Indonesian Journal of Electrical Engineering and Computer Science
Vol. 40, No. 2, November 2025, pp. 758~771
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v40.i2.pp758-771

Ensemble recursive feature elimination-based ensemble classification for medical diagnosis

Thirumalaimuthu Thirumalaiappan Ramanathan[1], Md. Jakir Hossen[1], Abdullah Al Mamun[2], Joseph Emerson Raja[1]
[1]Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia
[2]School of Information and Communication Technology, Griffith University, Nathan, Australia

Article Info

Article history:

Received Aug 22, 2024
Revised Jul 20, 2025
Accepted Oct 14, 2025

Keywords:

Data mining
Ensemble learning
Machine learning
Medical diagnosis
Recursive feature selection

ABSTRACT

The application of data mining techniques for the extraction of patterns from medical datasets is useful in the prediction of various diseases from the data of patients. An appropriate feature selection method is required for the medical datasets to give better results for the medical data mining process. In data preprocessing, feature selection is an important process that finds the most relevant features from the dataset. Considering all features of the medical dataset without using any feature selection process may sometimes lead to inaccurate results. Most of the medical datasets contain meaningless data that are not relevant to the data mining process. These data can be eliminated through the feature selection process. This paper presents an integration of an ensemble feature selection approach and an ensemble classification approach through a classifier called the ensemble recursive feature elimination-based ensemble classifier (ERFE-EC) for the classification of medical data. Four different medical datasets were used for testing the ERFE-EC method, which showed promising results.

Corresponding Author:

Md. Jakir Hossen
Faculty of Engineering and Technology, Multimedia University
Jalan Ayer Keroh Lama, Bukit Beruang, 75450 Melaka, Malaysia
Email: jakir.hossen@mmu.edu.my

1. INTRODUCTION

This research work focuses on enhancing and applying a feature selection method called recursive feature elimination [1] for the medical diagnosis problem by using an ensemble classification approach. The recursive feature elimination method is one of the feature selection methods that selects the best features based on the machine learning classifier and the importance scores of the features generated by the trained classifier. It is possible to generate the feature weights that accurately represent the significance of each feature when a classifier is trained using the dataset. The feature with the lowest weight value is eliminated once the features have been ranked based on their respective weights. Until it runs out of features to train with, the classifier is then retrained using the remaining features. Lastly, the feature importance-based recursive feature elimination method can be used to acquire the whole feature ranking. Some of the latest research works are reviewed below, which apply the recursive feature elimination method for selecting the features from medical datasets.

The recursive feature elimination method based on the support vector machine (SVM) [2] model is used for the feature selection in [3]. Here, the SVM classifier is used for classification from the selected features. In their study, the Wisconsin diagnostic breast cancer (WDBC) dataset [4] is used for testing, where the SVM classifier showed an accuracy of 99%.

The recursive feature elimination method based on logistic regression model is used for the feature selection in [5]. Here, the logistic regression [6], artificial neural network [7], Naïve bayes [8], SVM, and decision tree classifiers are used for classification from the selected features. In their study, the Pima Indian diabetes (PID) dataset [9] is used for testing where all the classifiers showed an average accuracy of 80%.

The recursive feature elimination approach integrated with the decision tree, K-nearest neighbor (KNN) [10], random forest [11], and SVM classifiers were applied for the classification of kidney disease in [12]. Here, the chronic kidney dataset [4] is used for testing their proposed system. Based on their study, the SVM, KNN, decision tree, and random forest classifiers showed the accuracy of 96.67%, 98.33%, 99.17%, and 100%. The recursive feature elimination method based on logistic regression model is used for the feature selection in [13]. Here, the gradient boosting technique [14] based on decision tree learning [15] is used for classification from the selected features. Their proposed system showed an accuracy of 89.7% when testing using the cardiovascular disease dataset [16].

The recursive feature elimination method based on different classifiers including logistic regression, random forest, and decision tree classifiers are studied in [17] for the feature selection from PID dataset. Here, the decision tree, KNN, Naïve bayes, SVM, and random forest classifiers are studied for the classification of diabetes from the selected features. Based on their experiments, the accuracies of classifiers are varied with different recursive feature eliminators.

It can be seen from the reviewed approaches that the efficiency of recursive feature elimination depends on the classifier used with it. For example, if the feature importance scores estimated by the machine learning classifier is not effective for a particular dataset, then the recursive feature elimination method employing that classifier will also be not effective. There are also research gaps from the reviewed approaches in investigating the effectiveness of recursive feature elimination method in feature selection method when applying the recursive feature elimination method through an ensemble approach based on machine learning classifiers such as decision tree, and SVM, and ensemble classifiers like gradient boosting, AdaBoost [18], and random forest.

The efficiency of recursive feature elimination approach can be improved by using an ensemble approach. This research work improves the recursive feature elimination approach by presenting an ensemble classification system called ensemble recursive feature elimination (ERFE) based ensemble classifier (ERFE-EC) which is applied and investigated for the classification of breast cancer, diabetes, heart disease, and Parkinson's disease. The necessity of the recursive feature elimination and ERFE for the feature selection in medical datasets can be investigated by applying the ERFE-EC to different medical datasets.

This paper is organized as follows. The section 2 describes about the proposed ERFE-EC. The section 3 describes the performance of ERFE-EC for the classification of various diseases. The section 4 gives conclusion about the research work presented in this paper.

2. PROPOSED SYSTEM

The dataset used in this research work and the ERFE-EC are described in this section.

2.1. Dataset description

Four medical datasets, including WDBC, heart disease, Parkinson's disease datasets available at University of California machine learning repository [4] and PID dataset available at Kaggle repository [9] are used for testing the ERFE-EC. The WDBC dataset [4] consists of 30 input features which are the standard error (SE), mean, and worst values of features: compactness mean (CM), compactness standard error (CSE), compactness worst (CW), smoothness mean (SM), smoothness SE (SSE), smoothness worst (SW), perimeter mean (PM), perimeter SE (PSE), perimeter worst (PW), area mean (AM), area SE (ASE), area worst (AW), symmetry mean (SYM), symmetry SE (SYSE), symmetry worst (SYW), radius mean (RM), radius SE (RSE), radius worst (RW), texture mean (TM), texture SE (TSE), texture worst (TW), concave points mean (CPM), concave points SE (CPSE), concave points worst (CPW), concavity mean (CYM), concavity SE (CYSE), concavity worst (CYW), fractal dimension mean (FDM), fractal dimension SE (FDSE), and fractal dimension worst (FDW) of the cell nuclei. The output categories of the WDBC dataset are malignant and benign. There are 569 samples in the WDBC dataset.

The PID dataset [9] contains 768 samples where the input features are triceps skin fold thickness (TSFT), plasma glucose concentration (PGC), body mass index (BMI), number of times pregnant (NTP), age, 2-Hour serum insulin (2HSI), diastolic blood pressure (DBP), and diabetes pedigree function (DPF). The output categories of the PID dataset are non-diabetic and diabetic.

The heart disease dataset [4] contains 303 samples where the input features are exercise induced angina (EIA), number of major vessels colored by fluoroscopy (NMVCF), serum cholesterol (SC), resting electrocardiographic results (RES), gender, slope of the peak exercise ST segment (SPESTS), types of chest pain (TCP), thalassemia, resting blood pressure (RBP), fasting blood sugar (FBS), maximum heart rate

achieved (MHRA), age, and oldpeak. The output categories of the heart disease dataset are below 50% narrowing and above 50% narrowing.

The Parkinson's disease dataset [4] consists of 22 input features where the input features are different measures that are estimated by the multidimensional voice program (MVP). The input features of Parkinson's disease dataset are MVP: Fo, MVP: Fhi, MVP: Flo, MVP: jitter (%), MVP: jitter (Abs), MVP: RAP, MVP: PPQ, jitter: DDP, MVP: shimmer, MVP: shimmer (dB), shimmer: APQ3, shimmer: APQ5, MVP: APQ, shimmer: DDA, NHR, HNR, RPDE, D2, DFA, spread1, spread2, and PPE. The output categories of the Parkinson's disease dataset are healthy and Parkinson's disease. There are 195 samples in the Parkinson's disease dataset.

2.2. The proposed classification system

The architecture of ERFE-EC is shown in the Figure 1. In ERFE-EC, the ERFE method combines the decision tree-based recursive feature eliminator (DT-RFE), random forest-based recursive feature eliminator (RF-RFE), AdaBoost based recursive feature eliminator (AB-RFE), gradient boosting based recursive feature eliminator (GB-RFE), and SVM based recursive feature eliminator (SVM-RFE). The decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers are used as the estimators in DT-RFE, RF-RFE, AB-RFE, GB-RFE, and SVM-RFE, respectively, where the best features are selected through the recursive feature elimination method. The majority of the features selected by the DT- RFE, RF-RFE, AB-RFE, GB-RFE, and SVM-RFE are considered as the best features which are processed through an ensemble classifier for final classification. The ensemble classifier employed in ERFE-EC consists of decision tree, KNN, naïve bayes, SVM, AdaBoost, gradient boosting, and random forest classifiers. The classifiers: decision tree, KNN, naïve bayes, random forest, AdaBoost, gradient boosting, and SVM used in ERFE-EC are described below.

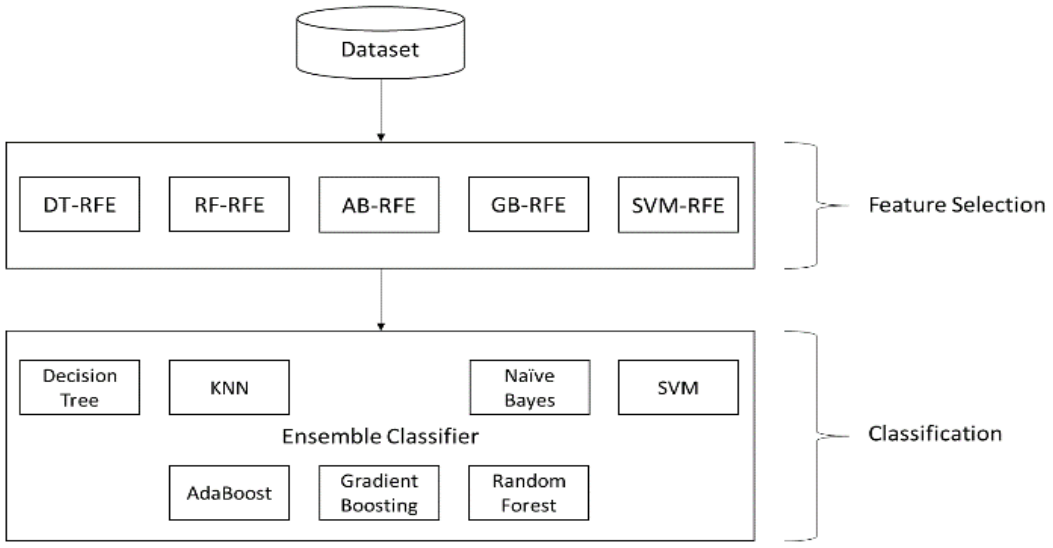

Figure 1. ERFE-EC architecture

2.2.1 Decision tree

A non-parametric supervised learning approach called a decision tree, with its hierarchical tree structure, is used for both regression and classification tasks. It is made up of leaf nodes, internal nodes, branches, and a root node. The decision tree's nodes are connected by directed edges. The internal and root nodes represent the input features of the training dataset. The terminal nodes reflect the output categories that are connected to the training dataset. There will be precise test criteria to divide the internal and root nodes based on their respective categories. The splitting procedure is repeated until the decision tree finds every category of the output variable given in the training dataset.

The decision tree algorithm employs the attribute test condition based on the type of attributes. There are just two possible outcomes when evaluating the binary attributes. Regarding nominal attributes, the outputs produced by the test condition are determined by the number of unique values associated with the relevant qualities. Ordinal attributes enable the number of unique values linked with the appropriate qualities

to be grouped together without going against their property, which could lead to a lot of splits or binary results. When working with continuous characteristics, the test condition can offer a binary split using a comparison test, or many splits using different value ranges.

Entropy [19], Gini impurity [20], and classification error are a few metrics that can be used to find the node that divides the training dataset's samples most efficiently. The splitting method chooses the node with the lowest value when the Gini impurity measure is used, and the node with the highest value when the information gain [21] measure is used. Let us assume that there are p number of output categories and that the subset of samples at node x that belong to category k is represented by $q(k|x)$. The entropy $E(x)$, Gini impurity $G(x)$, and classification error $C(x)$ measurements are found using (1), (2), and (3), respectively.

$$E(x) = - \sum_{i=0}^{p-1} q(k|x) \log_2 q(k|x) \quad (1)$$

$$G(x) = 1 - \sum_{i=0}^{p-1} [q(k|x)]^2 \quad (2)$$

$$C(x) = 1 - \max_j q(k|x) \quad (3)$$

The decision tree classifier in ERFE-EC will be able to choose the best features from datasets based on the Gini impurity measure where the features with lower values are deemed to be more essential in the dataset. The decision tree algorithm is not affected by outliers too much and has the added benefit of being able to model non-linear associations between the features and the target variable [22]. Decision trees do have the ability to perform feature selection during the training of the model [23]. They choose the best features to split on based on their potential to lower impurity (like Gini impurity or entropy in classification). This suggests that features deemed unimportant or less informative are essentially disregarded or minimally used.

2.2.2 KNN

KNN, sometimes referred to as lazy learners, classifies the data by determining how similar the test and training sets are to one another. In the multi-dimensional feature space, each training dataset sample is represented by the KNN as a data point. The distance in the feature space between each new test sample's data point and the other data points is computed. The distance between the data points can be calculated using a variety of distance measures. The majority of the KNN models make use of the Euclidean distance measure. Let us assume that there exist two data points, X_1 and X_2, representing instances, x_{1i} and x_{2i}, respectively, that possess attributes, A_1, A_2, …, A_l. The calculation of the Euclidean distance between X_1 and X_2 is demonstrated by (4).

$$\text{dist}(X_1, X_2) = \sqrt{\sum_{i=1}^{l} (x_{1i} - x_{2i})^2} \quad (4)$$

The K in KNN stands for the number of closest neighbors. The nearest neighbors of the test data point are those that are closest to it. Based on the categories of its closest neighbors in the feature space, the category of every test data point is predicted. The test sample will be assigned to a category if all of the test data point's closest neighbors fall into that group. The category of the majority of the closest neighbors will be applied to the test sample if the test data point's nearest neighbors fall into more than one category. Assume that there is a training dataset D and test instances, $z = (x_i`, y_i`)$. Let D consists of samples $((x_{1i}, y_{1i}), (x_{2i}, y_{2i}), \ldots, (x_{ni}, y_{ni}))$ with characteristics, $A_1, A_2, \ldots, A_l$. Let Y_i be the category of X_i that has to be predicted, $X`$ be the data point of the new test sample $x_i`$, t be the class label, X_i be the data point of sample from D, and Y_i be the category of X_i where $i = 1, 2, \ldots, l$. In the new test sample z, $Y`$ is estimated using the majority vote with (5) for the k neighbors list D_z.

$$Y` = \underset{t}{argmax} \sum_{(X_i, Y_i) \in D_z} I(t = Y_i) \quad (5)$$

In (5), the indicator function I (·) returns 1 if the argument is true and returns 0 otherwise. The number of nearest neighbors is set to the value of three for the KNN model used in the ERFE-EC. KNN is expected to be effective on large datasets [24], when the dimensionality is not very high. Important features are found by KNN through computing the distances between data points within the feature space [25].

2.2.3 Naïve bayes

The naïve bayes algorithm can be viewed as a probability classifier that utilizes the bayes theorem. The naïve bayes algorithm relies on a strong independent assumption between every variable in the dataset given a target variable. The naïve bayes algorithm, despite its simple assumption and ease of implementation, has proven useful for many applications, particularly in data classification problems. The naïve bayes classifier classifies an instance, x_i, of n features, A_1, A_2, …, A_n, and m categories of the target variable, t_1, t_2, ..., t_m, in t_i if and only if $P(t_i|x_i) > P(t_j|x_i)$, for $1 \leq j \leq m$, $j \neq i$. As demonstrated, the Bayes theorem is used to estimate $P(t_i|x_i)$ by using (6) and (7).

$$P(t_i|x_i) = \frac{P(x_i|t_i)P(t_i)}{P(x_i)} \tag{6}$$

$$P(x_i|t_i) = \prod_{k=1}^{n} P(x_k|\ t_i) \tag{7}$$

In this case, $P(t_i)$ is the priori probability of t_i, $P(x_i)$ is the priori probability of x_i, and $P(x_i|t_i)$ is the probability of x_i for a certain category of target variable t_i.

The Gaussian Naïve bayes classifier, which is predicated on the idea that the continuous values associated with each category of the target variable are distributed in accordance with the gaussian distribution, is used to handle the continuous data. In (8) is used to calculate the conditional probability $P(x_i|t_i)$ for the Gaussian distribution.

$$P(x_i|t_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i-\mu)}{2\sigma^2}\right) \tag{8}$$

Here, the variance is represented by σ^2 and the mean is denoted by μ. The gaussian naïve bayes model is used in ERFE-EC. Naive Bayes is effective for large datasets [26], particularly when speed is needed for all steps of the classification, because the only training needed is for calculating the probability distributions of each feature, and making predictions is based on simple calculations using these probabilities.

2.2.4 SVM

SVM can be used to solve the curse of dimensionality issue when working with datasets that have a lot of features. SVM classifiers can be used for both linear and nonlinear data. SVM classifiers conduct classification based on the maximal margin hyperplane technique. The maximal margin hyperplane technique divides the training samples of datasets into groups based on the hyperplanes that correspond to the relevant class labels. Not all the hyperplanes that can be plotted to split the samples are useful for classifying the test samples. The hyperplane with a larger margin will classify the test samples more accurately than the hyperplanes with smaller margins. The linear kernel based SVM classifier uses the data points that are on the borders of different data categories to find a hyperplane that separates the training data points shown on the feature space into different categories. Support vectors are these data points that are utilized to locate the hyperplane. SVM examines the data points (x_i, y_i) considering the training set. Here, x_i is the n-dimensional vector, and y_i is the target variable that is related to x_i, where i = 1, 2, ..., n. The operation of the decision boundary which divides the training data points is shown in (9).

$$w \cdot x + b = 0 \tag{9}$$

Here, w is the n-dimensional weight vector and b is the scalar. The parameters w and b must be calculated during the training phase. The SVM classifier raises the margin of hyperplanes when a particular kind of linear model for the data that are linearly separable is found. The SVM classifier in the ERFE-EC employs a linear kernel. SVM is well-suited for small to medium-sized datasets [27]. SVM is good for feature selection due to the distribution of the relevant features through margin maximization, the type of kernel being linear and nonlinear, and its robustness towards noisy and irrelevant features [28].

2.2.5 AdaBoost

AdaBoost is an ensemble classification technique that combines the results of numerous weak classifiers to produce a powerful classifier. A few criteria are used in the AdaBoost classification approach to choose the weak classifiers. When the training data is distributed randomly, the accuracy of the weak classifiers should be greater than 50%. The weighted training data ought to be manageable for the weak classifiers. When these conditions are met, the AdaBoost classification technique can provide a final

classifier with an accuracy that outperforms all of the chosen weak classifiers. The AdaBoost classification approach goes through several iterations in which it attempts to improve performance by lowering the training process error rate which is estimated from the previous weak classifier for the training data. By adjusting the weights for the training samples on each iteration, the training process's error rate is decreased.

Let $Y = \{y_i\}_{i=1}^{m}$ be the outcome class where $y_i \in \{0,1\}$ and consider $X = \{x_i\}_{i=1}^{m}$ as training data where $x_i \in R^D$. Let $W = \{w_i\}_{i=1/m}^{m}$ be considered where for each sample of the training data, the initial value of w_i is set to $\frac{1}{\mathrm{m}}$. The stages that the AdaBoost classification method takes in each iteration are listed below.

a. The training data with an initial weight value of w_i is used to train the weak classifiers, g_p. Here, the iteration level is indicated by p = 1 to k.
b. The training inaccuracy for the weak classifier g_p, $\mathcal{E}_p$ is calculated at each repetition level, p.
c. Determine α_t using (10).

$$\alpha_p = 0.5 \times (\ln(1- \mathcal{E}_p)/ \mathcal{E}_p) \quad (10)$$

d. Using (11), the weights of the inaccurate samples are changed.

$$w_i^{(p+1)} = \frac{w_i{}^p}{z_j} \times \begin{cases} e^{-\alpha_p} & if \;\; g_p(x_i) = y_i \\ e^{\alpha_p} & if \;\; g_p(x_i) \neq y_i \end{cases} \quad (11)$$

Z_j is the normalizing factor in this case. The above steps above demonstrated how the weight values of the training instances are changed at each loop to guarantee that the best classification is provided by enhancing the output of the weaker classifiers that approached before it. In (12) provides the final AdaBoost classification.

$$H(x') = \underset{f}{\operatorname{argmax}} \sum_{p=1}^{k} \alpha_p \, \mathrm{I}(g_p(x') = c) \quad (12)$$

Here, f is the class label, α_p is the value computed based on the training process error rate at iteration level p, $g_p(x')$ is the classification of the weak classifier at iteration level p for the test sample x', and I $(\cdot)$ is an indicator function that returns 1 if the argument is true or 0 otherwise. The decision tree is used as the weak classifier in AdaBoost model implemented in the ERFE-EC. The AdaBoost algorithm, which focuses on weak learner enhancement, has been reported to demonstrate positive results when classifying medical data [29], [30]. It has the benefits of providing solutions to problems like noise in the data and overfitting. In addition, AdaBoost can construct medical datasets using weak learner outputs to find important features and therefore, perform feature selection [18].

2.2.6 Gradient boosting

Gradient boosting is an ensemble classifier whose output is decided by the weighting scheme. It is built on numerous weak classifiers. In gradient boosting, the regression decision tree is typically employed as the weak classifier. By training each weak learner based on the error of the previous weak learner, gradient boosting reduces the error rate of the training process. Let $Y = \{y_j\}_{j=1}^{m}$ be the outcome class where $y_j \in \{0,1\}$ and consider $X = \{x_j\}_{j=1}^{m}$ as the training data where $x_j \in R^D$. Gradient boosting aims to reduce the aggregation of many specified loss functions $\mathcal{L}(y_j, F(x_j))$ by choosing a classification function $F(x)$, which is provided by (13).

$$F^* = \underset{F}{\operatorname{argmin}} \sum_{j=1}^{m} \mathcal{L}(y_j, F(x_j)) \quad (13)$$

In (14) illustrates the estimating function F in an additive form.

$$F(x) = \sum_{p=1}^{k} f_p(x) \quad (14)$$

In this case, k denotes the iteration count. An incremental pattern of processing is applied to the $\{f_p(x)\}$. In order to maximize the aggregated loss at level p, the recently added function f_p is chosen, keeping $\{f_i\}_{i=1}^{p-1}$ unchanged. Each parameterized weak learner is represented by the function f_i. Let θ be the decision tree's parameter vector. Subsequently, θ comprises characteristics that delineate the decision tree's structure, including the splitting feature and the threshold for splitting individual internal nodes. In (15) illustrates how an estimated loss function is built at the level p.

$$\mathcal{L}(y_j, F_{p-1}(x_j) + f_p(x_j)) \approx \mathcal{L}(y_j, F_{p-1}(x_j)) + g_j f_p(x_j) + \frac{1}{2} f_p(x_j)^2 \quad (15)$$

Here, $F_{p-1}(x_j)$ and g_j are given by (16) and (17), respectively.

$$F_{p-1}(x_j) = \sum_{i=1}^{p-1} f_i(x_j) \quad (16)$$

$$g_j = \frac{\partial \mathcal{L}(y_j, F(x_j))}{\partial F(x_j)} \mid F(x_j) = F_{p-1}(x_j) \quad (17)$$

The decision tree is used as the weak classifier in gradient boosting model implemented in the ERFE-EC. Gradient boosting algorithm, which is an ensemble of weak learners is combined with a strong predictive model in a sequential way to make a more accurate classification [31]. Because of its ability to model complex, nonlinear relationships between features and the target variable, it has become a preferred approach for addressing cases of imbalanced classification [32]. Because of the implicit feature selection which takes place during the building of the model, gradient boosting can also be used on feature selection tasks [33].

2.2.7 Random forest

Several decision trees are used in the ensemble-based random forest classification algorithm. The bagging technique is used by the random forest where the random samples are chosen from the training dataset and the decision trees are fitted to these samples. The majority votes from each decision tree constructed using the random forest model determine the final result. The candidate split technique for each decision tree model in the random forest classifier selects a group of features at random, and the best split feature from that subset is used to split each node of the corresponding decision tree. In the majority of random forest models, the best split selection is carried out using the Gini impurity measure. The dataset's features can also be ranked by the random forest classifier according to their significance. Each feature's quality is estimated using the impurity metrics used in the random forest model. The average impurity measure value for each feature across all the decision trees constructed using the random forest model reflects the feature's importance; for instance, a lower value implies a feature's high importance. The maximum depth of each decision tree is set to the value of two for the random forest model used in the ERFE-EC. Random forest is considered for the classification problems [34], since it is resistant to overfitting as well as being able to account for the missing values in the training data. Random forest is a highly effective algorithm for feature selection [35], thanks to its provision of built-in methods for ranking and selecting the important features based on their contribution in the decision-making process.

The flowchart of ERFE-EC is shown in Figure 2. As shown in Figure 2, initially, the decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers are trained with the dataset individually. Then the trained classifiers are individually used to estimate the scores of features from the dataset. Then the accuracy of the dataset is estimated. Then the feature with the least score is eliminated from the dataset. The resulting dataset is used again to train the classifiers individually. Then again, the trained classifiers are individually used to estimate the scores of features from the dataset. Then again, the accuracy of the dataset is estimated. Then again, the feature with the lowest score is eliminated from the dataset. This process gets repeated until the feature subset becomes empty. Finally, the feature subset with maximum accuracy is used for training the ensemble classifier. The trained ensemble classifier is used for performing the final classification.

3. RESULTS AND DISCUSSION

The performance measures: accuracy, precision, sensitivity, specificity, and F-measure are used to test the performance of machine learning classifiers for the optimized medical datasets obtained from RF-EMLC method. The performance measures: accuracy, sensitivity, and specificity are estimated using (18), (19), and (20), respectively [36]. The performance measures: precision and F-measure are estimated using (21) and (22), respectively [37]. Here, TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (18)$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \quad (19)$$

$$Sensitivity = \frac{TP}{TP+FN} \quad (20)$$

$$Specificity = \frac{TN}{TN+FP} \quad (21)$$

$$F\text{-}measure = \frac{2 \times precision \times sensitivity}{precision+sensitivity} \quad (22)$$

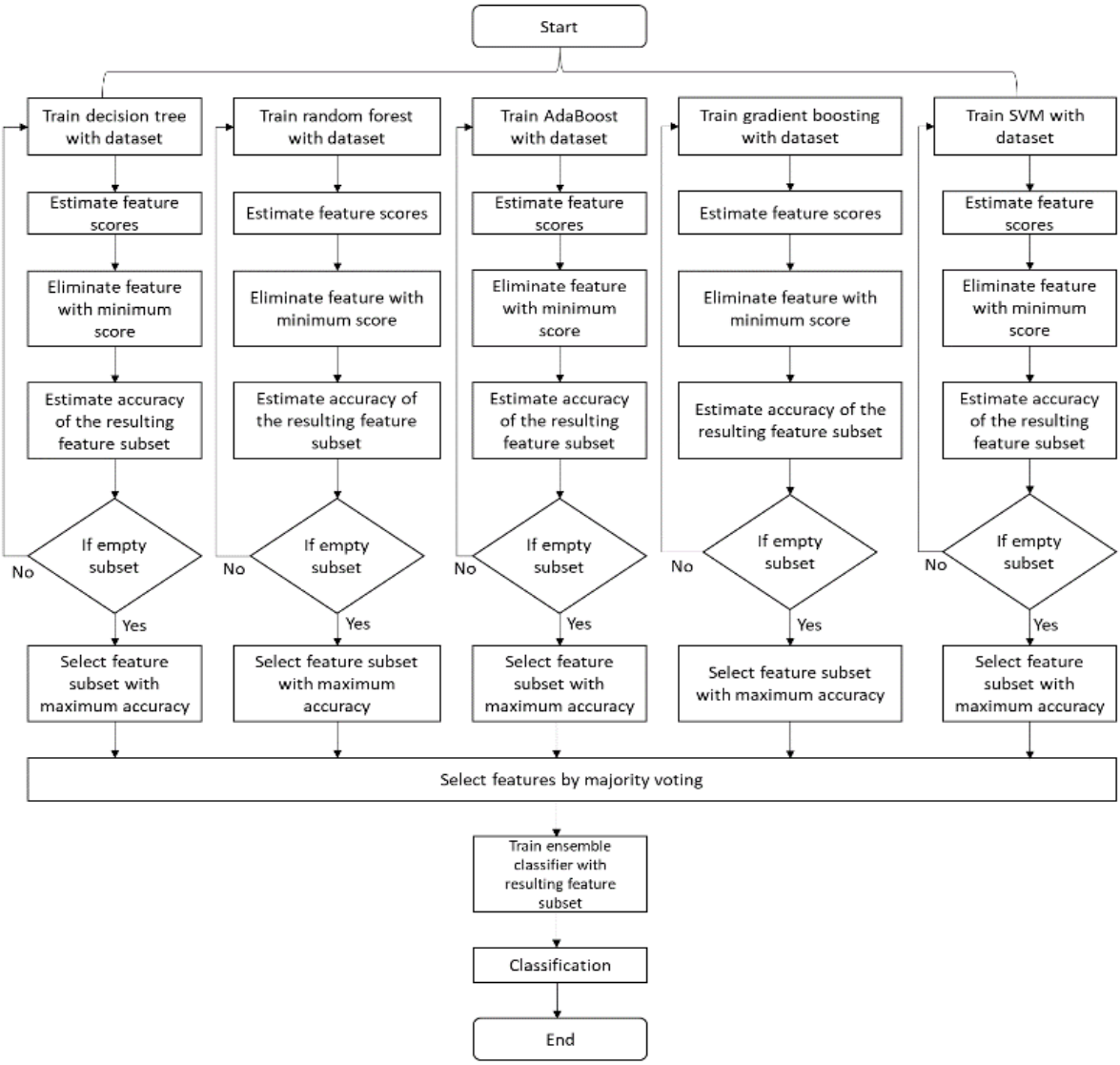

Figure 2. Flowchart of ERFE-EC

The PID diabetes and heart disease datasets contain missing values which are replaced by KNN imputation method [38]. Tables 1 shows the list of features that are selected by various recursive feature eliminators such as DT-RFE, RF-RFE, AB-RFE, GB-RFE, SVM-RFE and their respective classification accuracy for the WDBC, PID, heart disease, and Parkinson's disease datasets, respectively.

As shown in Table 1, the AB-RFE and RF-RFE showed relatively high accuracy, whereas the DT-RFE and SVM-RFE showed relatively low accuracy for the WDBC dataset. The SVM-RFE showed relatively high accuracy whereas the DT-RFE, AB-RFE, and GB-RFE showed relatively low accuracy for the PID dataset. The RF-RFE showed relatively high accuracy whereas the DT-RFE showed relatively low accuracy for the heart disease dataset. The AB-RFE showed relatively high accuracy whereas the DT-RFE and SVM-RFE showed relatively low accuracy for the Parkinson disease dataset. Table 2 shows the common

list of features that are selected by the ERFE method based on the majority voting from the recursive feature eliminators and their respective classification accuracy for the four medical datasets.

Table 1. Features selected by various recursive feature eliminators for different medical datasets

Datasets	Classifiers	Selected features	Accuracy (%)
WDBC	Decision tree	RW, TW, SW, CPW	93
WDBC	AdaBoost	TM, PM, SM, CM, CYM, CPM, SYM, RSE, TSE, PSE, ASE, SSE, CSE, CPSE, SSE, FDSE, RW, TW, PW, AW, SW, CW, CYW, CPW, SW, FDW	97
WDBC	Gradient boosting	TM, PM, AM, CYM, CPM, SYM, FDM, RSE, TSE, ASE, SSE, CSE, CPSE, FDSE, RW, TW, PW, AW, SW, CW, CYW, CPW, SW, FDW	96
WDBC	Random forest	RM, TM, PM, AM, CM, CYM, CPM, ASE, RW, TW, PW, AW, SW, CW, CYW, CPW, SW	97
WDBC	SVM	RM, CYM, TSE, RW, SW, CW, CYW, CPW, SW	93
PID	Decision tree	PGC, 2HSI, BMI, DPF, age	75
PID	AdaBoost	PGC, TSFT, 2HSI, BMI, DPF	75
PID	Gradient boosting	NTP, PGC, 2HSI, BMI, age	75
PID	Random forest	PGC, TSFT, 2HSI, BMI, age	77
PID	SVM	NTP, PGC, DBP, BMI, DPF	81
Heart disease	Decision tree	Age, gender, TCP, RBP, SC, FBP, RER, MHRA, EIA, oldpeak, SPESTS, NMVCF, thalassemia	74
Heart disease	AdaBoost	Age, gender, TCP, RBP, SC, MHRA, EIA, oldpeak, SPESTS, NMVCF, thalassemia	82
Heart disease	Gradient boosting	age, gender, TCP, RBP, SC, MHRA, EIA, oldpeak, SPESTS, NMVCF, thalassemia	79
Heart disease	Random forest	age, gender, TCP, RBP, SC, RER, MHRA, EIA, oldpeak, SPESTS, NMVCF, thalassemia	85
Heart disease	SVM	Gender, TCP, RER, EIA, oldpeak, SPESTS, NMVCF, thalassemia	84
Parkinson's disease	Decision tree	MVP: Fo, MVP: Fhi, shimmer: APQ5, RPDE, spread2, PPE	87
Parkinson's disease	AdaBoost	MVP: Fo, shimmer: APQ5, DFA, spread2, PPE	92
Parkinson's disease	Gradient boosting	MVP: Fo, MVP: Fhi, shimmer: APQ5, D2, PPE	90
Parkinson's disease	Random forest	MVP: Fo, MVP: Flo, spread1, spread2, PPE	90
Parkinson's disease	SVM	MVP: shimmer (dB), RPDE, DFA, spread1, D2	87

Table 2. Final list of features selected by the ERFE-EC method

Datasets	Selected final features
WDBC	TM, PM, CYM, CPM, TSE, ASE, RW, TW, PW, AW, SW, CW, CYW, CPW, SW
PID	PGC, 2HSI, BMI, DPF, Age
Heart Disease	Age, gender, TCP, RBP, SC, RER, MHRA, EIA, oldpeak, SPESTS, NMVCF, thalassemia
Parkinson's Disease	MVP: Fo, shimmer: APQ5, spread2, PPE

Figure 3 compares the classification accuracies of decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers with and without the ERFE feature selection method for the WDBC dataset. Figure 4 compares the classification accuracies of decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers with and without the ERFE feature selection method for the PID dataset. Figure 5 compares the classification accuracies of decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers with and without the ERFE feature selection method for the heart disease dataset. Figure 6 compares the classification accuracies of decision tree, random forest, AdaBoost, gradient boosting, and SVM classifiers with and without the ERFE feature selection method for the Parkinson's disease dataset.

As shown in Figure 3, the decision tree, random forest, AdaBoost, and gradient boosting classifiers showed relatively high accuracies for the optimized WDBC dataset after the ERFE based feature selection process when compared to their accuracies that are evaluated for the whole WDBC dataset. But the SVM classifier showed relatively low accuracy for the optimized WDBC dataset after the ERFE based feature selection process when compared to its accuracy that is evaluated for the whole WDBC dataset.

As shown in Figure 4, the SVM classifier showed relatively high accuracy for the optimized PID dataset after the ERFE based feature selection process when compared to its accuracy that is evaluated for the whole PID dataset. But the decision tree, random forest, AdaBoost, and gradient boosting classifiers showed relatively low accuracies for the optimized PID dataset after the ERFE based feature selection process when compared to their accuracies that are evaluated for the whole PID dataset.

As shown in Figure 5, the SVM classifier showed relatively high accuracy for the optimized heart disease dataset after the ERFE based feature selection process when compared to its accuracy that is

evaluated for the whole heart disease dataset. But the random forest, AdaBoost, and gradient boosting classifiers showed relatively low accuracies for the optimized heart disease dataset after the ERFE based feature selection process when compared to their accuracies that are evaluated for the whole heart disease dataset. In case of the decision tree classifier, it showed the same accuracy for the heart disease dataset when evaluated with and without the feature selection process.

As shown in Figure 6, the random forest, gradient boosting, and SVM classifiers showed relatively low accuracies for the optimized Parkinson's disease dataset after the ERFE based feature selection process when compared to their accuracies that are evaluated for the whole Parkinson's disease dataset. In case of the decision tree and AdaBoost classifiers, they showed the same accuracies for the Parkinson's disease dataset when evaluated with and without the feature selection process.

Figure 7 shows the performance measures of ERFE-EC for the four medical datasets. As shown in Figure 7, the sensitivity of ERFE-EC is relatively high, and the precision of ERFE-EC is relatively low when compared to the other performance measures for the WDBC dataset. The specificity of ERFE-EC is relatively high, and the sensitivity of ERFE-EC is relatively low when compared to the other performance measures for the PID dataset. The sensitivity of ERFE-EC is relatively high, and the specificity of ERFE-EC is relatively low when compared to the other performance measures for the heart disease dataset. The sensitivity of ERFE-EC is relatively high, and the specificity of ERFE-EC is relatively low when compared to the other performance measures for the Parkinson's disease dataset.

This study investigated the effects of recursive feature elimination method through an ensemble-based approach. While earlier studies [3], [5] have explored the impact of recursive feature elimination method for classifying the WDBC and PID datasets, they have not explicitly addressed its influence on medical data classification using an ensemble-based approach. We found that the effectiveness of ERFE method gets varied according to the testing datasets as shown in Figures 3-6. The ERFE method proposed in this study reduced the WDBC dataset to the most impactful features, preserving only those essential for predicting more accurate outcomes as shown in Figure 3.

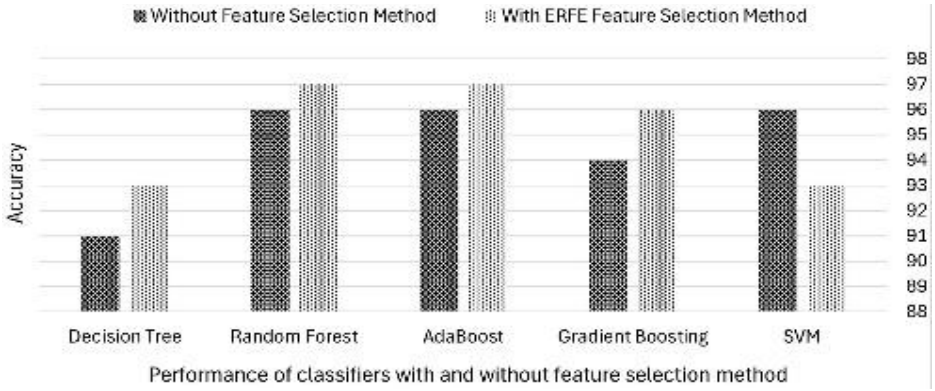

Figure 3. Performance comparison of various classifiers for the WDBC dataset

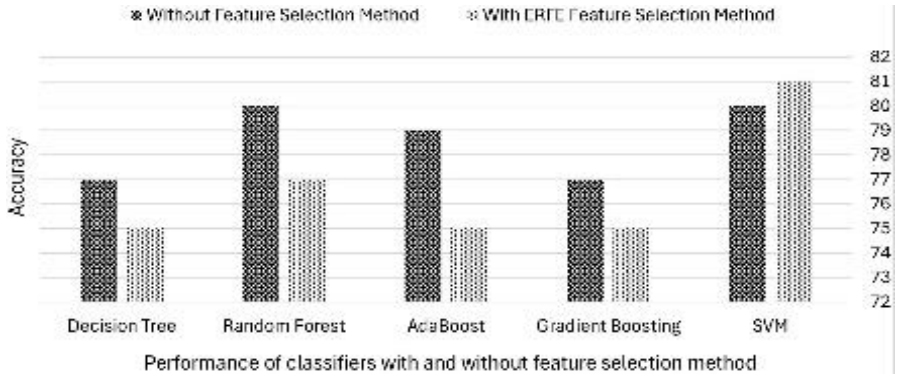

Figure 4. Performance comparison of various classifiers for the PID dataset

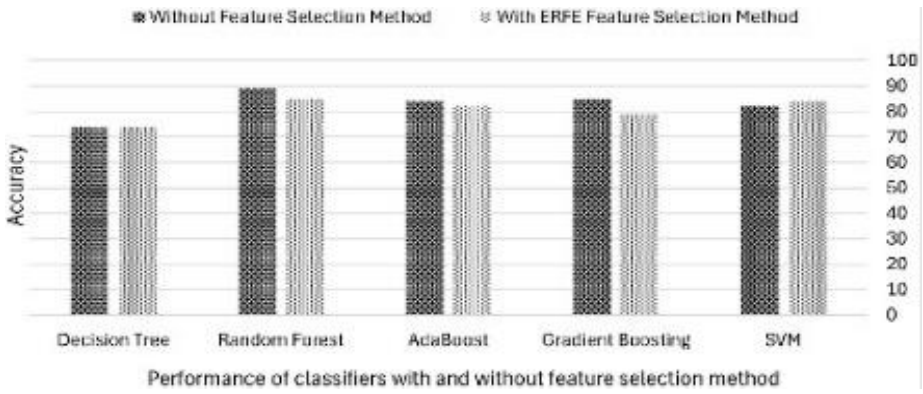

Figure 5. Performance comparison of various classifiers for the heart disease dataset

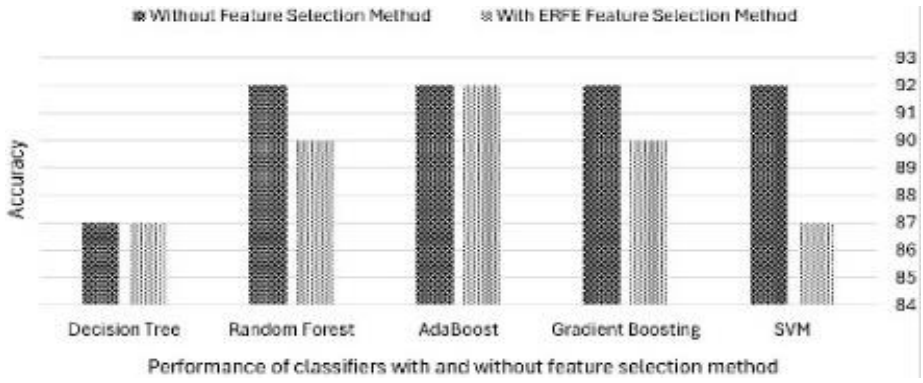

Figure 6. Performance comparison of various classifiers for the Parkinson's disease dataset

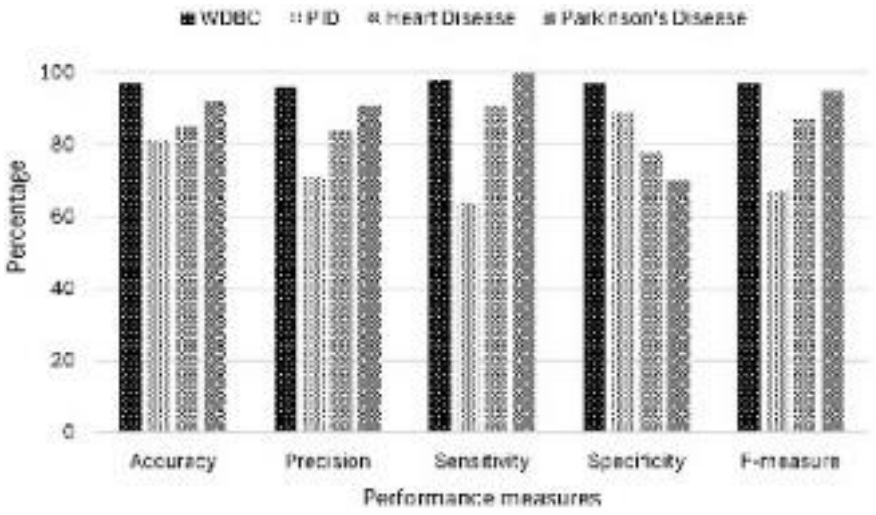

Figure 7. Performance of ERFE-EC

Although the accuracy of ERFE-EC is relatively low as shown in Figure 7, the sensitivity of ERFE-EC is relatively high for the WDBC, heart disease dataset, and Parkinson's disease datasets which shows that the ERFE-EC is able to show the positive test results for anybody who has the illness. The specificity of ERFE-EC is relatively high for the PID dataset which shows that the ERFE-EC is able to show the negative test results for anybody who doesn't have diabetes. Although there are research works done on applying ERFE [39]-[41], an investigation of recursive feature elimination process for the medical diagnosis problem

in an ensemble architecture with using an ensemble of maximum machine learning classifiers is a research challenge when we apply such an ensemble feature extraction process for the medical datasets. To address this research investigation, ERFE-EC is proposed and applied for the classification of different medical datasets. Our study suggests that higher number of machine learning classifiers in ERFE-EC is not associated with poor performance in classification. The proposed ERFE-EC may benefit in producing optimizing datasets without adversely impacting its performance on classification. This study explored a comprehensive study on recursive feature elimination method with an ensemble-based approach. However, further and in-depth studies may be needed to confirm the efficiency of ERFE-EC, especially regarding its performance with different deep learning models. Our study demonstrates that ERFE method is more resilient than the recursive feature elimination method. Future studies may explore the performance of EFRE-EC based on deep learning models with feasible ways of producing optimized datasets. Recent observations suggest that the feature selection approach is not necessary for all the datasets as some models showed slightly improved classification accuracy only when getting trained with the whole dataset when compared to the classification accuracy of the EFRE-EC as shown Figures 4-6. However, the outcomes of the EFRE-EC based on sensitivity and specificity measures implies that the ensemble feature selection approach-based classifier is more effective in diagnosing the diseases. Our findings provide conclusive evidence that this phenomenon is associated with datasets, not due to elevated numbers of multiple recursive feature elimination methods based on different machine learning algorithms.

4. CONCLUSION

ERFE-EC ensembles several machine learning classifiers for the classification of medical datasets. In ERFE-EC, the decision tree, SVM, and ensemble classifiers, including random forest, AdaBoost, and gradient boosting classifiers, are used to select the best features from medical datasets based on the recursive feature elimination process through an ensemble approach. The ensemble classifier consisting of decision tree, KNN, naïve bayes, SVM, AdaBoost, gradient boosting, and random forest classifiers is used to perform the final classification from the optimized dataset in ERFE-EC. The ERFE-EC showed promising outcomes for the WDBC, PID, heart disease, and Parkinson's disease datasets as discussed in Section III which proves the effectiveness of ERFE-EC for the medical data classification. Based on the performance of ERFE-EC in different medical datasets, it can be concluded that in order to make the recursive feature elimination method function properly, it is necessary to select an appropriate model (for instance, feature ranking). Although the recursive feature elimination method can be computationally expensive for large datasets with many features, as it requires training the model multiple times, it can be a very powerful technique. Recursive feature elimination method is a powerful method to boost model performance, increase interpretability, and overcome overfitting, particularly when working with high-dimensional datasets.

FUNDING INFORMATION

Authors state no funding involved.

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, pp. 389-422, 2002, doi: 10.1023/A:1012487302797.

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273-297, 1995, doi: 10.1023/A:1022627411411.

[3] M. H. Memon, J. P. Li, A. U. Haq, M. H. Memon, and W. Zhou, "Breast cancer detection in the IOT health environment using modified recursive feature selection," *Wireless Commun. Mobile Comput.*, vol. 2019, no. 1, p. 5176705, 2019, doi: 10.1155/2019/5176705.

[4] D. Dua and C. Graff, "UCI machine learning repository," *Irvine, CA: Univ. of Calif., Sch. Inf. Comput. Sci.*, 2019. Accessed: Aug. 1, 2024. [Online]. Available: http://archive.ics.uci.edu/ml

[5] P. Misra and A. S. Yadav, "Improving the classification accuracy using recursive feature elimination with cross-validation," *Int. J. Emerging Technol.*, vol. 11, no. 3, pp. 659-665, 2020.

[6] D. R. Cox, "The regression analysis of binary sequences," *J. R. Stat. Soc.: Ser. B (Methodol.)*, vol. 20, no. 2, pp. 215-232, 1958, doi: 10.1111/j.2517-6161.1958.tb00292.x.
[7] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, no. 6, pp. 386-408, 1958, doi: 10.1037/h0042519.
[8] G. I. Webb, "Naïve bayes," *Encycl. Mach. Learn.*, vol. 15, pp. 713-714, 2010.
[9] "Pima Indians diabetes database," *Kaggle*, 2016. Accessed: Aug. 1, 2024. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
[10] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *Am. Stat.*, vol. 46, no. 3, pp. 175-185, 1992, doi: 10.1080/00031305.1992.10475879.
[11] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, p. 5-32, 2001, doi: 10.1023/A:1010933404324.
[12] E. M. Senan et al., "Diagnosis of chronic kidney disease using effective classification algorithms and recursive feature elimination techniques," *J. Healthcare Eng.*, vol. 2021, no. 1, p. 1004767, 2021, doi: 10.1155/2021/1004767.
[13] P. Theerthagiri and J. Vidya, "Cardiovascular disease prediction using recursive feature elimination and gradient boosting classification techniques," *Expert Syst.*, vol. 39, no. 9, e13064, 2022, doi: 10.1111/exsy.13064.
[14] J. H. Friedman, "Stochastic gradient boosting," *Comput. Stat. Data Anal.*, vol. 38, no. 4, pp. 367-378, 2002, doi: 10.1016/S0167-9473(01)00065-2.
[15] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, "Classification and Regression Trees (1st ed.)," *Boca Raton: CRC Press*, 1984, doi: 10.1201/9781315139470.
[16] S. Ulianova, "Cardiovascular disease dataset," *Kaggle*, 2018. Accessed: Aug. 1, 2024. [Online]. Available: https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset
[17] E. Sabitha and M. Durgadevi, "Improving the diabetes diagnosis prediction rate using data preprocessing, data augmentation and recursive feature elimination method," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 9, pp. 921-930, 2022, doi: 10.14569/IJACSA.2022.01309107.
[18] R. E. Schapire, "Explaining adaboost," in *Empirical Inference*, B. Schölkopf, Z. Luo, V. Vovk, Springer, Berlin, Heidelberg, pp. 37–52, 2013, doi: 10.1007/978-3-642-41136-6_5.
[19] C. E. Shannon, "A note on the concept of entropy," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379-423, 1948.
[20] T. Daniya, M. Geetha, and K. S. Kumar, "Classification and regression trees with Gini index," *Adv. Math.: Sci. J.*, vol. 9, no. 10, pp. 8237-8247, 2020, doi: 10.37418/amsj.9.10.53.
[21] Y. Wang, Y. Li, Y. Song, X. Rong, and S. Zhang, "Improvement of ID3 algorithm based on simplified information entropy and coordination degree," *Algorithms*, vol. 10, no. 4, p. 124, 2017, doi: 10.3390/a10040124.
[22] H. Sharma and S. Kumar, "A survey on decision tree algorithms of classification in data mining," *Int. J. Sci. Res.*, vol. 5, no. 4, pp. 2094-2097, 2016.
[23] K. Grabczewski and N. Jankowski, "Feature selection with decision tree criterion," *Proc. IEEE Fifth Int. Conf. Hybrid Intell. Syst.*, pp. 6, 2005, doi: 10.1109/ICHIS.2005.43.
[24] Z. Deng, X. Zhu, D. Cheng, M. Zong, and S. Zhang, "Efficient kNN classification algorithm for big data," *Neurocomputing*, vol. 195, pp. 143-148, 2016, doi: 10.1016/j.neucom.2015.08.112.
[25] P. Cunningham and S. J. Delany, "K-nearest neighbour classifiers-a tutorial," *ACM Comput. Surv.*, vol. 54, no. 6, pp. 1-25, 2021, doi: 10.1145/3459665.
[26] S. S. Bafjaish, "Comparative analysis of Naive Bayesian techniques in health-related for classification task," *J. Soft Comput. Data Min.*, vol. 1, no. 2, pp.1-10, 2020.
[27] J. Nayak, B. Naik, and H. S. Behera, "A comprehensive survey on support vector machine in data mining tasks: applications & challenges," *Int. J. Database Theory Appl.*, vol. 8, no. 1, pp. 169-186, 2015, doi: 10.14257/ijdta.2015.8.1.18.
[28] M. L. Huang, Y. H. Hung, W. M. Lee, R. K. Li, and B. R. Jiang, "SVM-RFE based feature selection and Taguchi parameters optimization for multiclass SVM classifier," *Sci. World J.*, vol. 2014, p. 795624, 2014, doi: 10.1155/2014/795624.
[29] P. Thamilselvan, "Lung cancer prediction and classification using AdaBoost data mining algorithm," *Int. J. Comput. Theory Eng.*, vol. 14, no. 4, pp.149-154, 2022, doi: 10.7763/IJCTE.2022.V14.1322.
[30] S. Gamil, F. Zeng, M. Alrifaey, M. Asim, and N. Ahmad, "An efficient AdaBoost algorithm for enhancing skin cancer detection and classification," *Algorithms*, vol. 17, no. 8, p. 353, 2024, doi: 10.3390/a17080353.
[31] F. H. Yagin, I. B. Cicek, and Z. Kucukakcali, "Classification of stroke with gradient boosting tree using smote-based oversampling method," *Med.*, vol. 10, no. 4, pp.1510-1515, 2021, doi: 10.5455/medscience.2021.09.322.
[32] N. Cahyana, S. Khomsah, and A. S. Aribowo, "Improving imbalanced dataset classification using oversampling and gradient boosting," *Proc. IEEE 5th Int. Conf. Sci. Inf. Technol.*, pp. 217-222, 2019, doi: 10.1109/ICSITech46713.2019.8987499.
[33] A. I. Adler and A. Painsky, "Feature importance in gradient boosting trees with cross-validation feature selection," *Entropy*, vol. 24, no. 5, p. 687, 2022, doi: 10.3390/e24050687.
[34] M. Z. Alam, M. S. Rahman, and M. S. Rahman, "A Random Forest based predictor for medical data classification using feature ranking," *Inf. Med. Unlocked*, vol. 15, p. 100180, 2019, doi: 10.1016/j.imu.2019.100180.
[35] M. A. M. Hasan, M. Nasser, S. Ahmad, and K. I. Molla, "Feature selection for intrusion detection using random forest," *J. Inf. Secur.*, vol. 7, no. 3, pp. 129-140, 2016, doi: 10.4236/jis.2016.73009.
[36] W. Zhu, N. Zeng, and N. Wang, "Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations," in *NESUG Proc.: Health Care Life Sci.*, Baltimore, Maryland, November 2010.
[37] D. M. W. Powers, "Evaluation: From precision, recall and F-Measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011, doi: 10.48550/arXiv.2010.16061.
[38] O. Troyanskaya et al., "Missing value estimation methods for DNA microarrays," *Bioinf.*, vol. 17, no. 6, pp. 520-525, 2001, doi: 10.1093/bioinformatics/17.6.520.
[39] W. Lian, G. Nie, B. Jia, D. Shi, Q. Fan, and Y. Liang, "An intrusion detection method based on decision tree-recursive feature elimination in ensemble learning," *Math. Probl. Eng.*, vol. 2020, no. 1, p. 2835023, 2020, doi: 10.1155/2020/2835023.
[40] N. V. Sharma and N. S. Yadav, "An optimal intrusion detection system using recursive feature elimination and ensemble of classifiers," *Microprocessors Microsyst.*, vol. 85, p. 104293, 2021, doi: 10.1016/j.micpro.2021.104293.
[41] A. Filali, C. Jlassi, and N. Arous, "Recursive feature elimination with ensemble learning using som variants," *Int. J. Comput. Intell. Appl.*, vol. 16, no. 01, p. 1750004, 2017, doi: 10.1142/S1469026817500043.

BIOGRAPHIES OF AUTHORS

Thirumalaimuthu Thirumalaiappan Ramanathan is currently working as a Research Fellow in the Faculty of Engineering and Technology, Multimedia University, Malaysia. He received the Bachelor of Engineering degree in computer science from Anna University, India in 2010. He received the Master of Information Sciences degree from University of Canberra, Australia in 2016. He received the PhD degree from Multimedia University, Malaysia in 2024. His research interests are application of artificial intelligence techniques in health informatics. He can be contacted at email: thirumalaimuthu@mmu.edu.my.

Dr. Md. Jakir Hossen is currently working as an Associate Professor in the Department of Robotics and Automation, Faculty of Engineering and Technology, Multimedia University, Malaysia. He received the master degree in communication and network engineering from Universiti Putra Malaysia, Malaysia in 2003. He received the PhD degree in smart technology and robotic engineering from Universiti Putra Malaysia, Malaysia in 2012. His research interests are application of artificial intelligence techniques in data analytics, robotics control, data classifications and predictions. He can be contacted at email: jakir.hossen@mmu.edu.my.

Abdullah Al Mamun is currently pursuing a PhD in the School of Information and Communication Technology at Griffith University, Australia. He is also associated with the Imaging and Computer Vision Group, Data61, CSIRO, Australia, for his PhD. He received the Bachelor of Science degree in electrical and electronic engineering from Pabna University of Science and Technology, Bangladesh in 2018. He received the Master of Engineering degree from Multimedia University, Malaysia in 2021. His research interests are image processing, machine learning, and deep learning. He can be contacted at email: abdullahal.mamun@griffithuni.edu.au

Joseph Emerson Raja is currently working as an Assistant Professor in the Faculty of Engineering and Technology, Multimedia University, Malaysia. He received the Bachelor of Engineering and Master of Engineering degree in computer science from University of Madras, India in 1989 and 2001, respectively. He received the PhD degree in engineering from Multimedia University, Malaysia in 2015. His research interests are application of soft computing techniques in machine condition monitoring. He can be contacted at email: emerson.raja@mmu.edu.my.