failure dataset	[‘age’, ‘anaemia’, ‘creatinine_phosphokinase’, ‘diabetes’, ‘ejection_fraction’, ‘high_blood_pressure’, ‘platelets’, ‘serum_creatinine’, ‘serum_sodium’, ‘sex’, ‘smoking’, ‘time’, ‘Death_Event’]	[time, serum_creatinine, ‘ejection_fraction’ age, serum_sodium, ‘creatinine_phosphokinase’, platelets, ‘smoking, ‘sex’, ‘high_blood_pressure’, ‘diabetes’, ‘anaemia’] Total=12	[anaemia’ ‘smoking, ‘creatinine_phosphokinase’, platelets, ‘sex’, ‘diabetes’ ‘high_blood_pressure’,] Total=7	[anaemia’ ‘smoking, ‘creatinine_phosphokinase’, platelets, ‘sex’, ‘diabetes’ ‘high_blood_pressure’,] Total=7

Table 13 Comparative selected features through MI and p values (< 0.05) of three data set

	Original Data set	Selected features as MI	Selected features as p values(<0.05)	Number of selected features in common from both MI and P values
Mobile data set	['battery_power', 'blue', 'clock_speed', 'dual_sim', 'fc', 'four_g', 'int_memory', 'm_dep', 'mobile_wt', 'n_cores', 'pc', 'px_height', 'px_width', 'ram', 'sc_h', 'sc_w', 'talk_time', three_g', 'touch_screen', 'wifi'] Total=20	['ram', 'dual_sim', 'px_height', 'px_width', 'battery_power', 'm_dep', 'three_g', 'mobile_wt', 'int_memory', 'four_g', 'talk_time', 'wifi'] Total=12	[ram, 'battery_power', 'px_height', 'px_width', 'mobile_wt', 'sc_h', 'int_memory'] Total=7	[ram, 'battery_power', 'px_height', 'px_width', 'mobile_wt', 'int_memory'] Total=6
Diabetes dataset	[Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age, Outcome]	Diabetes Pedigree Function, Glucose, BMI, Insulin, Age, Skin Thickness, Pregnancies, Blood Pressure, Total=8	[Glucose, BMI, Pregnancies, Age, Diabetes Pedigree Function, Insulin] Total=6	[Glucose, BMI, Pregnancies, Age, Diabetes Pedigree Function, Insulin] Total=6
Heart_failure dataset	['age', 'anaemia', 'creatinine_phosphokinase', 'diabetes', 'ejection_fraction', 'high_blood_pressure', 'platelets', 'serum_creatinine', 'serum_sodium', 'sex', 'smoking', 'time', 'Death_Event']	[time, serum_creatinine, 'ejection_fraction' age, serum_sodium, 'creatinine_phosphokinase', platelets, 'smoking, 'sex', 'high_blood_pressure', 'diabetes', 'anaemia'] Total=12	[time, serum_creatinine, 'ejection_fraction' age, serum_sodium] Total=5	[time, serum_creatinine, 'ejection_fraction' age, serum_sodium] Total=5

7 Discussions

In this section, we have discussed various approaches, advantages and limitations of the applied scheme as follows. We considered filter-based feature selection based on Mutual Information and Linear Correlation Coefficient (LCC) approaches and developed two feature selection algorithms for analyzing of feature selection. As our feature selection model, we divide the data set into two sets: training and testing. Above approaches analyses the number of features with respect to the number of classes. We also developed two theorems for feature selection using Mutual Information and Linear Correlation Coefficient (LCC) methods and proved it with set theory. The computational complexity is also analyzed of two algorithms.

The experimental analysis uses experimental settings, data sets, and software and hardware tools. The performance evaluation is also taken to find out various measurements of evaluation items through True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) as per the actual and false classified feature. The experiment is developed by training and testing data from a different database. Thus, we got the evaluation performance as per the considered methods. We also used p-values to select the number of features from the other data set. F-test values are considered on the trained data set. We also developed various confusion matrices of different data sets and classifiers for comparison performance.

The advantages of the applied scheme are as follows: (a) This model helps to find feature selection with the help of p-values, (b) a different confusion matrix is generated for different classifiers. (c) F-tests are used for both normal and duplicate data sets and (d) two theorems are used and proved with set theory.

Limitations of the applied scheme are as follows: (a) Although we considered two traditional methods for feature selections, but used differently in our model, (b) classifiers are old, but designed through various confusion matrices for evaluation (c) confusion matrix is also old, but it considered the numerical data dissimilarly for different classifiers which is mentioned in Table 8. Since our proposed model can apply any data set for feature selection, we developed this model as a wide-range approach to effectively finding features from different datasets.

8 Conclusions

This paper considered three approaches for selecting features from different datasets: p-values, mutual information, and correlation coefficient. The approach of p values is vital for selecting features, excluding these approaches. Although two algorithms are used to select features, we considered MI and P-values approach for selecting features with the order, and comparison. We have also used merge sorting to order selected features in an algorithm that is very effective for selected features. To achieve accuracy, we evaluate the confusion matrix with different approaches for better performance. These confusion matrices are also developed for different tests by classifiers and data sets, but we have the same accuracy on various matrix forms. The proposed algorithm shows comparable results with other state-of-the-art approaches in testing for different data sets. It offers the best selection of data set experiments compared to other selection systems tested on the same data set. Finally, it assumed that the proposed selection system had achieved promising performance in selecting features from each data set based on the experimental results. The proposed AMIFS algorithm for selection successfully improved the search strategy. The effect of the feature and the small selection of features must also be carefully considered in future studies.

Appendix

Table 14 Evaluation results of correlation coefficient on diabetes dataset

	P	G	BP	ST	I	BMI	DPF	A
P	1	0.128	0.142	−0.064	−0.075	0.014	0.008	0.520
G	0.128	1	0.125	0.055	0.309	0.222	0.111	0.253
BP	0.142	0.125	1	0.193	0.077	0.261	0.055	0.244
ST	−0.064	0.055	0.193	1	0.449	0.387	0.167	−0.12
I	−0.075	0.309	0.077	0.449	1	0.222	0.181	−0.091
BMI	0.0138	0.222	0.261	0.387	0.222	1	0.108	0.031
DPF	0.008	0.111	0.055	0.167	0.180	0.108	1	0.043
A	0.520	0.253	0.244	−0.12	−0.090	0.031	0.04	1

Table 15 Evaluation results of correlation coefficient on Heart dataset

	A	ANM	CP	DB	EF	HBP
A	1	0.108	−0.06	−0.139	0.046	0.106
ANM	0.108	1	−0.204	−0.023	−0.005	0.041
CP	−0.06	−0.204	1	−0.029	−0.034	−0.071
DB	−0.139	−0.023	−0.029	1	0.018	−0.021
EF	0.046	−0.005	−0.034	0.018	1	0.028
HBP	0.106	0.041	−0.071	−0.021	0.028	1
P	−0.087	−0.029	0.037	0.116	0.077	0.031
SC	0.143	0.069	−0.014	−0.053	0.015	−0.025
SS	−0.003	0.029	0.043	−0.113	0.22	0.066
S	0.031	−0.076	0.113	−0.154	−0.149	−0.137
SMG	0.024	−0.075	0.012	−0.163	−0.048	−0.073
T	−0.225	−0.197	−0.028	−0.02	0.049	−0.168
	P	SC	SS	S	SMG	T
A	−0.087	0.143	−0.003	0.031	0.023	−0.224
ANM	−0.029	0.069	0.029	−0.076	−0.075	−0.197
CP	0.037	−0.014	0.043	0.113	0.012	−0.028
DB	0.116	−0.053	−0.113	−0.155	−0.163	−0.02
EF	0.077	0.015	0.220	−0.149	−0.048	0.049
HBP	0.031	−0.025	0.066	−0.137	−0.073	−0.168
P	1	−0.068	0.083	−0.113	0.056	0.016
SC	−0.068	1	−0.179	−0.012	−0.01	−0.147
SS	0.084	−0.179	1	0.015	0.063	0.067
S	−0.113	−0.011	0.014	1	0.431	0.019
SMG	0.056	−0.01	0.063	0.431	1	0.005
T	0.016	−0.147	0.067	0.019	0.005	1

Table 16 Evaluation results of correlation coefficient on mobile price dataset

	BP	B	CS	DS	FC	FG	IM	MD	MW	NC
BP	1	0.019	0.008	−0.036	0.041	0.023	0.003	0.049	−0.013	−0.034
B	0.019	1	0.038	0.053	−0.003	0.020	0.049	−0.004	−0.02	0.048
CS	0.008	0.038	1	0.022	0.009	−0.043	0.03	−0.036	0.018	−0.025
DS	−0.036	0.053	0.022	1	−0.018	−0.013	−0.018	−0.008	−0.011	−0.006
FC	0.041	−0.003	0.009	−0.018	1	−0.003	−0.019	−0.003	0.013	−0.024
FG	0.023	0.020	−0.043	−0.013	−0.003	1	0.003	−0.008	−0.016	−0.033
IM	0.003	0.049	0.03	−0.018	−0.019	0.003	1	0.013	−0.046	−0.03
MD	0.049	−0.004	−0.037	−0.001	−0.003	−0.008	0.013	1	0.022	−0.001
MW	−0.012	−0.02	0.017	−0.011	0.014	−0.016	−0.046	0.023	1	−0.014
NC	−0.034	0.048	−0.025	−0.006	−0.024	−0.033	−0.03	−0.002	−0.014	1
PC	0.039	−0.020	0.014	−0.007	0.637	−0.003	−0.03	0.030	0.025	−0.006
PH	0.002	−0.008	−0.042	−0.030	−0.011	−0.009	0.006	0.028	0.014	−0.027
PW	−0.014	−0.029	−0.020	0.010	−0.004	0.011	0.005	0.025	−0.01	0.021
R	−0.018	0.031	0.007	0.027	0.009	0.012	0.044	0.008	−0.013	−0.016
SH	−0.029	−0.003	−0.004	−0.019	−0.022	0.031	0.044	−0.034	−0.037	−0.008
SW	−0.019	0.024	0.007	−0.029	−0.014	0.032	0.01	−0.031	−0.027	0.024
TT	0.062	0.011	−0.016	−0.044	−0.008	−0.046	−0.008	0.009	0.009	0.007
TG	0.003	−0.032	−0.043	−0.018	0.010	0.586	−0.013	−0.014	−0.002	−0.010
TS	−0.012	−0.001	0.027	−0.006	−0.039	0.036	−0.033	0.006	−0.010	0.008
WF	−0.023	−0.018	−0.003	0.008	0.025	−0.016	0.015	−0.024	0.006	−0.002
	PC	PH	PW	R	SH	SW	TT	TG	TS	WF
BP	0.039	0.001	−0.014	−0.018	−0.028	−0.019	0.062	0.003	−0.012	−0.023
B	−0.02	−0.008	−0.029	0.031	−0.003	0.024	0.011	−0.033	−0.00115	−0.019
CS	0.014	−0.041	−0.020	0.007	−0.004	0.007	−0.017	−0.043	0.027	−0.003
DS	−0.007	−0.030	0.010	0.027	−0.019	−0.028	−0.044	−0.018	−0.006	0.008
FC	0.637	−0.011	−0.003	0.009	−0.021	−0.014	−0.008	0.011	−0.039	0.025
FG	−0.003	−0.009	0.011	0.012	0.031	0.032	−0.047	0.586	0.036	−0.017
IM	−0.03	0.006	0.005	0.044	0.044	0.009	−0.008	−0.013	−0.033	0.015
MD	0.030	0.028	0.025	0.008	−0.034	−0.031	0.009	−0.014	0.006	−0.024
MW	0.025	0.014	−0.01	−0.013	−0.037	−0.027	0.009	−0.002	−0.011	0.006
NC	−0.006	−0.027	0.021	−0.016	−0.008	0.024	0.007	−0.010	0.008	−0.002
PC	1	−0.022	0.008	0.031	−0.004	−0.022	0.006	0.004	−0.034	0.011
PH	−0.022	1	0.525	−0.031	0.056	0.028	−0.018	−0.021	0.031	0.038
PW	0.008	0.525	1	−0.002	0.039	0.036	0.009	−0.007	7.15	0.03
R	0.032	−0.031	−0.002	1	0.02	0.033	−0.011	0.019	−0.024	0.017
SH	−0.004	0.056	0.039	0.02	1	0.505	−0.027	0.022	−0.030	0.028
SW	−0.022	0.028	0.036	0.033	0.505	1	−0.025	0.037	0.01	0.039
TT	0.006	−0.018	0.009	−0.011	−0.027	−0.025	1	−0.046	0.005	−0.016
TG	0.004	−0.021	−0.007	0.02	0.022	0.037	−0.046	1	0.026	0.023
TS	−0.034	0.030	7.15	−0.025	−0.030	0.014	0.005	0.0260	1	0.045
WF	0.011	0.038	0.03	0.017	0.028	0.039	−0.016	0.0230	0.045	1

Code availability The code is available from the first author upon reasonable request.

Funding Not applicable.

Data availability The data that support the findings of this study are available from the first author upon reasonable request.

Declarations

Conflicts of interest/competing interests The authors declare no conflict of interest.

References

1. Ahmad S, Mehfuz S, Mebarek-Oudina F, Beg J (2022) RSM analysis based cloud access security broker: a systematic literature review. Cluster Comput 25:3733–3763
2. Amiri F, RezaeiYousefi M, Lucas C, Shakery A, Yazdani N (2011) Mutual information-based feature selection for intrusion detection systems. J Netw Comput Appl 34(4):1184–1199
3. Battiti R (Jul. 1994) Using mutual information for selecting features in supervised neural net learning. IEEE Trans Neural Netw 5(4):537–550
4. Bhuyan HK, Chakraborty C (2022) Explainable machine learning for data extraction across computational social system. In: IEEE Transactions on Computational Social Systems, pp 1–15. https://doi.org/10.1109/TCSS.2022.3164993
5. Bhuyan HK, Huque MS (2018) Sub-feature selection based classification. In: IEEE Explore, International Conference on Trends in Electronics and Informatics (ICOEI), pp 210–216. https://doi.org/10.1109/ICOEI.2018.8553763
6. Bhuyan HK, Kamila NK (2014) Privacy preserving Sub-feature Selection based on fuzzy probabilities. Cluster Comput (Springer) 17(4):1383–1399
7. Bhuyan HK, Kamila NK (2015) Privacy preserving sub-feature selection in distributed data mining. Appl Soft Compu, Elsevier 36:552–569 ISSN: 1568-4946
8. Bhuyan HK, Ravi VK (2021) Analysis of sub-feature for classification in data mining. In: IEEE Transaction on Engineering Management, pp 1–15. https://doi.org/10.1109/TEM.2021.3098463
9. Bhuyan HK, Mohanty M, Das SR (2012) Privacy preserving for feature selection in data mining using centralized network. Int J Compu Sci Issues (IJCSI) 9(3):434–440
10. Bhuyan HK, Raghu Kumar L, Reddy KR (2019) Optimization model for sub-feature selection in data mining. In: 2nd International Conference on Smart Systems and Inventive Technology (ICSSIT 2019). IEEE Explore, pp 1–6. https://doi.org/10.1109/ICSSIT46314.2019.8987780
11. Bhuyan HK, Kamila NK, Pani SK (2022) Individual privacy in data mining using fuzzy optimization. Engineering Optimization, Taylor & Francis 54(8):1305–1323
12. Bhuyan HK, Ravi V, Brahma B, Kamila NK (2022) Disease analysis using machine learning approaches in healthcare system. Health Technol, Springer 12(5):987–1005
13. Bhuyan HK, Ravi V, Yadav MS (2022) Multi-objective optimization-based privacy in data mining. Cluster Comput (Springer):1-13. https://doi.org/10.1007/s10586-022-03667-3
14. Chen C, Wei J, Peng C, Zhang W, Qin H (2020) Qingdao University, Stony Brook University, improved saliency detection in RGB-D images using two-phase depth estimation and selective deep fusion. IEEE Trans Image Process. https://doi.org/10.1109/TIP.2020.2968250
15. Chitrakar R, Huang C (2014) Selection of candidate support vectors in incremental SVM for network intrusion detection. Comput Sec 45:231–241
16. Chow TW, Huang D (Jan. 2005) Estimating optimal feature subsets using efficient estimation of high-dimensional mutual information. IEEE Trans Neural Netw 16(1):213–224
17. Croft WB, Metzler D, Strohman T (2010) Search engines: information retrieval in practice. Addison-Wesley, Reading, MA, USA
18. Dahiru T (2008) P – value, a true test of statistical significance? a cautionary note. Annals Ibadan Postgrad Med 6(1)
19. Dhaminda B, Abeywickrama NB, Mamei M, Zambonelli F (2020) The SOTA approach to engineering collective adaptive systems. Int J Softw Tools Technol Transfer 22:399–415. https://doi.org/10.1007/s10009-020-00554-3
20. Gakii C, Mireji PO, Rimiru R (2022) Graph based feature selection for reduction of dimensionality in next-generation RNA sequencing datasets, algorithms. MDPI 15(21):1–14
21. Greenland S, Senn SJ, Rothman KJ, Carlin JB, Poole C, Goodman SN, Altman DG (2016) Statistical tests, P values, confidence intervals, and power: a guide to misinterpretations. Eur J Epidemiol 31:337–350
22. He X, Cai D, Niyogi P (2005) Laplacian score for feature selection. Proc Int Conf Neural Inf Process Syst: 507–514
23. Hsu CN, Huang HJ, Dietrich S (2004) The ANNIGMA–wrapper approach to fast feature selection for neural nets. IEEE Trans Syst, Man, Cybern B, Cybern 32(2):207–212
24. https://archive.ics.uci.edu/ml/datasets.php, 2020.
25. https://www.kaggle.com/datasets, 2020.
26. Kamila NK, Jena LD, Bhuyan HK (2016) Pareto-based multi-objective optimization for classification in data mining. Cluster Compu (Springer) 19(4):1723–1745 ISSN: 1386–7857 (print version) ISSN: 1573–7543 (electronic version)
27. Kraskov A, Stogbauer H, Grassberger P (2004) Estimating Є mutual information. Phys Rev E 69(6):066138

28. Kwak N, Choi C-H (Jan. 2002) Input feature selection for classification problems. IEEE Trans Neural Netw 13(1):143–159
29. Li L, Weinberg CR, Darden TA, Pedersen LG (2001) Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method. Bioinformatics 17(12):1131–1142
30. Li Z, Yang Y, Liu J, Zhou X, Lu H (2012) Unsupervised feature selection using nonnegative spectral analysis. Proc 26th AAAI Conf Artif Intell:1026–1032
31. Ma G, Li S, Chen C, Hao A, Qin H (2020) Stage-wise Salient Object Detection in 360° Omnidirectional Image via Object-level Semantical Saliency Ranking. IEEE Trans Vis Comput Graph 26(12):3535–3545. https://doi.org/10.1109/TVCG.2020.3023636
32. Mao KZ (2004) Feature subset selection for support vector machines through discriminative function pruning analysis. IEEE Trans Syst, Man, Cybern B, Cybern 34(1):60–67
33. Myat Thet Nyo F Mebarek-Oudina, SSH, Khan NA (2022) Otsu's thresholding technique for MRI image brain tumor segmentation. Multimed Tools Appl. https://doi.org/10.1007/s11042-022-13215-1
34. Peng H, Long F, Ding C (2005) Feature selection based on mutual information criteria of max-dependency, max-relevance, and minredundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238
35. W. H. Press, P. Flannery, S. A. Teukolsky, W. T. Vetterling, et al., Numerical Recipes, Cambridge UP Cambridge etc, 1986.
36. Rifkin R, Klautau A (2004) In defense of one-vs-all classification. The J Mach Learn Res 5:101–141
37. Rossi F, Lendasse A, François D, Wertz V, Verleysen M (2006) Mutual information for the selection of relevant variables in spectrometric nonlinear modelling. Chemom Intell Lab Syst 80(2):215–226
38. Song J, Takakura H, Okabe Y, Eto M, Inoue D, Nakao K (2011) Statistical analysis of honeypot data and building of kyoto 2006+dataset for nids evaluation. Proc 1st Workshop Building Anal Datasets Gathering Exp Ret Sec:29–36
39. Tabakhi S, Moradi P, Akhlaghian F (2014) An unsupervised feature selection algorithm based on ant colony optimization. Eng Appl Artif Intell 32:112–123
40. Tavallaee M, Bagheri E, Lu W, Ghorbani A-A (2009) A detailed analysis of the kdd cup 99 data set. Proc 2nd IEEE Symp Comput Intell Security Defence Appl:1–6
41. Wan Y, Sun S, Cheng Z (2021) Adaptive similarity embedding for unsupervised multi-view feature selection. IEEE Trans Knowl Data Eng 33(10):3338–3350
42. Wang R, Bian J, Nie F, Li X (2022) Unsupervised Discriminative Projection for Feature Selection. IEEE Trans Knowl Data Eng 34(2):942–953
43. Wang G, Chen C, Fan D-P, Hao A, Qin H (2022) Weakly Supervised Visual-Auditory Saliency Detection with Multigranularity Perception. IEEE Trans Pattern Anal Mach Intell:1–18 (published in Early access)
44. Zaffar M, Hashmani MA, Habib R, Quraishi KS, Irfan M, Alqhtani S, Hamdi M (2022) A hybrid feature selection framework for predicting students performance, computers. Mater Continua 70(1):1893–1920
45. Zhang Y, Zhang Z, Li S, Qin J, Liu G, Wang M, Yan S (Dec. 2019) Unsupervised nonnegative adaptive feature extraction for data representation. IEEE Trans Knowl Data Eng 31(12):2423–2440
46. Zhang L, Liu J, Zhang B, Zhang D, Zhu C (2020) Deep cascade model-based face recognition: when deep-layered learning meets small data. IEEE Trans Image Process 29:1016–1029
47. Zhu J, Liu Y, Wen C, Wu X (2022) DGDFS: dependence guided discriminative feature selection for predicting adverse drug-drug interaction. IEEE Trans Knowl Data Eng 34(1):271–285