International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-025-02621-0

ORIGINAL ARTICLE

FFS-IML: fusion-based statistical feature selection for machine learning-driven interpretability of chronic kidney disease

Grace Ugochi Nneji[1,2] · Happy Nkanta Monday[1,2] · Venkat Subramanyam Reddy Pathapati[3] · Saifun Nahar[3] · Goodness Temofe Mgbejime[4] · Edwin Sunday Umana[5] · Md Altab Hossin[6]

Received: 22 March 2024 / Accepted: 21 March 2025

Abstract

Chronic kidney disease (CKD) is a prevalent and serious global health issue, with a significant impact on individuals globally. Hence, it is imperative to promptly obtain an accurate diagnosis and interpretation for the commencement of appropriate treatment as timely detection and intervention can enhance the probability of long-term survival. Existing projection-based methods for feature selection do not yield desired outcomes due to their different objectives necessitating the need for innovation approaches for a higher predictive performance. This study proposes a novel fusion-based feature selection (FFS) model for the optimization and selection of distinct features to enhance CKD diagnosis. This study utilizes the University of California, Irvine (UCI) CKD dataset and addresses missing data and imbalance issues through Multiple Imputations by Chain Equation (MICE) and Borderline Synthetic Minority Oversampling Technique (Borderline-SMOTE). The proposed model integrates different machine learning (ML) classifiers, conventionally known as black boxes, with SHAP values to provide interpretability and gain transparency in the decision-making process. The proposed FFS model performs better than single feature selection approaches, achieving 100% in all of the evaluation metrics for support vector machine, light gradient boosting, random forest, voting and extreme gradient boosting classifiers compared to other existing literature that also utilized the same dataset. Notably, the SHAP analysis reveals that features such as red blood cell, white blood cell count and the pus cell clumps show model specific interactions. This aids healthcare in understanding and effectively applying the model's outputs. Empirical evidence demonstrates that our proposed approach exhibits superior performance which has the potential to complement physicians' diagnosis of kidney diseases. Also, the incorporation of explainability enhances the clarity of outcomes and facilitates the identification of the underlying cause of the diseases, contributing to more transparency and ethically sound AI applications in healthcare.

Keywords Chronic kidney disease (CKD) · Explainability · Machine learning · Feature selection · Fusion-based · MICE · BorderlineSMOTE · Prediction · SHAP values

✉ Grace Ugochi Nneji
grace.nneji@zy.cdut.edu.cn

✉ Happy Nkanta Monday
happy.monday@zy.cdut.edu.cn

[1] School of International Education, Chengdu University of Technology Oxford Brookes College, Chengdu 610059, China

[2] Intelligent Computing Lab, HACE SOFTTECH, Lagos 102241, Nigeria

[3] University of Missouri St Louis, St. Louis, MO, USA

[4] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

[5] Department of Mathematical Science, Federal University of Technology Akure, P.M.B 704, Ondo, Nigeria

[6] School of Innovation and Entrepreneurship, Chengdu University, Chengdu 610106, China

Published online: 21 April 2025

1 Introduction

Currently, chronic kidney disorders (CKD) are a prominent factor contributing to mortality on a global scale. Chronic kidney disease (CKD) is a degenerative ailment marked by the steady decline in kidney function, potentially leading to kidney failure [1]. Despite notable advancements in the management of CKD, its high cost and rising prevalence have detrimental effects on human life expectancy. Oftentimes, a significant number of individuals are unable to detect Chronic Kidney Disease (CKD) during its initial phase due to the absence of any noticeable symptoms [2]. Consequently, the identification of CKD is feasible solely at the later stages of the illness when certain symptoms manifest. Nevertheless, it can promptly detect the presence of CKD in an individual by means of blood and urine analysis [3]. These two tests measure the concentrations of creatinine in the blood and protein in the urine. The blood and urine test results reveal the extent of renal disease. When renal function is impaired, the body retains excess fluid and waste, which can potentially result in a range of health complications, including cardiovascular disease and stroke. In addition, conditions such as hypertension, diabetes, hypercholesterolemia, and glomerulonephritis might increase the risk of chronic kidney disease (CKD) [4].

Additionally, early detection of many life-threatening disorders allows for effective management. Machine learning methods are extensively employed in the healthcare industry, namely for the identification and categorization of specific diseases using distinctive dataset [5–10]. These systems will assist clinicians in making precise determinations on patients. The input raw feature space often contains a substantial quantity of useless feature information and tends to have a high dimensionality when data is obtained using feature generation techniques in traditional machine learning systems [6]. Projection-based statistical techniques, such as factor analysis (FA), principal component analysis (PCA), and linear discriminant analysis (LDA), are effective in reducing dimensionality. PCA decreases the number of dimensions in the dataset while retaining important feature information. Factor analysis is a statistical technique that expands on principal component analysis (PCA) by explaining the covariance relationships between variables in regard to underlying components [7]. LDA utilizes the class label to calculate the matrix between and within the class, aiming to identify the directions that provide the most effective separation between the classes. Nevertheless, the unresolved matter remains about the optimal number of components to be maintained in fusion-based feature extraction. Most authors in previous CKD data studies utilize a single-feature extraction method [9, 10]. The PCA, FA, and LDA algorithms employ distinct strategies to convert the original features into a fusion-based feature.

Furthermore, medical data often exposes a disparity issue among different classes. The dimension reduction strategy based on projection, as well as the machine learning algorithm, exhibit poor performance when the dataset is imbalanced and often experience overfitting [11]. Our analysis of the CKD data indicates the presence of missing values, outliers, and a significant imbalance between the higher-class and lower-class instances, with the positive class being more than twice the size of the negative class. Therefore, there is need to resolve the missing values and imbalance dataset by inculcate the MICE imputation technique and the Borderline-SMOTE respectively in this research. In order to enhance the accuracy of CKD patient prediction using a computer-assisted diagnosis procedure, the main objective of this research is to ascertain the presence or not of CKD by analyzing different characteristics derived from the clinical examinations and also considering all the specified strategies for data preprocessing. More so, both single and ensemble machine learning models which include quadratic discriminant analysis (QDA), knearest neighbor (KNN), support vector machine(SVM), decision tree (DT), stacking (sc), random forest (RF), voting, xtreme gradient boosting (XGBoost), bagging, adaptive boosting, light gradient boosting (LightGBM), gradient boosting decision tree (GBDT) are applied to the dataset.

Accurate diagnosis of these disorders and optimal treatment choice are two essential elements towards effectively battling the disease in the shortest possible period and with little financial burden. Modern machine learning models have the ability to accurately diagnose diseases at a reasonably affordable price [12]. Nevertheless, machine learning models exhibit high levels of accuracy and mostly serve a supportive function in the process of medical decision-making. This is a result of the absence of trust and subsequent absence of social approval for the models, as they exhibit a black box behavior by concealing the intricacies of their decision-making process. Queries on the decision-making process and the individual impact of each characteristic on the final result are deemed crucial for instilling confidence in the system's results. Trust is essential between healthcare practitioners and information systems when making judgments. If a healthcare practitioner lacks comprehension of the determinations produced by machine learning models, they are unable to articulate their therapy to patients [13]. The absence of clear elucidation that certain algorithms experience, coupled with the fact that treatment alternatives generally yield lower success rates in routine clinical practice compared to initial evaluation, contribute to the heightened regulatory intricacy [14]. Health practitioners must consider the etiology and consequences of medical conditions, as well as the methodologies and frameworks that aid them in making informed judgments [15, 16]. Interpretability Machine Learning (XML) is a method used to elucidate the opaque functioning of artificial intelligence models. By incorporating the restrictions of explainability through the use of Interpretable Machine Learning (IML) techniques, the comprehensibility of the decision-making process in ML models is enhanced.

Based on the fact that ML in kidney functions as a black box is considered a significant drawback. This is because it is unable to provide tailored assessments by trained nephrologist, which might help clarify clinical information [17]. Certain approaches of interpretation are peculiar to particular models. They can only elucidate the model for which they are specifically developed. Another alternative is the Model agnostic interpretation approach which is applicable for interpreting any machine learning model, regardless of the model's level of complexity. Typically, the model agnostic approaches evaluate data by examining pairs of input and output features and can be applicable to nearly all models. The SHAP [16] technique is widely regarded as the most commonly employed model-agnostic approach. This approach utilizes cooperative game theory to determine the individual influence of each player (attribute) on the game's output, which can be measured with respect to accuracy, precision, recall, and F1-Measure. The SHAP values are effective for interpreting the model, both on a global and local scale. Through global interpretation, the model determines the significance of each predictor (feature) in making predictions. The resulting figure will illustrate both the beneficial and detrimental effects of each input. In addition to enhancing transparency, SHAP values can also offer local interpretation. This method involves analyzing the given data by calculating the SHAP values for each feature of every observation, which provides a localized interpretation of the model. The SHAP approach is a post-hoc technique that is utilized subsequent to model training, thus help build a trustable and comprehensive system for a better prediction and interpretation of the diagnose of CKD. The performance of the proposed strategy will be assessed using the CKD on the UCI machine learning dataset and numerous evaluation metrics are employed to evaluate the performance of each classifier and its interpretability.

The paper introduces the following novel contributions to the diagnosis of CKD:

- A novel fusion-based feature selection is proposed, optimizing the feature selection and enhancing predictive performance compared to existing methods.
- We analyzed comprehensive and optimized hyperparameter tweaking and evaluation of twelve machine learning classifiers, both single and ensemble, to ascertain the effectiveness of the proposed strategy.
- We employed the MICE technique to address the missing values and Borderline-SMOTE technique to manage data imbalance, improving data quality for CKD diagnosis.
- Interpretable Machine Learning technique including SHAP values, are integrated to improve the transparency, clarity, confidence and trustworthiness of the model's decision-making process.

The organization of this paper is followed sequentially: Sect. 2 provides a comprehensive literature review. Section 3 provides rich explanation of the data preprocessing, fusion-based feature selection, machine learning models and the Interpretability approach. Section 4 presents the experimental setup and the evaluation metrics whereas Sect. 5 elucidates the analyses and discussion of the experimental results and comparison with other research works. Section 6 serves as the final section which encompasses the conclusion and potential future work.

2 Related works

The rise in the number of patients with chronic kidney disease (CKD) is apparent, and the diagnostic expenses associated with this condition are considerably higher compared to those of other illnesses. In addition, numerous developing nations suffer from a scarcity of specialized nephrologists. This section will cover different literature review in threefold which include feature selection methods, machine learning models and Interpretability Machine Learning (IML) for the diagnose of CKD.

2.1 Feature selection methods for CKD diagnosis

Feature selection involves the removal of irrelevant attributes and the selection of useful features from a dataset, potentially enhancing the performance and prediction speed of the model. Senan et al. [18] utilized the recursive Feature elimination technique (RFE) with different ML model for an effective diagnosis of CKD achieving all round 100% for accuracy, precision, Recall and F1-score. The same feature selection was also considered with an univariate selection by Ogunleye et al. [19] with different ML model. XGBOOST with RFE achieved all round 100% for accuracy, precision, sensitivity, specificity. Moving over to another feature selection, Chittora et al. [20] utilized three methods which are filter, wrapper and embedded approaches with different ML models and achieved 98.86% accuracy. Drall et al. [21] utilized a correlation and dependence method with just two models, Naïve Bayes and KNN achieved an accuracy of 100%. Elheoseny et al. [22] utilized the density based feature election (DFS) with ant colony based optimization (ACO) algorithm and achieved an accuracy, sensitivity, specificity and F1-score of 95%, 96%, 93.3% and 96% respectively.

Additionally, Metaheuristic algorithms are commonly employed in feature selection approaches to achieve an optimal set of features [23]. Several widely recognized optimization algorithms include the ACO, Genetic Algorithm (GA), Whale Optimization Algorithm (WOA), Particle Swarm Optimization algorithm (PSO), and others [24, 25]. These methods can be employed in machine learning to identify the most optimal set of features or to determine the most optimal set of parameters for any machine learning model. Sakri et al. [26] employed PSO-based feature selection to enhance the accuracy of breast cancer recurrence prediction. Khehra et al. [27] conducted a study comparing GA, PSO, and BBO in their ability to find an optimal feature set from microcalcifications clusters (MCC). The results indicate that BBO outperforms the other two methods, albeit by a little margin. Manonmani et al. [28] applied the ITLBO approach to a CKD dataset and identified the optimal subset of features. Out of the 24 features, 16 were selected as the best subset.

Furthermore, projection-based features techniques like FA, PCA and LDA are effective in reducing dimensionality. PCA decreases the number of dimensions in the dataset while retaining important feature information. Factor analysis is a statistical technique that expands on principal component analysis (PCA) by explaining the covariance relationships between variables in regard to underlying components [7]. LDA utilizes the class label to calculate the matrix that represents the differences and similarities within and between the classes. It aims to identify the directions that provide the most effective separation between the classes and achieving a better prediction performance. Nevertheless, the unresolved matter remains regarding the optimal number of components to keep in projection-based feature extraction. These techniques PCA, FA, and LDA algorithms employ distinct strategies to convert the original attributes and this research endeavor to create a unified feature space that combines PCA, FA, and LDA projection.

2.2 Machine learning model for CKD diagnosis

Computer-assisted diagnostic systems are crucial, and prior studies have highlighted the application of machine learning (ML) techniques to aid expert clinicians in medical health sectors for generating diagnostic judgments [29]. Nevertheless, the identification and assessment of CKD by the utilization of artificial intelligence has gained prominence in recent times. Consequently, several research publications have employed ML techniques as effective automated methods that rely on characteristics to identify and diagnose chronic kidney disease (CKD) in its first phases [30]. Gabriel et al. [31] developed a model based on a twin system of Neural Network and Case-Based Reasoning (NN-CBR) to accurately assess the likelihood of an individual developing Chronic Kidney Disease (CKD). This study revealed that approximately 7% of the population in Colombia is susceptible to developing Chronic Kidney Disease (CKD). Additionally, the RF and SVM model were utilized to compare the results of the NN-CBR system. The NN-CBR system achieved a test dataset accuracy of 95%.

Khan et al. [32] employed seven machine learning approaches in order to categorize CKD as either CKD or NOT-CKD in all of the conducted tests. The model performance was assessed using various assessment measures, including mean absolute error (MAE), root mean squared error (RMSE), relative absolute error (RAE), root relative squared error (RRSE), recall, precision, F-measure, and accuracy. Overall, the findings indicate that CHIRP significantly decreases error categorization rates and enhances accuracy. The CHIRP demonstrated a precision of 99.75% with a Mean Absolute Error (MAE) of 0.0025. Alasker et al. [33] employed various intelligent techniques, including backpropagation neural network, Naïve Bayes, decision trees, k-nearest neighbor, and one rule classifier, to identify renal sickness. They utilized a dataset from the UCI ML repository, which consisted of 24 attributes and 400 individuals. The Naive Bayes approach surpasses other classification algorithms in terms of accuracy and sensitivity. The Naive Bayes (NB) algorithm achieved a remarkable accuracy of 99.36% and an impressively low error rate of 0.0057 by utilizing all 24 attributes in combination.

Padmanaban et al. [34] suggested that machine learning classifier techniques have the potential to detect CKD at an initial stage in diabetic individuals. A total of 600 diabetic patients were included in the study, and data were obtained from the diabetes research facility in Chennai, India. The CKD classification task utilized the Naïve Bayes and Decision tree algorithms, which are the most commonly employed machine learning methods. The performance indicators were measured to evaluate their effectiveness. The experiments were conducted exclusively with the Weka tool. The results indicated that the decision tree method outperformed the Naïve Bayes algorithm, with an accuracy rate of 91%. Drall et al. [21] employed Naïve Bayes and KNN algorithms to identify CKD using the UCI machine learning repository dataset, which consisted of 400 instances and 25 characteristics. They utilized statistical analysis techniques such as mean and mode to address the missing value in the dataset for their research. The correlation and dependence strategy was utilized to generate highly interdependent features for the prediction of CKD. Ultimately, it was demonstrated that the k-nearest neighbor algorithm performed better than the Naïve Bayes algorithm when considering the five features. Demographic and biochemical blood characteristics can sometimes be more important than measuring urine protein levels when predicting chronic kidney disease (CKD) in the future.

Almansour et al. [35] utilized an artificial neural network (ANN) and support vector machine (SVM) to analyze the UCI machine learning repository dataset for the purpose of diagnosing chronic kidney disease (CKD). In their research, the researchers utilized the mean value of the respective attributes to replace any missing values in the dataset. Furthermore, the significant parameters of each classifier were selected using the parameter tuning technique. The analysis determined that Artificial Neural Network (ANN) fared better than Support Vector Machine (SVM), with accuracy rates of 99.75% and 97.75% respectively. Akben et al. [36] developed a prognostic model that integrates the results of urine tests, blood tests, and the patient's medical history to categorize CKD. In order to conduct the tests, the researchers utilized preprocessed data and subsequently employed it in the classification models. The data underwent preprocessing through the utilization of the K-Means clustering technique. The recommended methodologies yielded an accuracy of 97.8%. Elhoseny et al. [22] have introduced a new framework named D-ACO algorithm. This framework combines the density-based feature selection (DFS) with ant colony based optimization and attained a peak accuracy and F-score of 95% and 96%, respectively. Alsuhibany et al. [24] utilized the EDL-CDSS approach to assess the automated detection and categorization of CKD. This article encompassed several procedures, including data collection, outlier detection, hyperparameter tuning, and deep learning-based categorization. The maximum mean accuracy achieved was 96.71%. In this research, it is important to assess the robustness of the model performance on different single and ensemble models after our data processing and fused feature election using the same dataset for the prediction of CKD.

2.3 Interpretability machine learning

It is worth noting that most machine learning models are considered to be “black boxes”. A black-box model refers to a complex model that lacks easy interpretability for humans [38]. Doctors face difficulties in comprehending the underlying factors that led to a certain prediction made by the black box model when employing it as a diagnostic system [14, 31]. The presence of the black box poses obstacles to medical decision support from the viewpoints of both physicians and patients [39]. Therefore, it is imperative to create a diagnostic system that offers the comprehensibility of the machine learning model. The interpretability of the machine learning (ML) model serves as a means to verify the accuracy of the projected outcomes and enhances the confidence of physicians in the system [40]. The increasing interest in the field of eXplainable Artificial Intelligence (XAI) in recent years is driven by the need to enhance the interpretability of machine learning models [41].

Numerous studies investigated the field of Explainable Artificial Intelligence (XAI) and identified several techniques for interpreting machine learning models, such as LIME, Decision Trees, Saliency Maps, and Shapley Explanations [42]. Recently, the SHapley Additive exPlanation (SHAP) has been employed in many medical researches [43, 44] and other domain to determine the significance of features within a given set of features for a robust interpretability. Tasmin et al. [45] employed xgboost and explainability ML model for the diagnosis of lung cancer. Liao et al. [46] utilized an interpretable and predictive model for the diagnose of hypothyroidism. Zhang et al. [47] employed the Shapley Additive exPlanations (SHAP) method to develop a comprehensible model for Reinforcement Learning for Grid Control (RLGC). Dikshit et al. [48] employed SHAP to demonstrate the importance of climatic conditions in influencing drought forecasting. Parsa et al. [49] utilized SHAP to assess the significance of traffic-related attributes in the model that contributes to an increased occurrence of traffic accidents. The relevant studies are listed in Table 1 to illustrate the performance and investigate the motivation based on the literature review. Also, based on these researches and conclusions, we have created an automated and easily understandable ML fusion-based feature selection diagnostic method for Chronic Kidney Disease (CKD). This system identifies the most important factors that contribute to the diagnosis of a patient as either having CKD or not.

3 Materials and methods

This section will discuss the dataset collection, data preprocessing steps, the feature selection strategies, the ML classifiers, and the SHAP values for the proposed approach.

3.1 Description of the dataset

This research conducts an analysis of Chronic Kidney Disease (CKD) utilizing both single and ensemble classifiers. The CKD dataset used in this analysis is publicly available and was obtained from the UCI machine learning repository website [50]. Furthermore, a number of researchers conducted experiments to evaluate the effectiveness of their classification model by utilizing the UCI ML repository dataset, a well-known benchmark dataset, for the purpose of predicting CKD [32–35]. The dataset as seen in Table 2 comprises 400 instances, with 250 classified as CKD and 150 classified as NOTCKD, which is considered as data imbalance. The dataset has a total of twenty-five attributes, with eleven falling into the numeric category and the remaining attributes falling into the nominal category. One

Table 1 Summary of related works

References	Feature selection type	Split ratio	Models	Results (%)
[18]	Recursive feature elimination (RFE)	Train: 75% Val & Test: 75%	SVM, Tree, RF, KNN	Accuracy: 100% Recall: 100% F1-score: 100% Precision: 100%
[19]	RFE, Univariate selection (US) and extra classifier (ETC)	Tenfold cross validation (CV)	KNN, SVM, Tree, LDA, XGBoost, LG	RFE with XGBoost: Accuracy: 100% Sensitivity: 100% Specificity: 100% Precision: 100%
[20]	Wrapper, filter and embedded method	Train and test: 50% each	ANN, Chi-square, LG, Tree, Linear SVM	Linear SVM- Accuracy: 100%
[21]	Dependence and correlation method	–	Naive Bayes (NB) and Tree	Tree – Accuracy: 90%
[22]	Density based feature selection	–	ACO algorithm	Accuracy: 95% F1-Score: 96% Sensitivity: 96% Specificity: 93.33%
[31]	–	–	NN, SVM, RF	NN – Accuracy: 95% Precision: 94% F1-Score: 95% Recall: 97%
[32]	–	Tenfold cross validation (CV)	NBTree, SVM, LG, NB and Composite Hypercube on Iterated Random Projection (CHIRP)	CHIRP – Accuracy: 99.75% Precision: 99.8% F1-Score: 99.8% Recall: 99.8%
[33]	–	–	NN, NB, Tree, KNN	NB – Accuracy: 99.36% Specificity: 100% Sensitivity: 97.7%
[34]	–	Tenfold cross validation (CV)	NB and Tree	Tree – Accuracy: 91%
[35]	–	tenfold CV, Train-Test: 90–10%	ANN, SVM	ANN – Accuracy: 99.75%
[36]	–	–	KNN, SVM, NB	Accuracy: 99.75%
[37]	–	k-fold CV	Ensemble deep neural network (EDNN)	EDNN – Accuracy: 96.71%

attribute which is the target variable denotes the classification of CKD, specifically CKD or Not CKD. Table 2 provides more details regarding the dataset. Every feature in the dataset, except for the class attribute, contains some missing values indicated by a question sign.

3.2 Preprocessing of data

The majority of data in the actual world requires preprocessing to address issues such as inconsistency, missing values, noisy features, and outliers, else, improving the machine model's quality becomes challenging, leading to subpar outcomes. In this subsection, variety of procedures has been executed to cleanse and enhance the data.

3.2.1 Imputation of missing values

The CKD dataset utilized in this work underwent a cleaning process whereby missing values were addressed through the process of filling them in. Prior research have utilized several techniques to handle missing values in datasets, such as deleting rows or columns and employing statistical and machine learning algorithms for imputation. These algorithms include mean, median, mode, logistic regression, K-nearest neighbors (KNN), and others [19, 51–53]. Regrettably, the majority of the studies employed statistical imputation as a means of estimating the value of missing data in the dataset, primarily due to its straightforwardness. This approach involves calculating the mean, mode, and median

Table 2 CKD dataset description [51]

S/no	Description (attribute)	Scale	No. of missing value
Numerical attribute			
1	Age (age)	Years	9
2	Blood pressure (bp)	Mm/Hg	12
10	Blood glucose random (bgr)	mgs/dl	44
11	Blood Urea (bu)	mgs/dl	19
12	Serum Creatinine (sc)	mgs/dl	17
13	Sodium (sod)	mEq/L	87
14	Potassium (pot)	mEq/L	88
15	Hemoglobin (hemo)	Gms	52
16	Packed Cell Volume (pcv)	P cv	70
17	White Blood Cell Count (wbcc)	Cells/cumm	105
18	Red Blood Cell Count (rbcc)	Millions/cmm	130
Nominal attribute			
3	Specific gravity (sg)	1.005 to 1.025	47
4	Albumin (al)	0 to 5	46
5	Sugar (su)	0 to 5	49
6	Red blood cells(rbc)	Abnormal, Normal	152
7	Pus cell (pc)	Abnormal, Normal	65
8	Pus cell clumps (pcc)	Not present, Present	40
9	Bacteria (ba)	Not present, Present	4
19	Hypertension (htn)	No, Yes	2
20	Diabetes Mellitus (dm)	No, Yes	2
21	Coronary Artery Disease (cad)	No, Yes	2
22	Appetite (appet)	Poor, Good	1
23	Pedal Edema (pe)	No, Yes	1
24	Anemia (ane)	No, Yes	1
25	Class (class)	CKD, Not CKD	0

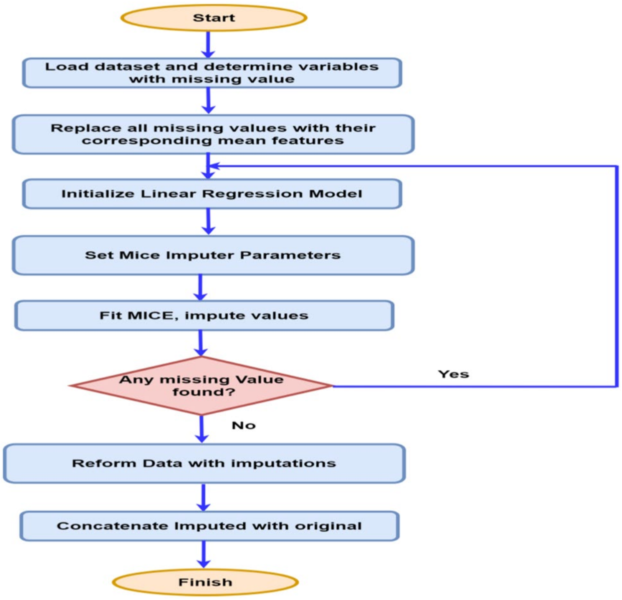

Start
Load dataset and determine variables with missing value
Replace all missing values with their corresponding mean features
Initialize Linear Regression Model
Set Mice Imputer Parameters
Fit MICE, impute values
Any missing Value found?
Yes
No
Reform Data with imputations
Concatenate Imputed with original
Finish

Fig. 1 MICE imputation flowchart

for the observation with the missing value, and then substituting those values in the associated characteristics.

Nevertheless, statistical imputation methods have the potential to provide biased or unrealistic findings. Consequently, this approach leads to a reduction in the diversity of the dataset and exhibits subpar performance. Furthermore, this dataset comprises 24 distinct characteristics, with certain ones being categorical in nature. Hence, there are two approaches that can be employed to fill in the missing value in this dataset. For categorical variables such as appetite, pedal edema, anemia, etc., the missing values can be substituted with either a constant value or the statistical measures (mean, median, or mode) of the corresponding columns. Initially, this article substituted categorical missing data with constant values referred to as missing as it can yield superior outcomes compared to statistical methods such as mode.

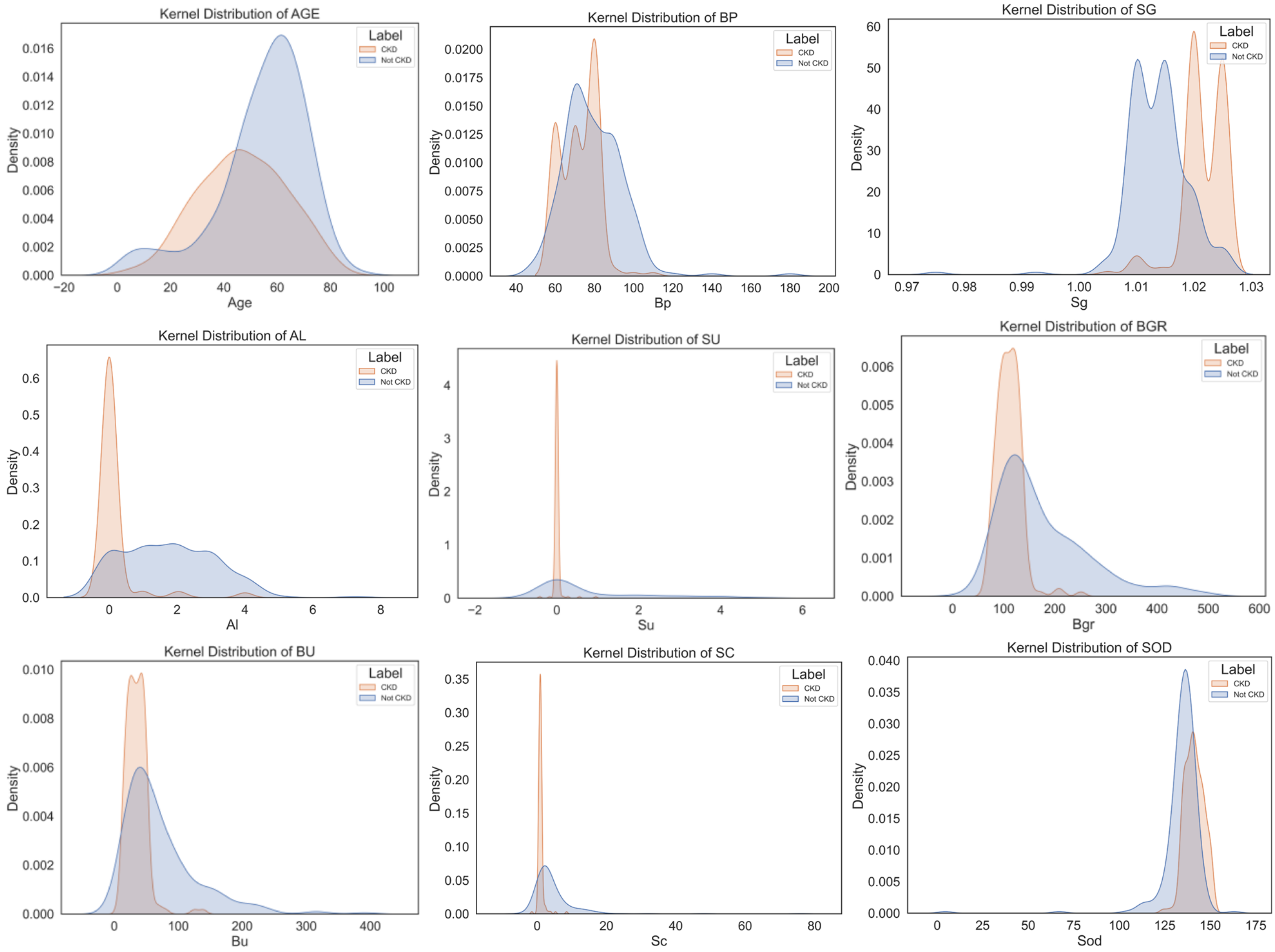
Consequently, a distinct category has been identified for each feature that contains a missing value. For example, hypertension can be classified into two categories: yes or no. However, when a basic imputer is employed with a constant, hypertension is divided into three categories: yes, no, and missing. Multiple Imputations by Chain Equation (MICE) is an advanced technique used to handle missing data sets. In this study, we have utilized the MICE method to fill in the missing values for each feature, with the exception of the categorical features. Figure 1 displays the flowchart illustrating the MICE imputation approach. The MICE technique involves designating one feature column with missing data as the output, while the remaining feature columns are assigned as inputs. Subsequently, the regression model is employed to forecast the output data, and this procedure is carried out in an iterative manner. During each iteration, any missing values in the dataset are replaced with other values from the dataset and the procedure continues until it convergences.

3.2.2 Rescaling of dataset

The entire dataset has undergone rescaling due to the presence of data stored on disparate scales. Ordinal and binary encoding techniques are used to transform category data into ordinal and binary values, respectively. Furthermore, we apply a process of standardization to the dataset to guarantee that every feature possesses an average of zero and a standard deviation of one. Furthermore, the kernel distribution estimation (KDE) is assessed by utilizing every individual sample value of each feature to describe the probability density function. The kernel distribution, similar to a histogram, constructs a function that represents the probability distribution of the sample data. This technique can effectively elucidate the range of values for each feature that are responsible for chronic kidney disease (CKD). While there is substantial overlap between CKD and NOTCKD samples, the visualization can effectively capture the most significant information present in the sample. The Kernel Density Estimation (KDE), as known as the Kernel Distribution Estimation is employed in data visualization to portray the fundamental distribution of the features as illustrated in Fig. 2.

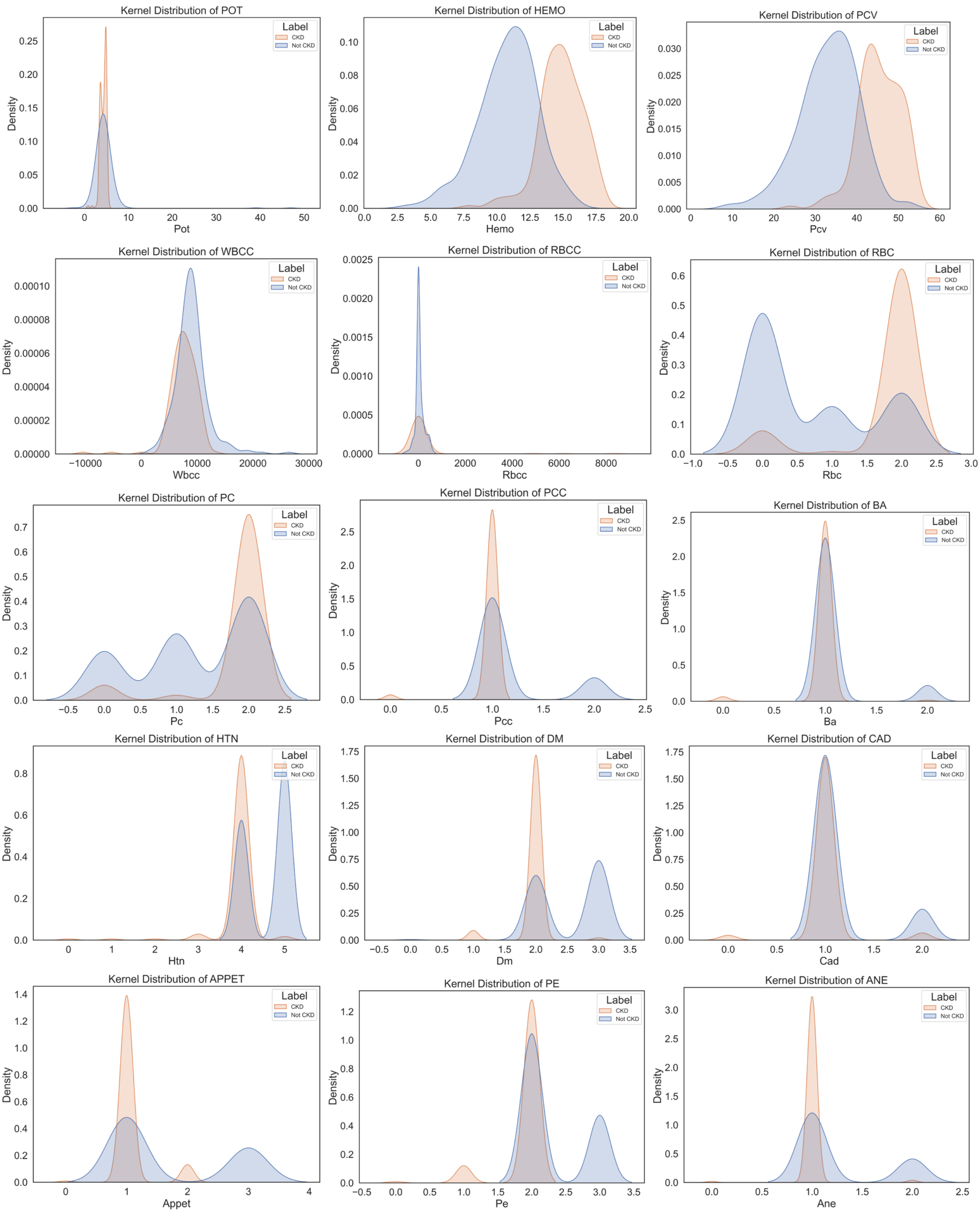
3.2.3 Equilibrate dataset

An imbalance has been noted in the dataset regarding the number of classifications, specifically CKD or NOTCKD. Imbalanced classifications pose a challenge in predictive modeling, leading to models that have low predictive accuracy. Two techniques, specifically oversampling and under-sampling, can be employed to address the issue of an imbalanced dataset. Several sophisticated over-sampling techniques have been suggested in previous research, with SMOTE being widely recognized as one of the most favored methods [54]. However, an alternative oversampling technique known as borderline-SMOTE [55] shown superior performance compared to SMOTE. Hence, this paper employs the borderline-SMOTE technique in place of the SMOTE technique. This article used the borderline SMOTE, where a Support Vector Machine (SVM) is employed instead of K-Nearest Neighbors (KNN) to generate synthetic instances of the minority class near the boundary separating the two classes in order to address the issue of imbalanced class distribution in a dataset. This method involves oversampling the data points located at the borderline between the minority and majority classes. Given the potential for misclassification in borderline circumstances, it is essential to establish the optimal decision boundary. Furthermore, SVM endeavors to create novel examples in the vicinity of the majority class, when the density of majority class instances is minimal. It has been noted that this technology has achieved better results than the SMOTE technique.



(a) Kernel density estimation of age, bp, sg, al, su, bgr, bu, sc, sod

Fig. 2 Kernel density estimation of the individual feature in terms of the output class



(b) Kernel density estimation of pot, hemo, pcv, wbcc, rbcc, rbc, pc, pcc, ba, htn, dm, cad, appet, pe, ane

Fig. 2 (continued)

3.3 Feature selection

Feature selection involves eliminating irrelevant characteristics to reduce the computational burden on the machine and improve the performance of the machine model [7–9]. Hence, feature selection is advantageous in enhancing the classifier’s performance and minimizing the execution time. Consequently, this study utilizes a proposed fusion-based feature selection strategy using PCA, FA and LDA to extract essential features to predict kidney patients.

3.3.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) [7] is a crucial technique for reducing the dimensionality of a dataset. It efficiently condenses a dataset with multiple dimensions into a smaller number of dimensions while retaining crucial information. In order to establish the optimal number of principal components, a method is utilized to guarantee that these components account for 95% of the variance by producing 18 top components as seen in Fig. 3, thereby encompassing the majority of the information included in the dataset.

Pseudo-code for the Principal Component Analysis (PCA)

Inputs:

- Dataset D with m samples and r features $D = \{d_1, d_2, \ldots, d_m\}$
- Target number of principal components n to retain.

Output:

- Reduced dataset D' with m samples and n principal components.
- Explained variance for each principal component

Process:

1. Standardize the dataset D to have a mean of zero and unit variance for each feature.

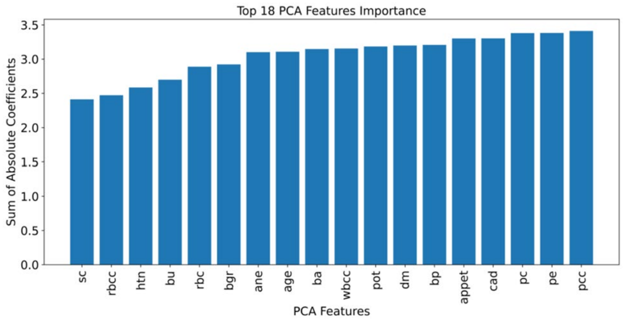


Fig. 3 Top 18 Components of PCA Feature selection

2. Compute the covariance matrix $\sum$ of the standardized dataset.

$$\sum = \frac{1}{m-1}(D-\overline{D})^T\left(\mathrm{D}-\overline{D}\right).$$

where $\overline{D}$ is the mean vector of the original dataset D.

3. Calculate the eigenvalues λ_i and eigenvectors E of the covariance matrix $\sum$.
4. Sort the eigenvectors by decreasing eigenvalues and select the top n eigenvectors.
5. Form the feature vector F by stacking the n eigenvectors.
6. Transform the original dataset D into the new subspace using the feature vector F to obtain the reduced dataset D':

$$D' = F^T.D$$

7. Calculate the explained variance for each principal component.
8. Return the reduced dataset D' and the explained variance.

3.3.2 Factor analysis (FA)

Factor analysis [7] is a technique used to reduce the number of features by uncovering underlying variables that are not directly observed but are inferred from the observed or manifest variables. This technique aims to extract the highest amount of common variation from the observed variables and combine it into a single score for further analysis. In general, extracting a large number of factors might result in undesirable consequences, while reducing the number of factors can decrease the amount of shared variance without any negative impacts. Hence, the careful selection of an optimal number of components is pivotal in the process of analysis. Common techniques for identifying the optimal number of elements include the eigenvalue approach, scree plot, Kaiser's criterion, and Jolliffe's criterion. We combined the scree plot and the Kaiser’s criterion as an enhanced FA technique to provide a more nuanced and accurate method for determining the number of factors and top 16 components were selected as seen in Fig. 4.

Pseudo-code for Factor Analysis (FA) with Scree Plot and Kaiser's Criterion

Inputs:

- Dataset D with m samples and r features $D = \{d_1, d_2, \ldots, d_m\}$
- Initial number of factors f.

Output:

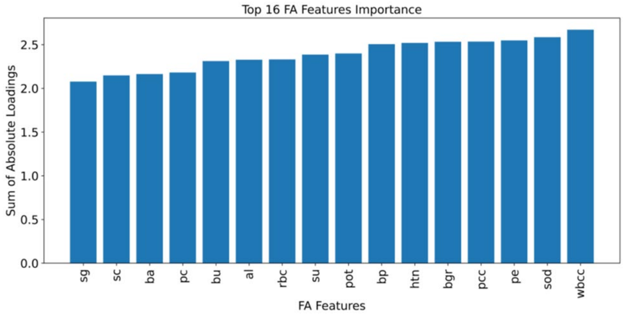


Fig. 4 Top 16 Components of FA Feature selection

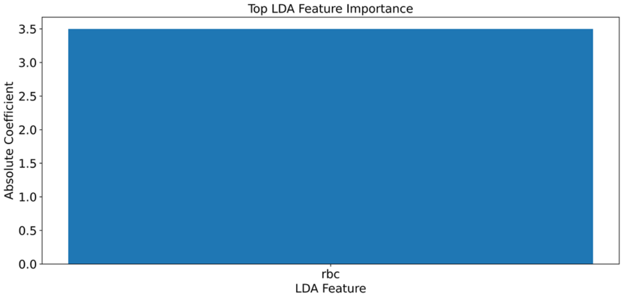


Fig. 5 Feature selection of LDA Component

- Reduced dataset D'.
- Number of significant factors n.

Process:

1. 1. Initial Factor Analysis:
   - Apply FA on D with f initial factors.
   - Compute eigenvalues for each factor: Eigenvalue = Noise Variance + Diagonal of Covariance (D)
2. Scree Plot Visualization:
   - Plot eigenvalues against factor indices and identify 'elbow point' to decide on significant factors.
   - Scree Plot Formula: Plot (Factor Index, Eigenvalues)
3. Kaiser’s Criterion Application:
   - For significant factor selection, determine n where eigenvalues are greater than the threshold (1)
     n = (Eigenvalues > 1)
4. Refined Factor Analysis:
   - Reapply FA on D using n factors and obtain a reduced feature space D'.
5. Explained Variance Calculation:
   - Calculate the proportion of variance explained by each factor.

     $$\text{Explained Variance} = \frac{\text{Sum of squared loadings for each factor}}{\text{Total Variance of D}}$$
6. Factor Loadings Analysis:
   - Examine how original features contribute to each factor.
   - Factor Loadings = Components of FA model

3.3.3 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a supervised method for reducing the dimensions of a dataset, primarily used for classifying datasets into two or more categories. The core objective of LDA is to compress an n-dimensional space into an m-dimensional one. Generally, the number of dimensions produced by LDA is fewer than the total number of classes in the dataset. In this technique, the newly projected data matrix is of a lower dimensionality, which aims to minimize variance within each class while maximizing variance between different classes. Each class is characterized by a unique dimension that sets it apart. In this research, since we have got two classes, therefore our LDA will be Class-1, which becomes 1 component for LDA as seen in Fig. 5.

Pseudo-Code for Linear Discriminant Analysis (LDA)

Inputs:

- Dataset D with m samples and r features $D=\{d_1,d_2,\ldots,d_m\}$.
- Class labels L corresponding to each sample in D.

Output:

- Transformed dataset D' in the reduced LDA space.

Process:

1. Calculate Class Means:
   - Compute the mean vectors for each class in L: $\overline{d}_k = \frac{1}{n_k}\sum_{d_i \in classk} d_i$, where n_k is the number of samples in class k.
2. Between-Class Scatter Matrix S_B:

- Calculate the between-class scatter matrix: $S_B = \sum_{k=1}^{k} n_k(\overline{d}_k - \overline{d})\ (\overline{d}_k - \overline{d})^T$, where $\overline{d}$ is the overall mean of the dataset, and K is the number of classes.

3. Within-Class Scatter Matrix S_w:

   - Calculate the within-class scatter matrix: $S_w = \sum_{k=1}^{k} \sum_{d_i \in class\,k} (d_i - \overline{d}_k)\ (d_i - \overline{d}_k)^T$

4. Compute the LDA Criteria:

   - Solve the eigenvalue problem for the matrix $S_W^{-1} S_B$ to obtain the eigenvectors.
   - These eigenvectors define the LDA components.

5. Select Top Components:

   - Choose eigenvectors with the highest eigenvalues to form a transformation matrix W.

6. Transform Dataset:

   - Project D onto the new LDA space using W:

     $D' = D \times W$

7. Output Transformed Dataset:

   - The transformed dataset $D\prime$ is now in a space where classes are linearly separable.

3.4 Proposed fusion-based feature selection method

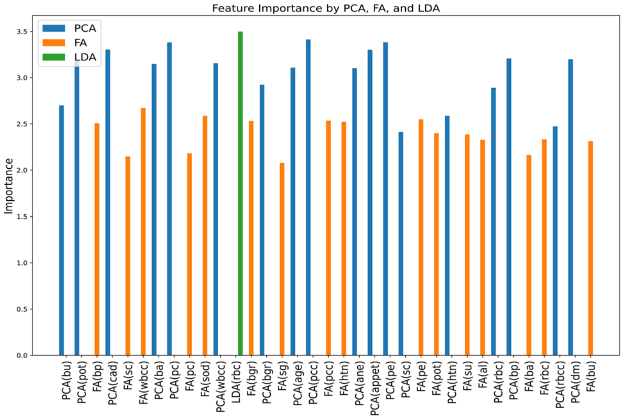


Fig. 6 Top 35 components of proposed fusion-based feature selection (FFS)

When analyzing the Chronic Kidney Disease (CKD) dataset, a novel approach called Fusion-based Feature Selection involves combining features derived from Principal Component Analysis (PCA), Factor Analysis (FA), and Linear Discriminant Analysis (LDA) in a concatenated manner. This approach utilizes the synergistic advantages of these three separate feature selection processes, resulting in a resilient and multifaceted feature set. The concatenated approach commences by separately implementing Principal Component Analysis (PCA), Factor Analysis (FA), and Linear Discriminant Analysis (LDA) on the standardized Chronic Kidney Disease (CKD) dataset. Every technique retrieves a distinct set of characteristics: Principal Component Analysis (PCA) is a method that identifies the main components that explain the most variation in the data. Factor Analysis (FA) reveals underlying factors that explain shared variances. Linear Discriminant Analysis (LDA) identifies discriminants that provide the best separation between different classes, which is particularly helpful in discriminating between occurrences of Chronic Kidney Disease (CKD) and instances of non-CKD. The essence of this approach is based on horizontally combining the output features obtained from PCA, FA, and LDA, thereby a total of top 35 feature components for the different feature selection analyzed as seen in Fig. 6. Each observation in the dataset is combined to create a feature vector that includes PCA components, FA factors, and LDA discriminants.

A concatenated feature set combines the variance, latent structures, and class-specific properties of the data, while also including unique perspectives from each approach, so enriching the dataset. The use of this comprehensive set of features offers numerous benefits, especially in intricate datasets such as CKD where multidimensional insights are essential. It enhances the ability to comprehend the data in a more detailed and sophisticated manner, hence strengthening the possibility for accurate categorization and forecasting. The wide range of characteristics, which can be focused on either variance or class separability, provides researchers and data scientists with a robust set of tools to discover concealed patterns, create well-informed forecasts, and find crucial elements that contribute to CKD. This strategy is a comprehensive and purposeful combination of feature selection strategies, setting a standard for advanced data analysis in medical research. In addition, in order to prevent bias and overfitting in our experimental results, we employ a random oversampling technique.

Pseudo-Code for the Propose Fusion-based Feature Selection

Inputs:

- Dataset D with m samples and r features.
- Class labels L for LDA.
- Number of components n_{PCA} for PCA, n_{FA} for FA, and n_{LDA} for LDA.

Output:

- Concatenated feature set from PCA, FA, and LDA.

Process:

1. Preprocessing:
   - Standardize the dataset D.
2. PCA Component Extraction:
   - Apply PCA on D to extract n_{PCA} principal components.
   - Let D_{PCA} be the PCA component matrix.
3. FA Component Extraction:
   - Perform FA on D to extract n_{FA} factors.
   - Let D_{FA} be the FA component matrix.
4. LDA Component Extraction:
   - Apply LDA on D using class labels L to extract n_{LDA} discriminants.
   - Let D_{LDA} be the LDA component matrix.
5. Concatenation of Components:
   - Concatenate D_{PCA}, D_{FA} and D_{LDA} horizontally.
   - Let D be the resulting concatenated feature set.
6. Output:
   - Return D as the integrated feature set from PCA, FA, and LDA.

3.5 Models for categorization

A classifier or classification model is essential for accurately predicting different diseases based on the patient's medical history. The classifier undergoes training using a set of training data, and subsequently, it is evaluated by classifying the corresponding target variables. This article utilized different single and ensemble classification algorithms to differentiate between the patient's condition of Chronic Kidney Disease (CKD) and non-CKD.

3.5.1 Single models

i. Support Vector Machine (SVM)

The Support Vector Machine (SVM) [56] is an effective classifier that identifies the ideal hyperplane to accurately split data into different classes. The support vector machine (SVM) is highly efficient in high-dimensional spaces and exhibits versatility by employing several kernel functions, such as linear, polynomial, and radial basis function (RBF). SVM, or Support Vector Machine, is a predictive model that can accurately represent intricate connections between symptoms and disease outcomes. It achieves this by optimizing the distance between the data points that are closest to the decision border, known as support vectors. The process of maximizing the margin in SVM enhances its robustness, reducing the likelihood of overfitting, particularly in situations when the number of features is significantly more than the number of observations.

ii. K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) [57] algorithm is a type of instance-based learning. It predicts the outcome for a new instance by considering the majority vote of its 'k' nearest points in the feature space. The non-parametric method is frequently selected because to its simplicity and efficacy in classifying tasks that involves a non-linear relationship between the feature space and the output. In the case of Chronic Kidney Disease (CKD), the K-Nearest Neighbors (KNN) algorithm can be employed to detect patterns by comparing the similarity of symptoms among patients. The proximity of cases is utilized as an indicator of the probability of the disease being present.

iii. Decision Tree (Tree)

Trees are a non-linear predictive modeling technique that divides the data into subsets based on the input feature values by recursive partitioning. Every node in the tree represents a characteristic in the dataset, and each branch represents a decision rule that leads to a leaf node, which corresponds to a predicted result. Decision Trees provide an inherent intuitiveness and their decisions are readily interpretable, rendering them highly effective for providing clinical decision support in the prediction of Chronic Kidney Disease (CKD). These models have the ability to process both numerical and categorical data and can effectively capture intricate relationships between symptoms.

iv. Quadratic Discriminant Analysis (QDA)

QDA is a statistical technique employed to classify datasets including two or more classes. This method assumes that the data from each class is sampled from a Gaussian distribution. It uses quadratic decision surfaces to distinguish between the classes. This approach is especially beneficial when there is a non-linear relationship between the independent factors and the dependent variable. QDA can be advantageous in predicting CKD when the classes have noticeable differences in their variance–covariance structures. This allows the model to adapt to the unique characteristics of the data's distribution.

3.5.2 Ensemble models

i. Extreme gradient boosting (XGBoost)

XGBoost is a powerful computational ML model that is based on gradient boosting. It has the advantage of requiring less computing time compared to traditional gradient boosting [58]. Furthermore, XGBoost incorporates regularization terms, resulting in superior performance compared to the gradient boosting method. Furthermore, XGBoost possesses the capability to automatically manage missing values and process larger datasets that exceed the memory capacity. In addition, it has the capability to do parallel operations by utilizing column sub-sampling techniques.

ii. Light gradient boosting (LightGBM)

LightGBM technique combines gradient-based one-side sampling (GOSS) with exclusive feature bundling (EFB) [58]. Gradient-based One-Side Sampling (GOSS) is utilized to conduct gradient-based subsampling of the training data. This technique effectively reduces overfitting and accelerates the training process. Significantly, the algorithm employs a leaf-wise growth technique to construct the decision trees, resulting in a reduction in the number of tree nodes and an enhancement in algorithm efficiency. Consequently, LightGBM exhibits a shorter execution time compared to other ensemble approaches.

iii. Voting

Majority voting is a prevalent and widely recognized technique among the eight ensemble methods. In a voting system, each model in the ensemble is trained autonomously on a subset of the training data or utilizing a distinct algorithm or combination of hyperparameters. The prediction is thereafter exhibited by adjusting the classifier weights based on the majority of votes from the classifier [59]. This paper examines the utilization of the support vector machine (SVM) and k-nearest neighbor (KNN) classification algorithms in a majority voting ensemble.

iv. Bagging (Bag)

Bagging is a machine learning technique used for ensemble learning, where the term “Bagging” is actually an abbreviation for “Bootstrap Aggregating”. Bagging involves training each model in the ensemble on a randomly selected subset of the training data, with replacement. The method utilizes bootstrap sampling to generate data subsets for training each base classifier separately. There is a possibility that certain data points from the original training set may be duplicated in each bootstrap sample. The class that receives the highest level of popularity is determined once the algorithm has been trained using all bootstrap samples.

v. Adaptive boosting (Ada)

Adaptive boosting [58] is a method that combines multiple weak classifiers to create a strong classifier. The strategy, introduced by Freund and Schapire, utilizes decision tree stumps as weak algorithms in a consecutive manner. Each subsequent algorithm corrects the inaccurately predicted output of the previous learners. The weight of each sample is allocated during every phase of the training period to execute this procedure. To enhance the performance of the learners that follow, Ada typically focuses on minimizing the misclassification error. The records are selected to optimize the weighting of training samples for the subsequent classifier. It selects a learner that minimizes the error rate when classifying data during the training phase.

vi. Gradient boosting decision tree (GBDT)

GBDT is a method that combines multiple weak learners to generate powerful learners for classification and regression tasks. This method employs an iterative procedure to incrementally construct decision trees, aiming to minimize the errors of the previous trees. The ultimate goal is to create a powerful learner by combining these trees [60]. This technique employs an ensemble of many weak learners to generate robust learners for classification and regression tasks. This method uses an iterative procedure to sequentially construct decision trees, with the aim of minimizing errors from previous trees. Ultimately, the trees are combined to create a powerful learner. When incorporating new models, it utilizes a gradient descent approach to minimize loss.

vii. Random Forest (RF)

The RF algorithm is an ensemble learning technique that builds many decision trees during training and outputs the most frequent class (classification) among the individual trees. Random Forest (RF) incorporates randomness into the model by employing bootstrapping to sample the data and selecting a subset of characteristics at each split while constructing the trees. The incorporation of randomization in the model aids in the generation of a collection of varied trees, hence, mitigating the likelihood of overfitting and enhancing the model’s resilience in predicting Chronic Kidney Disease (CKD). It excels at managing extensive datasets with several features, capturing intricate relationships between symptoms without requiring the reduction of features.

viii. Stacking

STACK refers to the act of arranging objects or items in a vertical or horizontal manner, one on top of another, in a neat and organized fashion. Stacking is an ensemble method that involves performing classification and regression operations in two stages [60]. During the initial stage, classifiers undergo training using the provided data and provide predictions that serve as input for the subsequent classifier. In the last phase, all projected results are treated as the input for the new classifier and assessed as the ultimate output.

3.5.3 SHAP interpretation model

The objective of this work is to extract the complexities of machine learning predictions and convert them into practical insights for Chronic Kidney Disease (CKD) by employing Explainable Artificial Intelligence (XAI). Our goal is to use machine learning methods to analyze the CKD dataset and understand the complex relationship between symptoms and CKD. Our method utilizes machine learning algorithms, including the single model and ensemble models that have been carefully optimized to classify CKD data accurately. In order to assure the strength and avoid overfitting [60, 61], the models undergo thorough validation using unseen data. Performance indicators such as accuracy, precision, f1-score and recall are carefully examined during this process.

After the validation is successfully completed, we proceed to utilize the SHapley Additive exPlanations (SHAP) approach in order to clarify and understand the decision-making processes of the models. SHAP values provide a detailed perspective on the contributions of each characteristic, illuminating the importance of each symptom in predicting CKD. The algorithms analyze the combined feature selection and train on a carefully selected dataset to predict chronic kidney disease outcomes. SHAP possesses significant interpretive capabilities, allowing us to analyze the models' predictions and the influence of specific features. The overall proposed architecture is illustrated in Fig. 7 and with employing this strategy, the prediction accuracy is enhanced and the models become more transparent, hence promoting trust and comprehension in the diagnostic predictions of CKD.

4 Experimental setup

This section will explain the experimental setup executed in this paper which includes the environment setup, dataset split, hyperparameter tweaking and evaluation metrics.

4.1 Environment setup

The proposed model was compiled used the python programming language with machine learning libraries on Windows 11 operating system. The hardware system operates with 64 GB RAM and 8 GB GPU, 11th Generation Intel core i7-11800H at 2.30HHz.

4.2 Dataset split

For each experiment, the performance of the proposed model was assessed by utilizing the train-test split and the cross-validation method. In every experimental setting, 80% of the entire data was utilized as train set and the remaining 20% for train set both during the train-split and cross-validation method. Cross-validation is a statistical technique employed to assess the proficiency of machine learning models. It is frequently employed to ensure that the model is resilient and operates effectively on data that it has not been trained on. We partitioned our dataset into five equitably sized subsets, utilizing four of them for training our model and reserving the remaining subset for validation purposes.

4.3 Hyper-parameter tweaking

Configuring the hyperparameters of each algorithm based on the CKD dataset enables us to optimize the algorithms, making them more adaptable and impactful. Nevertheless, adjusting the hyperparameter for each model is always discretionary. In addition, the performance of the majority classifier is dependent on the hyperparameters. Optimal solutions and outcomes can be achieved by selecting effective parameters for various applications.

Utilizing all parameters in the model will result in increased complexity and might slow down computations. In order to obtain the optimum model, it is crucial to conduct a focused search for a limited set of hyperparameters for each method. This work utilized a randomized search cross-validation approach to adjust the crucial hyperparameters of each classifier, which is considered superior to the conventional grid search technique. Moreover, the prediction of chronic kidney disease (CKD) was seen using the optimal combination of hyperparameters for each classifier. Table 3 presents comprehensive information regarding the primary hyperparameters of each classifier for binary classification of CKD.

4.4 Evaluation metrics for the model

Numerous evaluation metrics can assess the accuracy of a model's predictions. The performance evaluation metrics for the prediction of CKD include accuracy, precision, recall, F-measure, AUC-ROC, and confusion matrix. Below describes each performance evaluation metrics:

a. Accuracy.

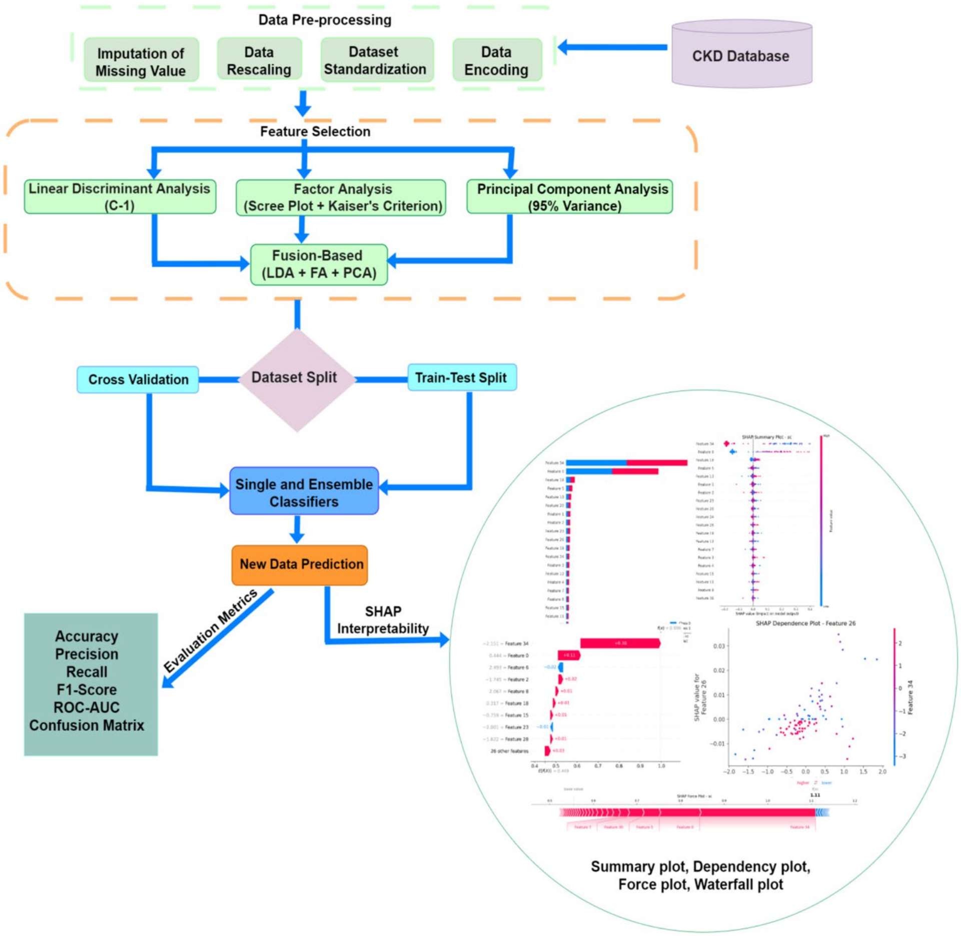

Data Pre-processing
Imputation of Missing Value
Data Rescaling
Dataset Standardization
Data Encoding
CKD Database
Feature Selection
Linear Discriminant Analysis (C-1)
Factor Analysis (Scree Plot + Kaiser's Criterion)
Principal Component Analysis (95% Variance)
Fusion-Based (LDA + FA + PCA)
Cross Validation
Dataset Split
Train-Test Split
Single and Ensemble Classifiers
New Data Prediction
Evaluation Metrics
SHAP Interpretability
Accuracy
Precision
Recall
F1-Score
ROC-AUC
Confusion Matrix
SHAP Dependence Plot - Feature 26
Summary plot, Dependency plot, Force plot, Waterfall plot

Fig. 7 Proposed fusion-based feature selection interpretability machine learning model (FFS-IML)

Accuracy quantifies the proportion of properly predicted instances from a specific class, relative to the total number of samples. It is often calculated using Eq. (1).

$$\text{Accuracy} \;=\frac{TP+TN}{TP+FP+TN+FN} \tag{1}$$

b. Recall.

Recall is determined by calculating the ratio of correctly identified positive cases (TP) to the total number of genuine positive cases (TP + FN). Equation (2) is employed to compute the recall.

$$\text{Recall} \;=\frac{TP}{TP+FN} \tag{2}$$

c. Precision.

Precision is the ratio of true positive cases (TP) to the sum of true positive and false positive cases (TP + FP). The calculation can be performed using Eq. (3).

$$\text{Recall} \;=\frac{TP}{TP+FP} \tag{3}$$

d. F-measure.

The F-measure is computed by combining precision and recall. The F1 score is a mathematical metric that represents the weighted average of precision and recall. The F-measure is denoted by Eq. (4).

$$\text{F1}-\text{score} \;=2*\frac{Precision*Recall}{Precision+Recall} \tag{4}$$

e. Confusion matrix.

Confusion matrix is a technique used to precisely evaluate the effectiveness of a classification model. It displays the relationship between the actual and predicted classes of the target variable.

f. Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

The plot illustrating the relationship between the True Positive Rate (TPR) and False Positive Rate (FPR) is commonly referred to as the Receiver Operating Characteristic (ROC) curve (FPR) which showcases the efficacy of a categorization model. The region beneath the receiver operating characteristic (ROC) curve is sometimes referred to as the area under the curve (AUC). The AUC-ROC statistic quantifies the discriminatory power of a classification model by evaluating its performance

Table 3 CKD prediction model hyperparameters for randomized grid search

Algorithm	Hyperparameter Explanation	Randomized Grid Search Range
SVM	C (Regularization strength), kernel (Type of kernel), gamma (Kernel coefficient), probability (Probability estimates)	C: [0.1, 10], kernel: [‘linear’, ‘poly’, ‘rbf’], gamma: [0.001, 0.1], probability: [True]
KNN	n_neighbors (Number of neighbors)	n_neighbors: [1, 10]
Random Forest	n_estimators (Number of trees), max_features (Max features for split), min_samples_split (Min samples to split), min_samples_leaf (Min samples at leaf), max_depth (Tree depth), bootstrap (Use bootstrap)	n_estimators: [100, 500], max_features: [‘sqrt’, ‘log2’], min_samples_split: [2, 10], min_samples_leaf: [1, 4], max_depth: [None, 10], bootstrap: [True, False]
QDA	priors (Class priors), reg_param (Regularization parameter)	priors: [None], reg_param: [0.0, 0.5]
Decision tree	criterion (Split quality measure), min_samples_split (Min samples to split), max_depth (Tree depth)	criterion: [‘gini’, ‘entropy’], min_samples_split: [2, 10], max_depth: [None, 10]
Voting classifier	estimators (Classifiers list), voting (Voting strategy)	estimators: [‘svm’, ‘knn’, ‘rf’, ‘qda’], voting: [‘hard’, ‘soft’]
Bagging classifier	base_estimator (Base estimator), n_estimators (Number of estimators), bootstrap (Use bootstrap), bootstrap_features (Feature bootstrapping)	base_estimator: [‘tree’], n_estimators: [10, 100], bootstrap: [True, False], bootstrap_features: [False]
AdaBoost classifier	base_estimator (Base estimator), n_estimators (Number of estimators), learning_rate (Contribution shrinkage)	base_estimator: [‘tree’], n_estimators: [50, 500], learning_rate: [0.01, 1]
Gradient boosting classifier	n_estimators (Number of stages), learning_rate (Contribution shrinkage), max_depth (Tree depth), min_samples_leaf (Min samples at leaf)	n_estimators: [100, 500], learning_rate: [0.01, 0.1], max_depth: [3, 10], min_samples_leaf: [1, 5]
LightGBM	n_estimators (Number of trees), max_depth (Max depth), learning_rate (Learning rate), min_child_weight (Child weight requirement)	n_estimators: [100, 1000], max_depth: [3, 8], learning_rate: [0.01, 0.1], min_child_weight: [1, 5]
XGBoost	n_estimators (Number of trees), max_depth (Max depth), learning_rate (Learning rate), min_child_weight (Child weight requirement)	n_estimators: [100, 1000], max_depth: [3, 10], learning_rate: [0.01, 0.1], min_child_weight: [1, 5]
Stacking classifier	estimators (Base estimators), final_estimator (Final estimator)	estimators: [‘svm’, ‘knn’, ‘rf’], final_estimator: [‘rf’, ‘qda’]

across different threshold values. A model with a higher AUC value indicates better performance.

5 Experimental results and analysis

This section will discuss the output result of each classifier while considering the different feature section and even when there is no application of any feature selection. The performance of the data split method- the train-test split and the cross validation methods will be checkmated. Overall, the performance of each classifier for the prediction of CKD will be analyze with the various evaluation metrics as stated in Sect. 4.3.

5.1 Absence of feature selection

In this scenario, we conducted an experiment to assess each classifier for the prediction of CKD without any feature selection processes. Table 4 depicts the outcomes of each classification model utilizing all features of the provided dataset. The table clearly indicates that for the single models aspect, QDA algorithm achieved a greater percentage within three metrics which include, accuracy of 96.84%, recall of 95.92% and F1-score of 96.90% with a lesser computational time when compared to others. Additionally, both KNN and SVM achieved 100% in AUC-ROC whereas KNN achieved the highest performance value in Precision of 100%. Generally, it is important to note that the ensemble model, RF classifier exhibits the lowest performance of 87.04% when compared to other classifiers even with the single model classifiers. Furthermore, for the fivefold cross-validation for the different classifiers exhibits a satisfactory performance when compared to the train-test split method. The voting classifier achieves the highest average accuracy over all folds with 98.95% accuracy, thus, suggesting a strong level of consistency in our model's predictions for Chronic Kidney Disease (CKD). In addition, the standard deviation was low, measuring at 1.29%, indicating that there was limited variance in accuracy among the different folds as seen in Table 4.

5.2 Factor analysis (FA) based feature selection

In this context, we conducted an experiment to evaluate each classifier's ability to predict CKD utilizing factor analysis feature selection. In the factor analysis, 16 latent factors were found with the utilization of both scree plot and Kaiser's criterion. Comparing Tables 4 and 5, it is noticeable that the utilization of Factor analysis feature selection is quiet higher than without employing feature selection. Table 5 illustrates the results of each classification model using all the features from the given dataset. The chart clearly demonstrates that the SVM algorithm outperformed other algorithms in terms of four evaluation metrics for both single and ensemble models with accuracy of 98.95%, recall of 100.00%, F1-score of 98.99% and AUC-ROC of 100.00% and also the cross-validation exhibits an average accuracy of 98.95% with standard deviation of 1.29% across all folds. More so, both stack and voting model exhibit the same values across all of the evaluation metrics.

5.3 Linear discriminant analysis (LDA) based feature selection

During the feature selection phase, we experimented on utilizing linear discriminant analysis (LDA) so as to

Table 4 Classifiers' performance without any feature selection

Feature selection	Model type	Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-S (%)	AUC (%)	Time (sec)	Cross-Val Acc (Mean ± std dev)
No feature selection	Single models	KNN	94.74	100.00	89.80	94.62	100.00	0.141	92.11 ± 4.40
		QDA	96.84	97.92	95.92	96.90	99.82	0.008	95.79 ± 2.11
		SVM	95.78	97.87	93.88	95.83	100.00	0.070	98.95 ± 2.11
		Tree	88.42	89.58	87.76	88.65	97.93	0.041	91.05 ± 5.42
	Ensemble models	Ada	91.58	93.62	89.80	91.67	99.05	0.284	95.26 ± 4.53
		Bag	90.53	88.46	93.88	91.09	99.71	4.382	95.79 ± 3.57
		GBD	92.63	90.38	95.92	93.07	99.66	16.135	95.79 ± 3.57
		LGB	91.58	88.68	95.92	92.16	99.80	0.197	95.79 ± 2.11
		RF	90.53	87.04	95.92	91.26	99.70	2.691	96.84 ± 3.07
		Stack	95.79	95.92	95.92	95.92	100.00	2.904	98.42 ± 2.11
		Voting	96.84	100.00	93.88	96.84	100.00	5.431	**98.95 ± 1.29**
		XGB	91.58	91.84	91.84	91.84	99.7	5.69	93.68 ± 3.57

Table 5 Classifiers' performance with FA based feature selection

Feature selection	Model type	Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-S (%)	AUC (%)	Time (sec)	Validation Acc (Mean ± std dev)
Factor analysis (FA)	Single models	KNN	98.95	100.00	97.96	98.97	100.00	0.126	97.37 ± 1.66
		QDA	96.84	96.00	97.96	96.97	98.3	0.011	94.74 ± 1.66
		SVM	98.95	98.00	100.00	98.99	100.00	0.067	**98.95 ± 1.29**
		Tree	93.68	97.78	89.79	93.62	98.9	0.031	93.68 ± 3.16
	Ensemble models	Ada	93.68	97.78	89.79	93.62	100.00	0.074	93.68 ± 2.68
		Bag	95.79	95.92	95.92	95.92	100.00	4.771	96.32 ± 2.68
		GBD	95.79	95.92	95.92	95.92	100.00	18.783	96.84 ± 3.07
		LGB	98.95	100.00	97.6	98.97	100.00	0.165	96.84 ± 3.07
		RF	93.68	92.16	95.92	94.00	100.00	2.408	95.79 ± 2.68
		Stack	98.95	100.00	97.96	98.97	100.00	2.690	98.42 ± 2.11
		Voting	98.95	100.00	97.96	98.97	100.00	5.943	97.89 ± 1.97
		XGB	97.89	97.96	97.96	97.96	99.7	5.390	96.32 ± 2.68

project feature space into a low-dimensional space and from our binary classification task, the resultant number of LDA is limited to a single component (1). With this application, most of the models achieved similar performance values in the evaluation metrics as illustrated in Table 6. Additionally, both SVM and GDB achieved the same performance of average accuracy of 97.89% and standard deviation 0f 1.05% during the cross-validation stage among the different folds. Lastly, it is observed the performance of LDA is quite higher than just using FA or no feature selection.

5.4 Principal component analysis (PCA) based feature selection

We conducted another feature selection using the principal component analysis (PCA) and it can be seen in Table 7 that the evaluation metrics have greater performance as compared to other two feature selection as previously mentioned. To gain an explained variance ratio of 95%, 18 components were used and this achieved a tremendous result in all of the evaluation metrics. An accuracy of 98.95% was exhibited across both single and ensemble model, which include KNN, GBD, LGB and stack and also 100% was achieved by most models across the precision, recall and AUC metrics. More so, the cross validation achieved the highest accuracy of 98.95% with a standard deviation of 1.29 for the five-fold measure using the voting classifier.

5.5 Fusion-based feature selection

For this experiment, we analyzed the proposed fusion-based feature selection that includes the concatenation of PCA, LDA and FA. The purpose of this fusion was to utilize the

Table 6 Classifiers' performance with LDA feature selection

Feature selection	Model type	Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)	Time	Validation Acc (Mean ± std dev)
Linear discriminant analysis (LDA)	Single models	KNN	97.89	97.96	97.96	97.96	99.8	0.011	97.37 ± 1.66
		QDA	97.89	97.96	97.96	97.96	99.8	0.009	97.37 ±00
		SVM	97.89	97.96	97.96	97.96	99.8	0.034	97.89 **± 1.05**
		Tree	95.79	94.12	97.96	96.00	95.70	0.009	96.32 ± 2.68
	Ensemble models	Ada	95.79	94.12	97.96	96.00	95.60	0.029	95.79 ± 2.11
		Bag	96.84	96.00	97.96	96.97	99.6	3.005	95.79 ± 2.11
		GBD	97.89	97.96	97.96	97.96	99.7	1.705	97.89 **± 1.05**
		LGB	97.89	97.96	97.96	97.96	99.8	0.168	97.37 ± 00
		RF	97.89	97.96	97.59	97.96	99.6	1.410	96.84 ± 1.05
		Stack	97.89	97.96	97.96	97.96	99.2	1.453	97.37 ± 1.66
		Voting	97.89	97.96	97.96	97.96	99.8	3.52	97.37 ± 1.66
		XGB	97.89	97.96	97.96	97.96	99.8	0.276	97.37 ± 1.66

Table 7 Classifiers' performance with PCA feature selection

Feature selection	Model type	Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-S (%)	AUC (%)	Time (sec)	Validation Acc (Mean ± std dev)
Principal component analysis (PCA)	Single models	KNN	**98.95**	**100.00**	97.96	98.97	**100.00**	0.186	96.84 ± 1.97
		QDA	94.74	94.00	95.92	94.95	96.6	**0.012**	97.37 ± 00
		SVM	95.79	92.45	**100.00**	96.08	99.9	0.127	98.42 ± 1.29
		Tree	95.79	97.87	93.88	95.83	95.9	0.032	97.37 ± 1.66
	Ensemble models	Ada	95.79	97.87	93.88	95.83	95.9	0.059	98.42 ± 1.29
		Bag	96.84	97.92	95.92	96.91	99.8	3.495	97.37 ± 1.66
		GBD	**98.95**	98.00	**100.00**	**98.99**	99.8	8.964	97.37 ± 1.66
		LGB	**98.95**	98.00	**100.00**	**98.99**	99.9	0.324	97.89 ± 1.97
		RF	96.84	96.00	97.96	96.97	99.8	0.127	98.42 ± 1.29
		Stack	**98.95**	**100.00**	97.96	98.97	**100.00**	3.251	97.89 ± 1.97
		Voting	96.84	96.00	97.96	96.97	99.90	5.912	**98.95 ± 1.29**
		XGB	95.79	97.87	93.88	95.83	99.9	0.582	97.37 ± 1.66

advantages of each method, creating a comprehensive feature space that effectively manages variation, hidden connections, and the capacity to distinguish between different classes. PCA was used to maximize variance, FA was used to reveal underlying factors, and LDA was used to optimize class discrimination. These techniques combined to provide a strong and enhanced collection of features. The comparison analysis conducted on several classifiers yielded valuable insights as seen in Table 8.

The SVM classifier exhibited outstanding performance, attaining a perfect score of 100% in all criteria of the metrics, while also demonstrating remarkable computing economy. Similarly, ensemble models such as LGB, RF, Voting, and XGB demonstrated impeccable scores of 100% across all criteria. The results highlight the capability of ensemble models to utilize intricate, combined feature spaces to improve forecast accuracy.

On the other hand, individual models such as KNN and QDA showed impressive performance. KNN achieved flawless precision, while QDA stood out in terms of recall. The Decision Tree classifier demonstrated a balanced performance rate, affirming its trustworthiness. The proposed feature selection overcomes the inherent constraints of individual feature selection techniques. Although PCA primarily emphasizes variance, it may disregard class-specific characteristics that are essential for classification purposes. Factor analysis, while skilled at uncovering underlying patterns, may overlook important features of variability necessary for accurately representing the data. LDA prioritizes maximizing the distinction between classes, although it may not fully capture the overall variability present in the data. By

Table 8 Classifiers' Performance with Proposed Fusion-based Feature Selection

Feature selection	Model type	Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-S (%)	AUC (%)	Time (secs)	Validation Acc (Mean ± std dev)
PCA + LDA + FA	Single models	KNN	97.89	100.00	95.92	97.92	100.00	0.146	96.84 ± 1.05
		QDA	97.89	96.08	100.00	98.00	97.7	0.033	96.84 ± 1.05
		SVM	100.00	100.00	100.00	100.00	100.00	0.162	97.89 ± 1.05
		Tree	97.89	97.96	97.96	97.96	97.9	0.050	97.89 ± 1.05
	Ensemble models	Ada	97.89	97.96	97.96	97.96	97.9	0.067	97.89 ± 1.97
		Bag	98.95	100.00	97.96	98.97	100.00	3.858	99.47 ± **1.05**
		GBD	98.95	100.00	97.96	98.97	100.00	38.330	98.95 ± 1.29
		LGB	100.00	100.00	100.00	100.00	100.00	0.159	99.47 ± **1.05**
		RF	100.00	100.00	100.00	100.00	100.00	2.681	98.95 ± 1.29
		STACK	98.95	100.00	97.96	98.97	100.00	2.892	98.95 ± 1.29
		Voting	100.00	100.00	100.00	100.00	100.00	5.912	98.42 ± 2.11
		XGB	100.00	100.00	100.00	100.00	100.00	0.504	97.89 ± 1.05

integrating these methods, we create a feature space that accurately captures the variability and fundamental organization of the data, while simultaneously being optimized for the classification objective and dependable medical diagnostics; nevertheless, there is also need to understand the interpretability of the models and the features which will be elucidated in Sect. 5.7.

Figure 8 display the ROC-AUC results for our classifiers exhibit remarkable efficacy with the proposed fusion-based FS, with several models like KNN, SVM, Bag, GBD, LGB, RF, STACK, Voting, and XGB achieving the pinnacle of performance with a 100% AUC score. This level of accuracy is indicative of an extraordinary discriminative capacity, suggesting that the models are highly adept at distinguishing between the positive and negative classes. Such high AUC scores, typically rare in complex real-world scenarios, reflect the potent impact of advanced preprocessing and feature selection techniques, particularly the Fusion-based Feature Selection method applied in this study. Meanwhile, classifiers like QDA, Tree, and Ada achieved an AUC scores between 97.7 and 97.9%, also displayed commendable performance, confidently identifying the correct class labels with a high degree of accuracy. Their slightly less than perfect scores may point to a robustness that is advantageous in practical applications, ensuring that the models maintain high performance on unseen data. In sum, the exemplary AUC scores across our classifiers are a testament to the robustness of the feature selection methodology and the classifiers' abilities. This performance not only sets a benchmark for future predictive modeling endeavors but also demonstrates the potential of well-tuned models to achieve near-perfect classification in complex datasets.

When dealing with imbalanced datasets, it is typical to utilize resampling approaches to prevent the model from exhibiting bias towards the dominant class and a good technique we utilized is oversampling, which involves duplicating the minority class in order to achieve a balanced distribution of classes. With the huge imbalance of cases between classes in the UCI CKD dataset, employing this augmentation makes the size of the new combined dataset to surpassed the original 400 samples, thus making a sum of 475 samples. Therefore the model is evaluated with 95 samples as seen in confusion matrix plot of Fig. 9 and models like RF, SVM, LGB, Voting and XGB have no misclassification on the secondary diagonal. Generally, a greater value on the main diagonal of the confusion matrix indicates better performance of the model.

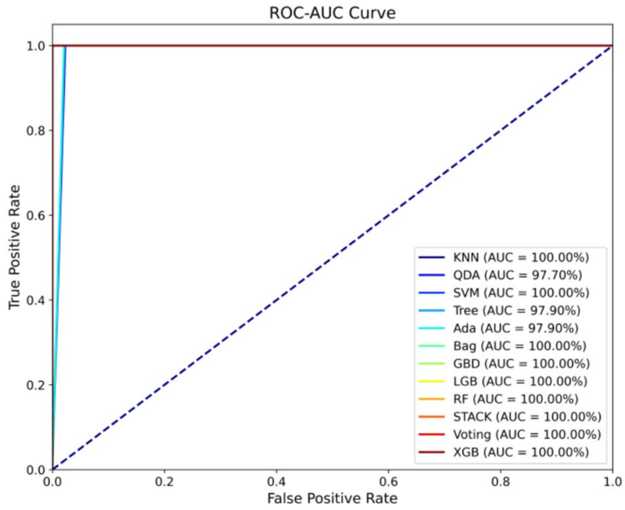


Fig. 8 ROC Curve of the proposed fusion-based FS using several classifiers

Figure 10 depicts the learning curves of both single and ensemble models by using the Proposed FFS method. Also, it can be seen that aside the quadratic discriminant analysis (qda), all the other classifiers converge gradually as the number of sample data increases. More so, it could be seen that all models aside from AdaBoost and SVM have better efficiency and gained higher cross validation score when the data sample is around 300. Generally, this depicts the robustness of the proposed model.

5.6 Computational cost

Generally, Figs. 11 and 12 illustrate the execution time taken for each single and ensemble classifier to make prediction and it could be seen that there remains consistency even with and without feature selection respectively. Additionally, the single classifier in Fig. 9 like KNN and QDA, a well as Ada classifier in Fig. 10 consistently have the lowest execution times within 0.008–0.284 s which is fast, while other classifiers like Bag and GBD show slightly higher times within 1.705–38.330 s.

However, the differences in execution times are not substantial and do not indicate a significant impact of feature selection on computation time. Therefore, it can be observed that feature selection does not significantly affect the computational cost of classifiers in this context, with minor variations in execution times for different models. Lastly, it can be seen that Fig. 11 as compared to Fig. 12 has QDA to be the least execution time of 0.008 s, thereby outperforming other algorithms in terms of speed.

5.7 SHAP interpretability discussion

This section illustrates the analysis of the black-box ML models used in the experiment with various SHAP plots such as summary plot, waterfall plot and dependency plot.

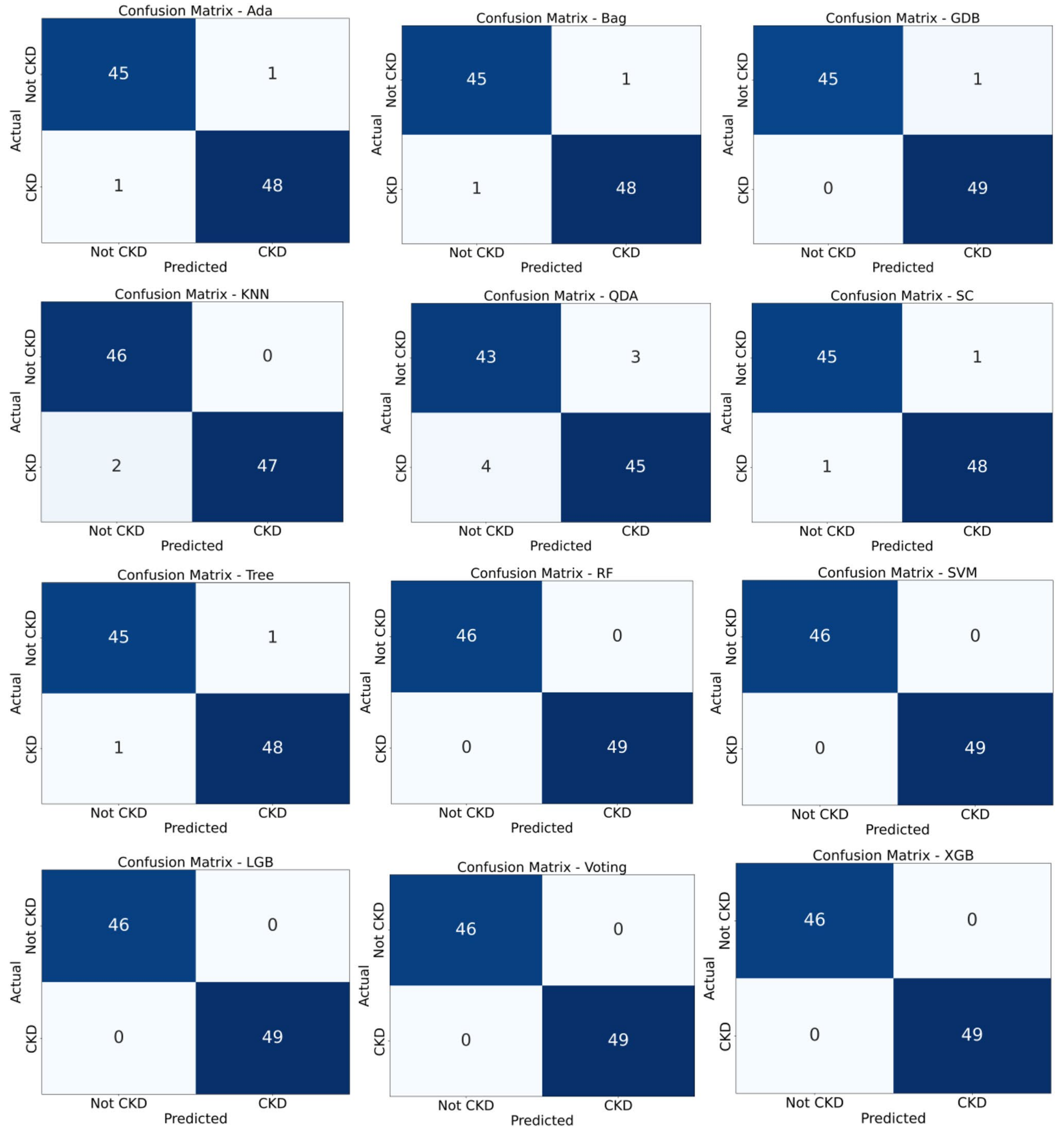


Fig. 9 Confusion matrix of Proposed FFS on both single and ensemble models

5.7.1 SHAP summary plot (bar) interpretability

The SHAP feature summary plot as seen in Fig. 13 is a method to provide a comprehensive explanation of each model and the features that have significant SHAP values are considered more important. The features are arranged in descending order of importance. The significance of a feature can be elucidated by the extent of its categorization capability and greater its SHAP value, the more significant the feature becomes. Identifying the key factors that significantly influence the prediction of CKD can enhance its diagnostic effectiveness and save unnecessary expenses associated with examinations. The significance of each feature is computed using both the single and ensemble models.

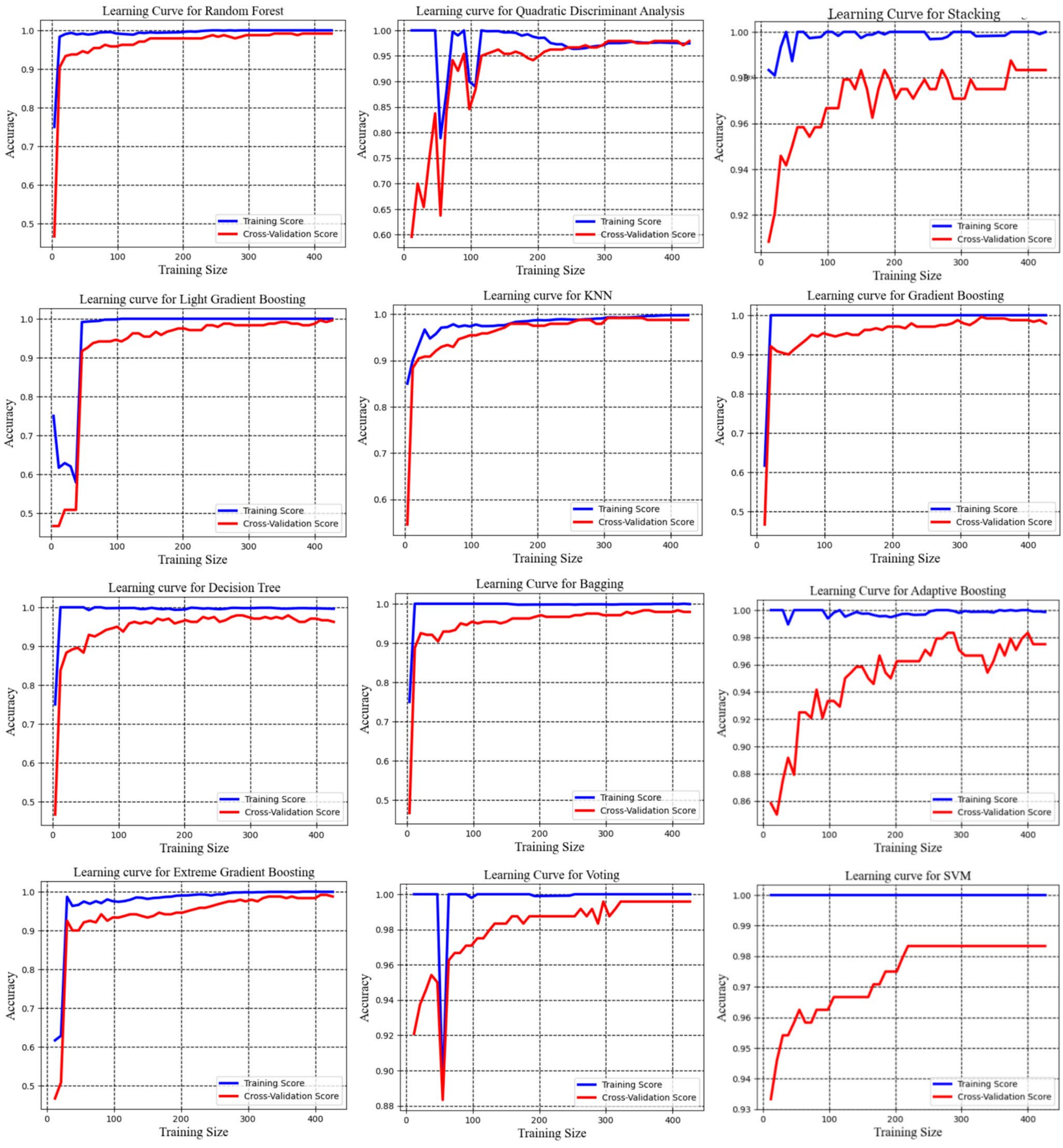


Fig. 10 Comparison of the learning curve of different models using Proposed FFS technique

The SHAP summary plots illustrate that the primary influences affecting the diagnosis of CKD are rbc from LDA and pcc from PCA found in most models. These major influences, LDA (rbc) and PCA (pcc) were identified as significant characteristics influencing CKD in SVM, KNN, Tree, QDA, XGB, GBDT, Ada, RF, Stacking, Voting, bagging and LGB models. Furthermore, rbc from LDA is identified as the most significant influence in terms of relevance in both the single and ensemble models.

5.7.2 SHAP individual summary plot (dot) interpretability

In order to investigate the extent to which the aforementioned features, as well as other features, affect the prediction

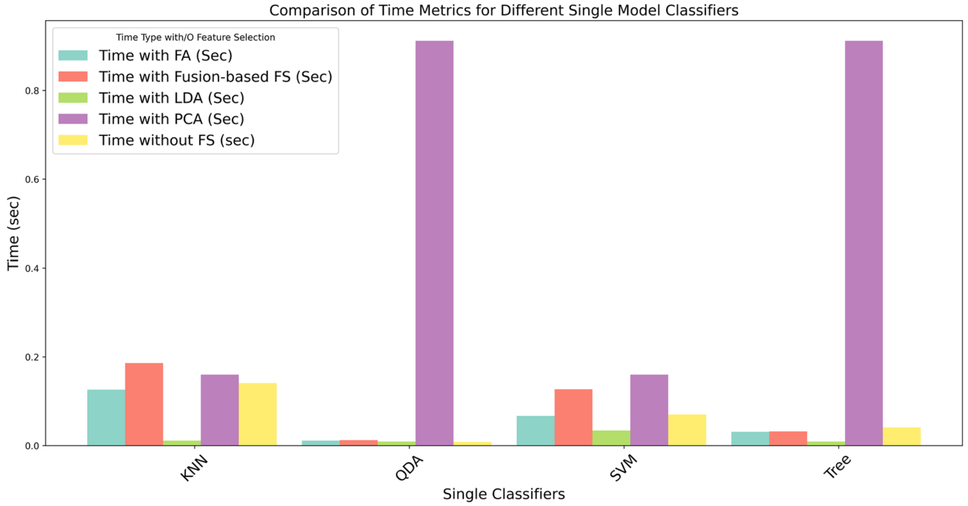


Fig. 11 Computational cost of each single classifier in second (s)

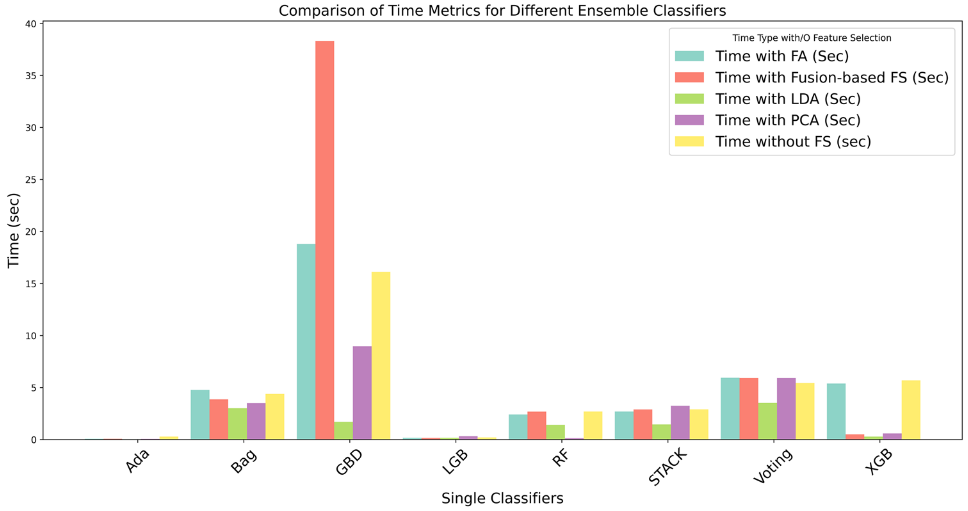


Fig. 12 Computational cost of each ensemble classifier in second (s)

of CKD, the SHAP values of the relevant features were individually computed as depicted in Fig. 14. Each data point in the dataset is depicted on the graph, with the characteristics displayed on the y-axis and the Shapley values displayed on the x-axis. The data is organized based on its relevance to disease prediction and ranked in a descending order. It is commonly employed for the purpose of interpreting feature importance and its impact. A data instance with a feature value of zero on the x-axis indicates that this characteristic has no contribution to the overall value of that instance. The magnitude and direction of the contribution increase when the SHAP value deviates from zero. The zero line symbolizes no contribution, but the magnitude of contributions increases as the SHAP value deviates from zero. In Fig. 14a, an increased occurrence of LDA (rbc) reduces the likelihood of developing CKD and vice versa, whereas, the presence of PCA (pcc) reduces the risk of CKD. Furthermore, it could be seen in stacking model plot of Fig. 14a that other features contribute less as the SHAP value is located within zero line. SVM plot in Fig. 14b reveals that LDA(rbc) and PCA(pcc) exhibit the most significant positive SHAP values, exerting a substantial influence on the SVM model's output by promoting higher predictions when these features have elevated values.

Additionally, LDA(rbc) and PCA(pcc) wide spread indicates that these features have a substantial influence on the predictions of the KNN model as presented in Fig. 14c, with considerable variation observed across different instances. In Fig. 14d, the RF model demonstrates a robust and favorable effect on the output for LDA(rbc) with closely grouped SHAP values, suggesting a persistent and significant influence on model predictions. The features in the QDA model, including PCA(ane) and FA(al), have SHAP values that are tightly grouped around zero, indicating a consistently small and uniform influence on the model's output as described in Fig. 14e. The features LDA(rbc) and PCA(pcc) have a significantly positive SHAP value impact, indicating that they are powerful predictors in the Voting ensemble model

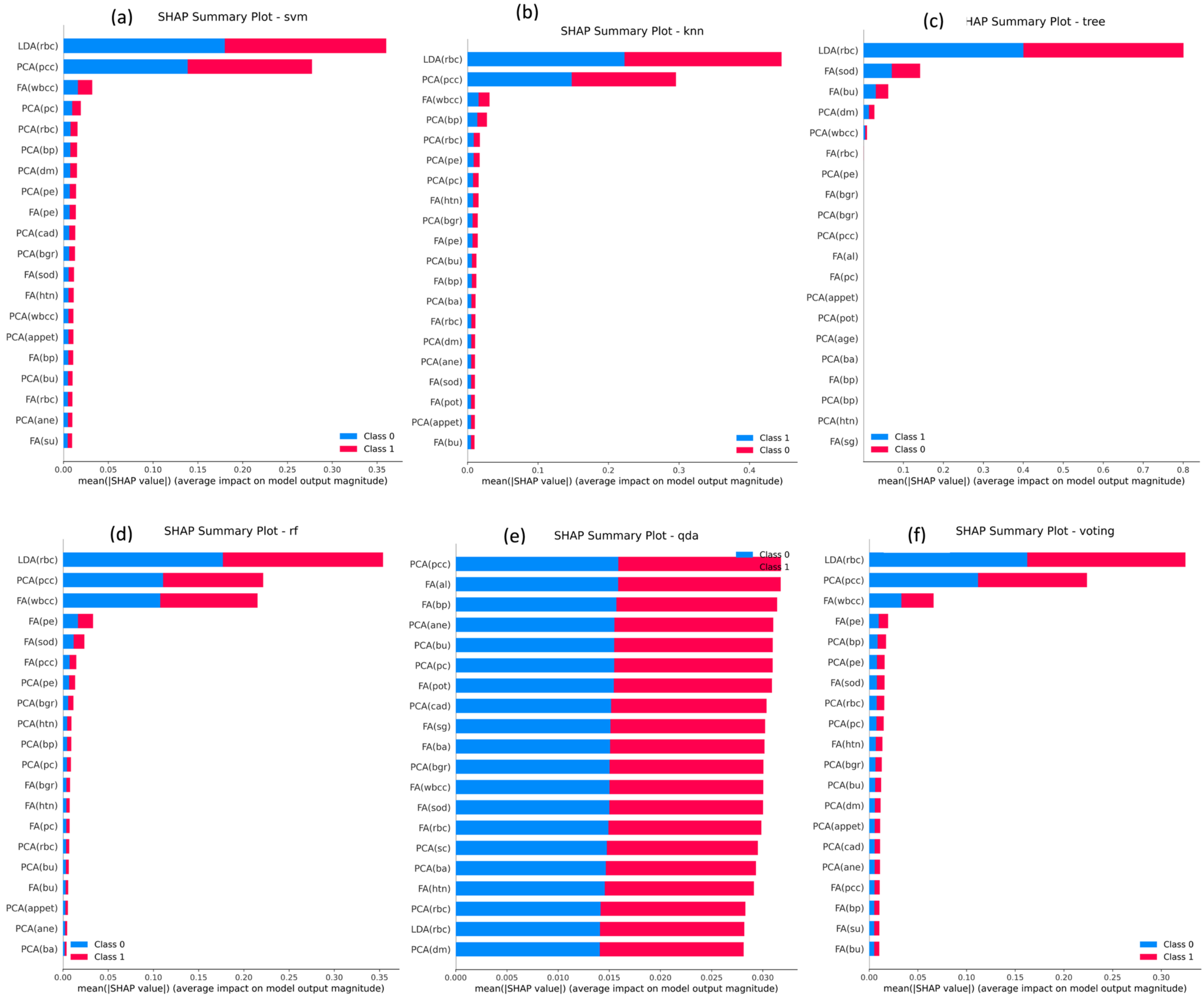


Fig. 13 Visual representation of the importance feature in SHAP summary value for: **a** SVM, **b** KNN, **c** Decision Tree, **d** RF, **e** QDA, **f** Voting, **g** Bag, **h** Ada, **i** GDBT, **j** XGB, **k** LGB, **l** Stacking (sc)

depicted in Fig. 14f. The Bagging model demonstrates that characteristics such as LDA(rbc) exhibit both positive and negative SHAP values, which are evenly distributed, hence influencing the model's output in both positive and negative directions as detailed in Fig. 14g.

More so, the SHAP values of AdaBoost for LDA(rbc) and PCA(pe) are much higher, suggesting that these features have a substantial impact on increasing the model's output as illustrated in depicted in Fig. 14h. In addition, Gradient Boosting model in depicted in Fig. 14i shows that the feature LDA(rbc) has high positive SHAP values, indicating that this feature has a large influence on predicting higher outcomes. LightGBM model highlights LDA(rbc) as a prominent feature with significantly positive SHAP values, suggesting that it plays a crucial role in producing greater prediction values as presented in Fig. 14j. Figure 14k which is the XGBoost model reveals that characteristics like LDA(rbc) have a broader distribution of SHAP values, indicating a varied influence on the model's output across various instances. In general, we observe a consistent pattern of feature impacts, even when there exist variations in the degree and range of these impacts. Lastly, Fig. 14l the stacking classifier, highlight that features like LDA(rbc), PCA(pcc) and FA(wbcc) are the most influential. The color gradient indicates the direction of this influence, with high values of these features typically driving the model towards a positive diagnosis, thereby providing insights into the model's decision making process.

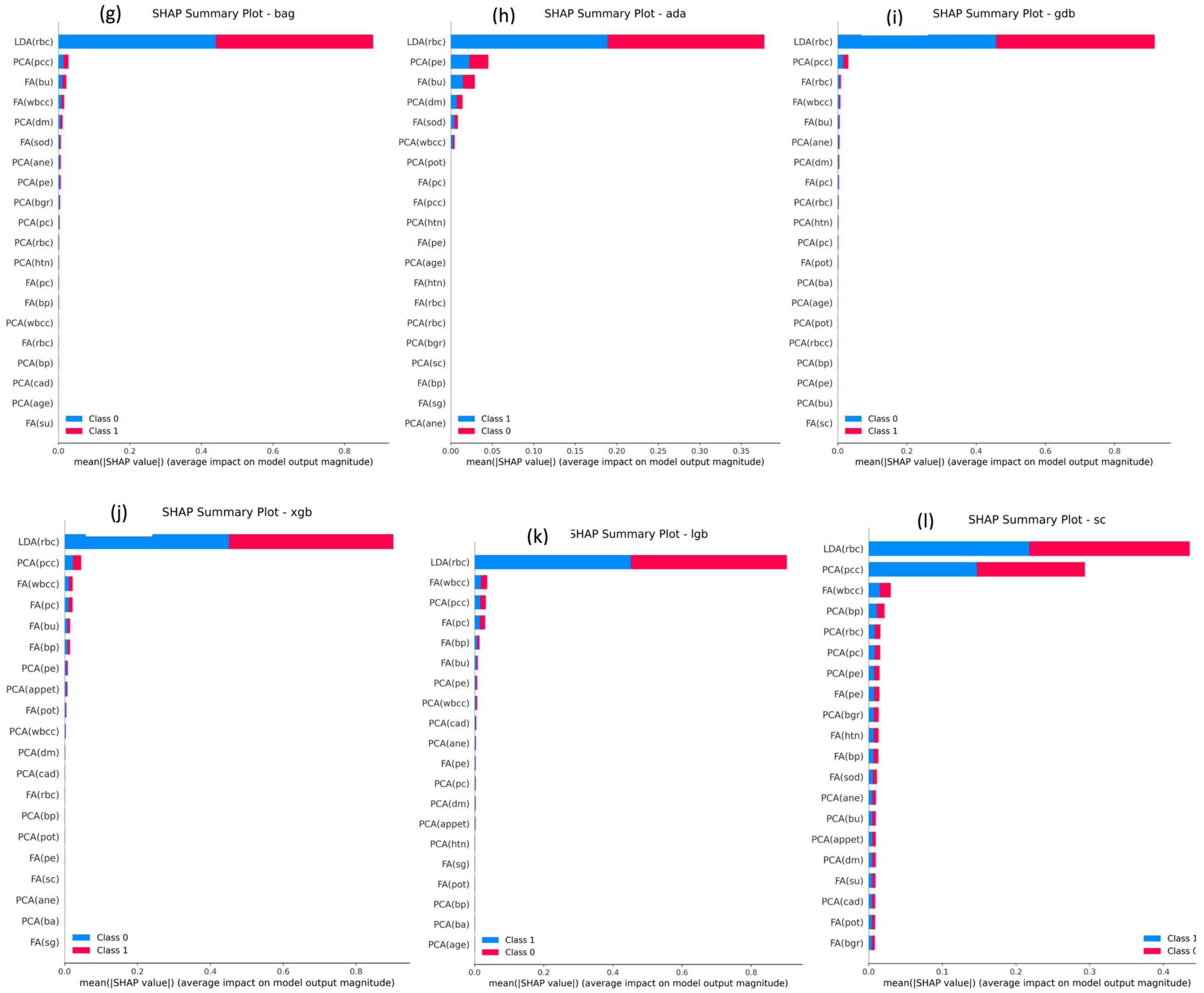


Fig. 13 (continued)

5.7.3 SHAP waterfall plot interpretability

Figure 15 displays the waterfall plot which represents the local explainability of the model performance on a per-instance basis. This plot provides a concise overview of the contribution of each attribute in predicting the classification of an individual occurrence. The features of the instance are represented on the y-axis. Each row is designated with either a red or blue hue. The color red signifies a positive contribution of the corresponding aspect, whereas the color blue signifies a negative contribution of the corresponding feature in the outcome of the given instance. The impact of each feature is determined by the value displayed in the horizontal box, which represents the deviation from the expected model output based on the background dataset to the model output for this specific prediction.

The SHAP waterfall plot in Fig. 15a indicates that the initial output begins with a value that implies a lower probability event. The model's prediction had the greatest significant augmentation due to LDA(rbc), which contributed a SHAP value of + 0.54. This was followed by smaller positive changes from PCA(pcc) and other features. The cumulative impact yields a conclusive forecast of 0.995, signifying a substantial probability of the anticipated result. The model's output is significantly impacted by a considerable positive contribution from LDA(rbc), resulting in a final prediction value of 0.995. The feature exhibits the most significant increase, but PCA(pcc) also has a positive contribution, although to a lower degree as displayed in Fig. 15b. In Fig. 15c, the initial value is negative, suggesting a less probable occurrence. The inclusion of the LDA(rbc) feature significantly enhances the model's output, with the PCA(pcc)

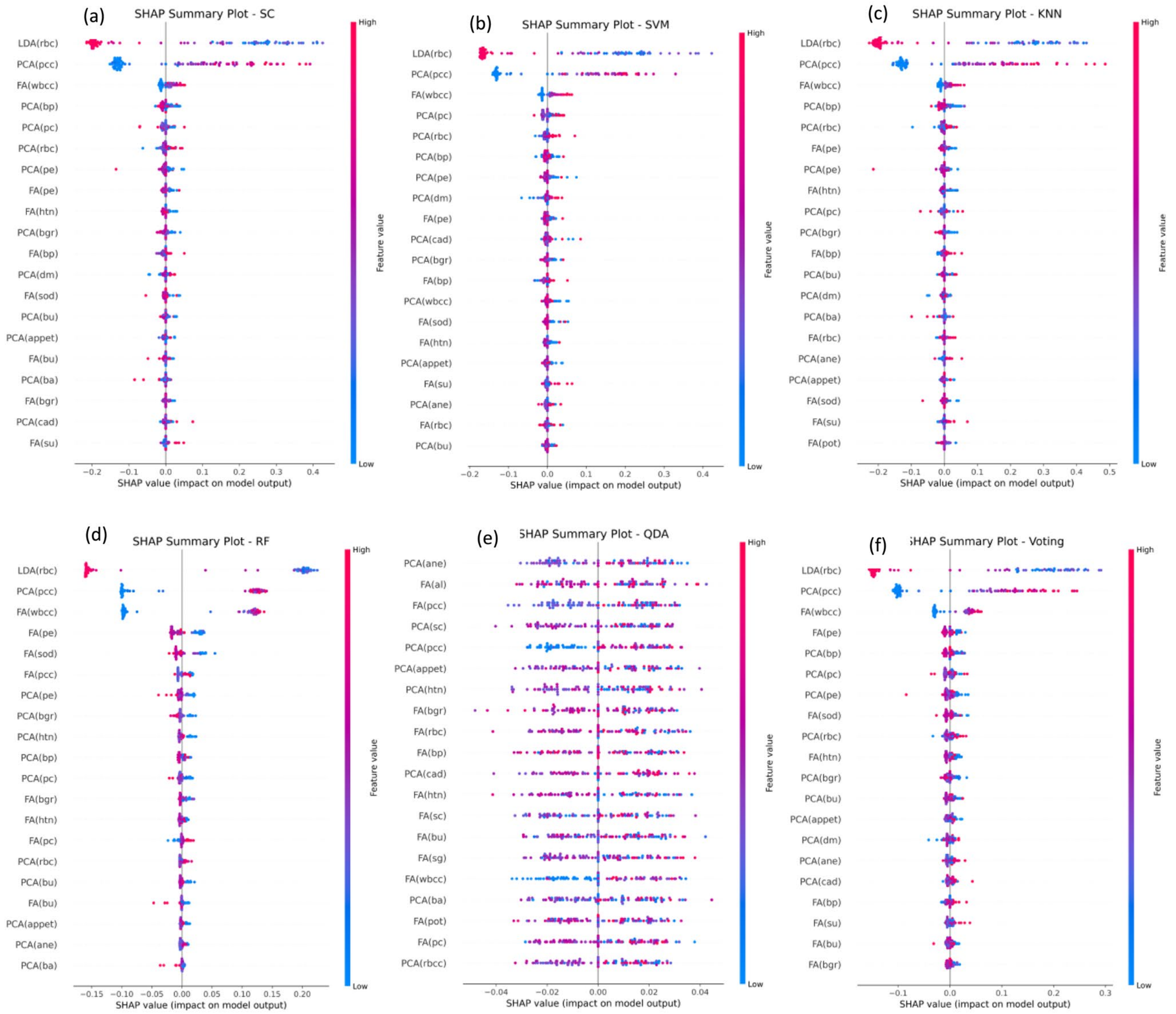


Fig. 14 Summary Plot of SHAP value for: **a** Stacking (SC), **b** SVM, **c** KNN, **d** RF, **e** QDA, **f** Voting, **g** Bag, **h** Ada, **i** GDBT, **j** XGB, **k** LGB, **l** Stacking (sc)

feature making the most major beneficial impact, resulting in a final prediction that is very near to one.

Similar to Fig. 15c, LDA(rbc) once again has a notable impact on enhancing the model's output. Principal Component Analysis (PCA) positively contributes, while Factor Analysis (FA) negatively impacts it, resulting in a final output of exactly one, indicating a highly probable event as presented in Fig. 15d. This plot in Fig. 15e likewise begins with a baseline value representing a less probable occurrence. Among the features, FA(bu) has the highest influence, contributing a considerable positive SHAP value of + 0.27 with LDA(rbc) also contributing positively. The model's ultimate forecast attains a value of 1, indicating a high level of assurance in the projected outcome. In Fig. 15f, the model output is primarily affected by positive contributions from FA(bu) and PCA(pcc), with a small negative contribution from PCA(ba) and the ultimate result is significantly elevated, measuring 0.91. Also, Fig. 15g demonstrates a significant favorable impact from FA(bu), with LDA(rbc) also exerting a positive influence on the model's output. The final predicted value is decreased by FA(pot), but, it still stays quite high at 0.818.

Furthermore, in Fig. 15h, PCA(pe) and FA(wbcc) have the most notable beneficial impacts, while PCA(appet) has a modest negative effect on the output. The ultimate forecast is singular, emphasizing the significant impact of the positive SHAP values. Also, it could be seen that FA(bu) has a significant positive impact, accompanied by relatively

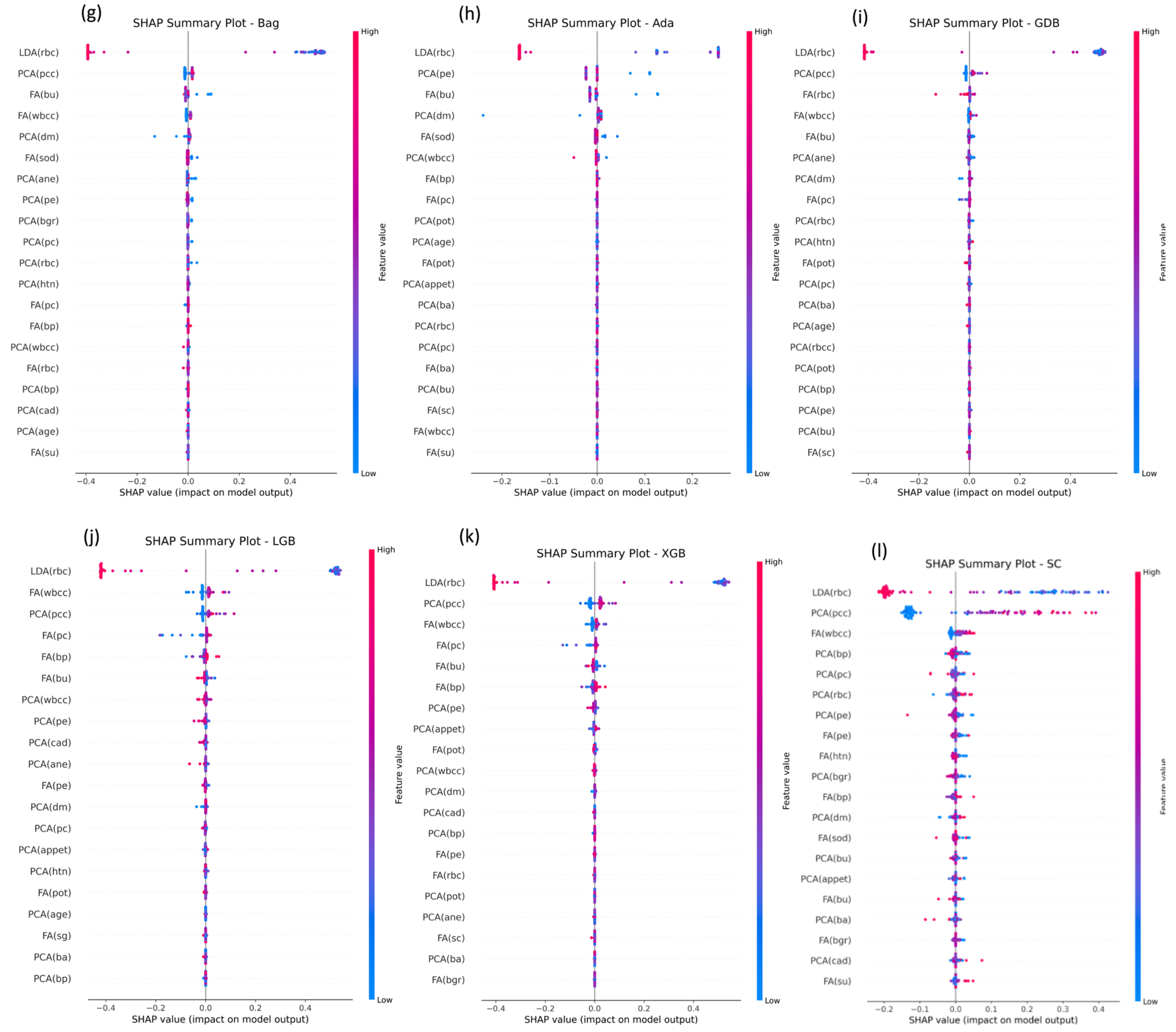


Fig. 14 (continued)

lower contributions from PCA(pcc) and FA(wbcc). The ultimate model forecast is exceptionally high, reaching 0.998 in Fig. 15i. FA(bu) once again exhibits the highest positive increment, with PCA(pcc) following closely behind. The ultimate result is a single value, which represents the combined total of these favorable inputs in Fig. 15j. The initial prediction is mostly enhanced by FA(bu), with LDA(rbc) also making a beneficial contribution. The ultimate model output is a single value, demonstrating the combined impact of favorable contributions from these characteristics as illustrated in Fig. 15k. Lastly, Fig. 15l shows that the prediction becomes almost certain, with a final output of 0.999. The feature LDA(rbc) continues to exert the greatest positive influence, while the contributions from other features such as FA(bu) and PCA(pcc) are very tiny but nevertheless positive. Together, these features collectively improve the accuracy of the prediction. Within every plot, LDA(rbc) and FA(bu) features significantly influence the model's output, directing predictions towards higher probabilities of anticipated occurrences, despite other features' influence.

5.7.4 SHAP dependency plot interpretability

Next, we examine the relationship between individual features of CKD and determine if feature values remain constant or if they change in relation to other features. Figure 16 provides a SHAP dependency plot which has detailed analyses of the significance and influence of features for CKD

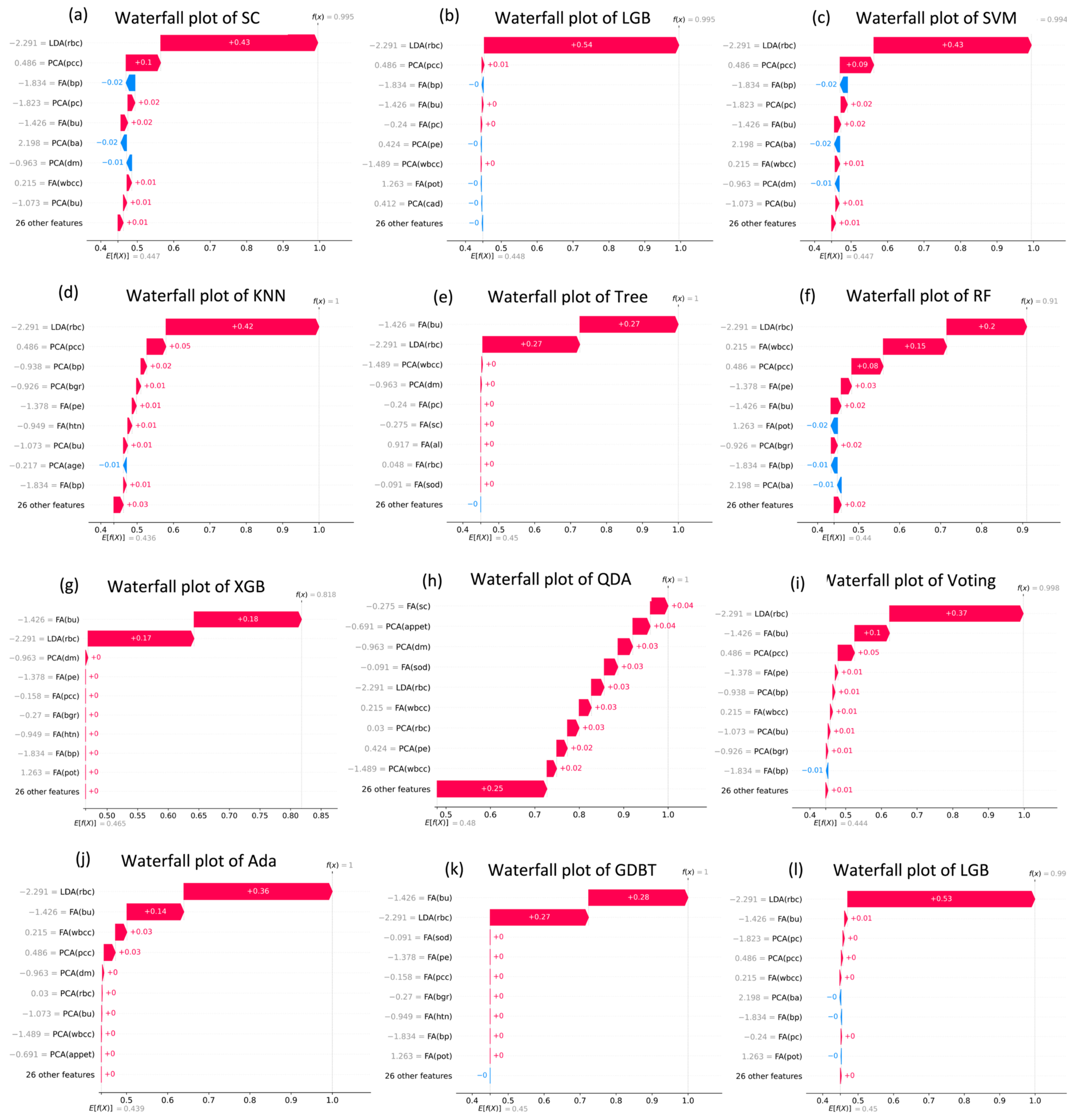


Fig. 15 Waterfall Plot of CKD for: **a** SC, **b** LGB, **c** SVM, **d** KNN, **e** Tree, **f** RF, **g** XGB, **h** QDA, **i** Voting, **j** Ada, **k** GDBT, **l** LGB

diagnosis. The analysis compares one feature to another and determines whether there is an interaction effect between the two. The plot's x-axis displays the feature, while the y-axis depicts the anticipated SHAP value for that feature.

The SHAP dependence plot in Fig. 16a for FA(pe) against the SHAP value for LDA(rbc) is used to determine the impact of a feature on CKD prediction. If there is a gradient or pattern in color that corresponds with the change in SHAP values, it suggests that LDA(rbc) interacts with FA(pe) to influence the model prediction. Clusters of similar colors with higher or lower SHAP values indicate that LDA(rbc) significantly contributes to the predictive effect of FA(pe). A vertical spread of points at a single value of FA(pe) with varying colors indicates that the SHAP value of FA(pe) is affected by different values of LDA(rbc). If the SHAP value increases or decreases as FA(pe) increases, it suggests a

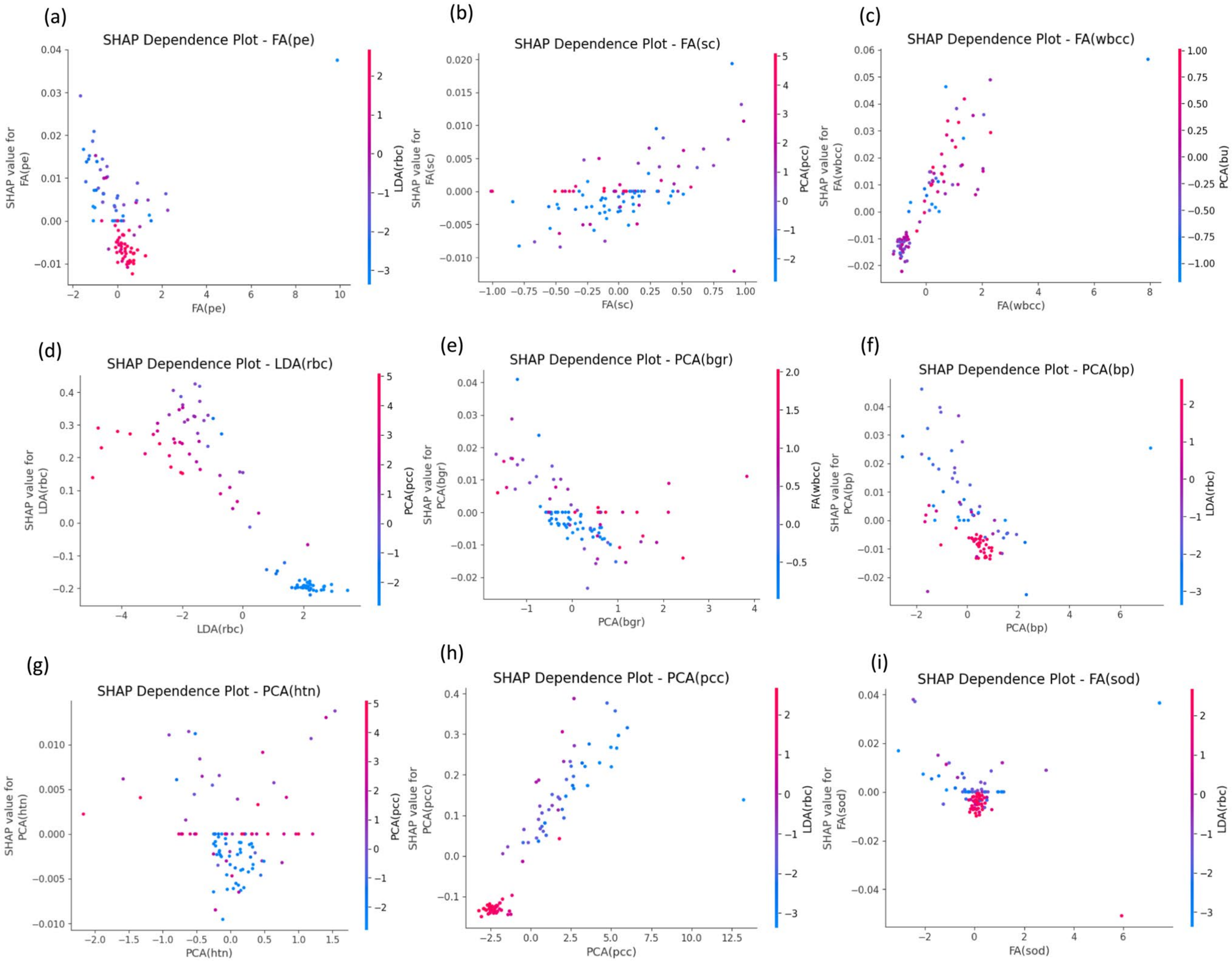


Fig. 16 SHAP dependency plot with some features

positive or negative correlation with the risk of CKD. Figure 16b shows that the SHAP values for FA(sc) vary in impact on the model's output, with most clustered near zero. The color difference between pink and blue points suggests an interaction effect, with higher SHAP values enhancing the impact of FA(sc) on the model's prediction which is important for identifying nuances in CKD diagnosis. Figure 16c indicates consistent higher or lower SHAP values, while outliers indicate strong or weak interactions. Also, the color gradient suggests an interaction effect between FA(wbcc) and PCA(bu), with higher or lower values of PCA(bu) enhancing or diminishing FA(wbcc)'s effect on model predictions. Figure 16d plot shows a cluster of points with high SHAP values in the middle range of LDA(rbc) values, suggesting a positive impact on the model's prediction of CKD. However, the cluster of blue points on the far right with low SHAP values suggests a decrease in the likelihood of CKD. The color distribution within the clusters suggests an interaction between LDA(rbc) and PC(Apcc).

Also, Fig. 16e shows that PCA(bgr) typically has a modest impact on the model's predictions, with SHAP values clustered between − 0.02 and 0.03. The color distribution suggests an interaction effect between PCA(bgr) and FA(wbcc), with points with higher FA(wbcc) values having positive SHAP values and vice versa. Outliers with higher SHAP values indicate a stronger influence on the model's prediction. Figure 16f depicts that PCA(bp) has a variable impact on the model's predictions, with a concentration of points around zero. A cluster of pink points indicates a slight decrease in the likelihood of the outcome being predicted as PCA(bp) approaches zero. Blue points correspond to lower LDA(hco) values, with some concentration at higher values, thus indicating that the interaction effect is more noticeable with PCA(bp) higher values. Figure 16g shows a cluster of blue points near zero, suggesting a low impact of PCA(htn) on the model's prediction. Outlying pink points indicate significant positive or negative impacts. A cluster of blue points around the center suggests a less influential range of PCA(htn) values combined with lower PCA(pcc) values indicate a significant interaction region. Figure 16h reveals a significant variation in SHAP values, particularly for PCA(pcc) values around 0, suggesting a significant impact on CKD prediction. Higher values are associated with CKD severity. The color distribution explores the interaction between PCA(pcc) and LDA(rbc). Figure 16i illustrates that FA(sod) has varying effects on the model's output, with larger absolute SHAP values having a more significant impact. A single outlier with a high positive SHAP value and high FA(sod) value indicates a significant impact. The color distribution shows a cluster of points with mid-range LDA(hbc) values, suggesting low impact. However, extreme values indicate an interaction effect, influencing the model's predictions. This interaction could help understand the interaction of biochemical markers in CKD pathology.

5.8 Comparative study of related works

Table 9 illustrates the comparison between our proposed method and other recent efforts, based on the average accuracy. This table demonstrates that each of the studies utilized distinct classification methods to identify Chronic Kidney Disease (CKD) from the UCI ML repository. Conversely, we utilized single and ensemble models, summing up to twelve ML classifiers. In addition, the borderline SMOTE and MICE algorithms [60, 61] have been utilized to address the issue of imbalanced data and to fill in the missing values respectively. Akter et al. [2] obtained an average accuracy of 91.14% by employing numerous machine learning methods with multiple imputations for missing values, whereas researchers in [35, 62] achieved similar performance of 98.75%, thus, their performances are less when compared to our accuracy result. Likewise, other studies [3, 20, 22, 32, 63] that utilized the ML classifiers as seen in Table 9 obtained an accuracy of 97.77%, 94.12%, 95%, 97.71% and 97.5% respectively, which is far smaller with our method

Table 9 Comparative Analysis with Recent Works

References	Handling data imbalance	Missing value imputation	Models	Split-ratio method	SHAP values	Avg Acc (%)
Akter et al. [2]	–	Multiple imputations	ANN, RF, LSTM, RNN, GRU, MLP, AdaBoost	tenfold CV Training:90% Test: 10%	–	91.14
Dritsas et al. [3]	–	SMOTE	NB, SVM, ANN, KNN, RF, Tree, SGD, Stacking, Voting. AdaBoost	tenfold CV Training:90% Test: 10%	–	97.77
Mondal et al. [4]	–	Multiple imputations	Optimized CNN, ANN and LSTM	Training:80% Test: 20%	–	96.5
Chittora et al. [20]	SMOTE	–	ANN, LR, LSVM, KNN, RT	Training:50% Test: 50%	–	94.12
Elhoseny et al. [22]	–	–	ACO Algorithm	tenfold CV Training:90% Test: 10%	–	95.00
Khan et al. [32]	–	Mean	NBTree, NB, LR, SVM, CHIRP, MLP	tenfold CV Training:90% Test: 10%	–	97.71
Almansour et al. [35]	–	Mean	SVM, ANN	tenfold CV Training:90% Test: 10%	–	98.75
Alsuhibany et al. [37]	ADASYN	Data mining	DBN, CNN,	-	–	96.91
Raihan et al. [62]	–	–	XGBoost	tenfold CV Training:70% Test: 30%	Summary plot	98.75
Moreno-Sanchez et al. [63]	–	Multiple imputations -	DT, RF, Tree, Ada, XGB	Training:70% Test: 30% fivefold CV	PDP plot and waterfall plot	97.5
Proposed FFS-IML	Borderline SVMSMOTE	MICE	Single and Ensemble classifiers	Shuffle-split CV Training:80% Test: 20%	Waterfall, summary, and dependency plots	99.47

with a decrease rate of 1.53%, 5.18%, 4.3% and 1.8% sequentially. Furthermore, other researchers [4, 37] applied deep neural networks on the same dataset obtained an accuracy of 96.5% and 96.91% which is 2.8% and 2.39 respectively less than our proposed method. Thus, our proposed model has been demonstrated to effectively enhance the accuracy of predicting CKD.

6 Conclusion and future work

This research proposes a novel fusion-based feature selection (FFS) method for classifying chronic kidney disease using machine learning classifiers and SHAP values. The proposed model employs the integration of dimensionality reduction of PCA, FA, and LDA which generates feature space that considers the highest amount of variation in the data, the covariance between the observed variables, and a linear combination of observed variables that optimizes the separation between classes. Furthermore, the missing values and imbalanced data were addressed using robust statistical techniques using MICE imputation and borderline SVMSMOTE algorithms respectively, in order to prevent overfitting and bias. The dataset was obtained from the UCI Machine Learning Repository which comprises 400 patients with 24 distinct attributes. Data split of 70:30 for train-test split ratio and also on tenfold cross validation were both used to validation the efficacy of the proposed FFS model. More so, both single and ensemble classifiers were optimized and employed to predict the diagnosis of CKD. Quadratic Discriminant Analysis (QDA) outperformed the other eleven algorithms in terms of speed with 0.033 s while the longest execution time is the gradient boosting decision tree (GBDT) with 38.33 s. The advantage of the study is that the proposed method demonstrated superior performance based on empirical evidence, achieving a perfect score of 100% in all of the evaluation metrics for SVM, LGB, RF, Voting and XGB classifiers compared to other existing literature that also utilized the same dataset. Furthermore, another advantage of this study is the integration of SHAP values which provided a clear, interpretable understanding of model predictions, enhancing transparency and aiding clinician experts in decision making. Therefore, clinicians would not only be able to diagnose the condition early using a smaller range of signs, but they could also concentrate on addressing specific characteristics to prevent the formation of chronic kidney disease or perhaps reverse its progression.

Despite its strength, the proposed model has limitations, such as its dependency on linear dimensionality reduction strategies, which may not capture complex, nonlinear relationships in the data. The study is also limited by the use of a single dataset, which may constrain the generalizability of the findings. Future work will explore non-linear dimensionality reduction methods and also focus on evaluating the prediction model in a clinical environment to assess its accuracy and reliability when applied to new patients' data. Also, we plan to employ deep learning methodology to accurately determine the stage of this disease and also utilizing other datasets to assess the dependability of our proposed approaches.

Author contributions Grace Ugochi Nneji and Happy Nkanta Monday: Data curation, Investigation, Formal analysis, Conceptualization, Supervision, Validation, Visualization, Software, Writing-original draft; Venkat Subramanyam Reddy Pathapati, Saifan Nahar, Goodness Temofe Mgbejime, Edwin Sunday Umana and Md Altab Hossin: Data curation, Resources, Formal analysis, Methodology, Review and editing process.

Funding This work was supported by the National Natural Science Foundation of China, the National High Technology Research and Development Program of China, the project of Science and Technology Department of Sichuan Province(Grant No. 2021YFG0322), the project of Science and Technology Department of Chongqing Municipality, the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJZD-K202114401), Chongqing Qinchengxing Technology Co., Ltd., Chengdu Haitian Digital Technology Co., Ltd., Chengdu Chengdian Network Technology Co., Ltd., Chengdu Civil-military Integration Project Management Co., Ltd., and Sichuan Yin Ten Gu Technology Co., Ltd.

Data availability The dataset is obtained from the UCI Machine Learning Repository, curated by D. Dua and C. Graff in 2019. It is located in Irvine, CA, and is affiliated with the University of California, School of Information and Computer Science and has been provided within the manuscript. Dataset is available at: http://archive.ics.uci.edu/dataset/336/chronic+kidney+disease.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Ruiz-Arenas R et al (2022) A summary of worldwide national activities in chronic kidney disease (CKD) testing. EJIFCC 28 (4):302. Accessed: Jan. 13, 2022. Available: /pmc/articles/PMC5746839/.
2. Akter S et al (2021) Comprehensive performance assessment of deep learning models in early prediction and risk identification of chronic kidney disease. IEEE Access 9:165184–165206. https://doi.org/10.1109/ACCESS.2021.3129491
3. Dritsas E, Trigka M (2022) Machine learning techniques for chronic kidney disease risk prediction. Big Data Cogn Comput 6(3):98. https://doi.org/10.3390/BDCC6030098
4. Mondol C et al (2022) Early prediction of chronic kidney disease: a comprehensive performance analysis of deep learning models. Algorithms 15(9):308. https://doi.org/10.3390/A15090308/S1
5. Nneji GU, Monday HN, Mgbejime GT, Pathapati VSR, Nahar S, Ukwuoma CC (2023) Lightweight separable convolution network for breast cancer histopathological identification. Diagnostics 13(2):299. https://doi.org/10.3390/diagnostics13020299

6. Monday HN, Li J, Nneji GU, Hossin MA, Nahar S, Jackson J, Ejiyi CJ (2022) COVID-19 diagnosis from chest x-ray images using a robust multi-resolution analysis siamese neural network with super-resolution convolutional neural network. Diagnostics 12(3):741. https://doi.org/10.3390/DIAGNOSTICS12030741
7. Stone JV (2018) Principal component analysis and factor analysis. Indep Compon Anal. https://doi.org/10.7551/mitpress/3717.003.0017
8. Monday HN, Li JP, Nneji GU, Hossin MA, Nahar S, Jackson J (2022) COVID-19 pneumonia classification based on neurowavelet capsule network. Healthcare 10(3):422. https://doi.org/10.3390/healthcare10030422
9. Sreejith S, Khanna Nehemiah H, Kannan A (2020) Clinical data classification using an enhanced SMOTE and chaotic evolutionary feature selection. Comput Biol Med 126(February):103991. https://doi.org/10.1016/j.compbiomed.2020.103991
10. Nneji GU, Cai J, Deng J, Hossin MA, Nahar S, Jackson J (2022) Identification of diabetic retinopathy using weighted fusion deep learning based on dual-channel fundus scans. Diagnostics 12(2):540. https://doi.org/10.3390/diagnostics12020540
11. Singh V, Jain D (2023) A hybrid parallel classification model for the diagnosis of chronic kidney disease. IJIMAI 8(2):14–28
12. Singh V, Asari VK, Rajasekaran R (2022) A deep neural network for early detection and prediction of chronic kidney disease. Diagnostics 12(1):116
13. Jain D, Singh V (2020) A novel hybrid approach for chronic disease classification. Int J Healthcare Inf Syst Inform (IJHISI) 15(1):1–19
14. Jain D, Singh V (2021) A two-phase hybrid approach using feature selection and adaptive SVM for chronic disease classification. Int J Comput Appl 43(6):524–536
15. Bhavekar GS, Das Goswami A, Vasantrao CP, Gaikwad AK, Zade AV, Vyawahare H (2024) Heart disease prediction using machine learning, deep Learning and optimization techniques—a semantic review. Multimed Tools Appl 83(39):86895–86922
16. Bhavekar GS, Goswami AD (2022) Herding exploring algorithm With light gradient boosting machine classifier for effective prediction of heart diseases. Int J Swarm Intell Res (IJSIR) 13(1):1–22
17. Miao J, Niu L (2016) A survey on feature selection. Procedia Comput Sci 91:919–926
18. Senan EM et al (2021) Diagnosis of chronic kidney disease using effective classification algorithms and recursive feature elimination techniques. J Healthc Eng. https://doi.org/10.1155/2021/1004767
19. Ogunleye A, Wang QG (2020) XGBoost model for chronic kidney disease diagnosis. IEEE/ACM Trans Comput Biol Bioinform 17(6):2131–2140. https://doi.org/10.1109/TCBB.2019.2911071
20. Chittora P et al (2021) Prediction of chronic kidney disease—a machine learning perspective. IEEE Access 9:17312–17334. https://doi.org/10.1109/ACCESS.2021.3053763
21. Drall S, Drall GS, Singh S, Naib BB (2018) Chronic kidney disease prediction using machine learning: A new approach. Int J Manag 8:278
22. Elhoseny M, Shankar K, Uthayakumar J (2019) Intelligent diagnostic prediction and classification system for chronic kidney disease. Sci Rep 9(1):1–14. https://doi.org/10.1038/s41598-019-46074-2
23. Agrawal P, Abutarboush HF, Ganesh T, Mohamed AW (2021) Metaheuristic algorithms on feature selection: a survey of one decade of research (2009–2019). IEEE Access 9:26766–26791
24. Hossain MM, Swarna RA, Mostafiz R, Shaha P, Pinky LY, Rahman MM, Rahman W, Hossain MS, Hossain ME, Iqbal MS (2022) Analysis of the performance of feature optimization techniques for the diagnosis of machine learning-based chronic kidney disease. Mach Learn Appl 9:100330
25. Wong W & Ming CI (2019) A review on metaheuristic algorithms: recent trends, benchmarking and applications. In: 2019 7th International Conference on Smart Computing & Communications (ICSCC) 1–5 https://doi.org/10.1109/ICSCC.2019.8843624
26. Sakri SB, Abdul Rashid NB, Muhammad Zain Z (2018) Particle swarm optimization feature selection for breast cancer recurrence prediction. IEEE Access 6:29637–29647
27. Khehra BS, Pharwaha APS (2017) Comparison of genetic algorithm, particle swarm optimization and biogeography-based optimization for feature selection to classify clusters of microcalcifcations. J Inst Eng India Ser B 98:189–202
28. Manonmani M, Balakrishnan S (2020) Feature selection using improved teaching learning based algorithm on chronic kidney disease dataset. Procedia Comput Sci 171:1660–1669
29. Qin J, Chen L, Liu Y, Liu C, Feng C, Chen B (2020) A machine learning methodology for diagnosing chronic kidney disease. IEEE Access 8:20991–21002. https://doi.org/10.1109/ACCESS.2019.2963053
30. Singh V, Asari VK, Rajasekaran R (2022) A deep neural network for early detection and prediction of chronic kidney disease. Diagnostics 12(1):116. https://doi.org/10.3390/DIAGNOSTICS12010116
31. Vasquez-Morales GR, Martinez-Monterrubio SM, Moreno-Ger P, RecioGarcia JA (2019) Explainable prediction of chronic renal disease in the colombian population using neural networks and case-based reasoning. IEEE Access 7:152900–152910. https://doi.org/10.1109/ACCESS.2019.2948430
32. Khan B, Naseem R, Muhammad F, Abbas G, Kim S (2020) An empirical evaluation of machine learning techniques for chronic kidney disease prediction. IEEE Access 8:55012–55022. https://doi.org/10.1109/ACCESS.2020.2981689
33. Alasker H, Alharkan S, Alharkan W, Zaki A, Riza LS (2017) Detection of kidney disease using various intelligent classifiers. In: Proceeding—2017 3rd International Conference on Science and Information Technology of Theory Application IT Educ. Ind. Soc. Big Data Era, ICSITech 2017, vol. 2018- January, pp 681–684. https://doi.org/10.1109/ICSITech.2017.8257199
34. AnanthaPadmanaban KR, Parthiban G (2016) Applying machine learning techniques for predicting the risk of chronic kidney disease. Indian J Sci Technol 9(29):1–5. https://doi.org/10.17485/ijst/2016/v9i29/93880
35. Almansour NA et al (2019) Neural network and support vector machine for the prediction of chronic kidney disease: a comparative study. Comput Biol Med 109(October 2018):101–111. https://doi.org/10.1016/j.compbiomed.2019.04.017
36. Akben SB (2018) Early stage chronic kidney disease diagnosis by applying data mining methods to urinalysis. Blood Anal Dis History IRBM 39(5):353–358. https://doi.org/10.1016/J.IRBM.2018.09.004
37. Alsuhibany SA et al (2021) Ensemble of deep learning based clinical decision support system for chronic kidney disease diagnosis in medical internet of things environment. Comput Intell Neurosci. https://doi.org/10.1155/2021/4931450
38. Petch J, Di S, Nelson W (2022) Opening the black box: the promise and limitations of explainable machine learning in cardiology. Can J Cardiol 38:204–213
39. Adadi A, Berrada M (2018) Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). IEEE Access 6:52138–52160
40. Montavon G, Samek W, Müller K-R (2018) Methods for interpreting and understanding deep neural networks. Digit Signal Process 73:1–15

41. Ahmad MA, Eckert C & Teredesai A (2018) Interpretable machine learning in healthcare. In: Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics 559–560. https://doi.org/10.1145/3233547.3233667
42. Linardatos P, Papastefanopoulos V, Kotsiantis S (2021) Explainable AI: a review of machine learning interpretability methods. Entropy 23:18
43. Tjoa E, Guan C (2021) A survey on explainable artifcial intelligence (XAI): toward medical XAI. IEEE Trans Neural Netw Learn Syst 32:4793–4813
44. Lundberg SM & Lee S-I (2017) A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems vol. 30
45. Rikta ST, Uddin KMM, Biswas N, Mostafiz R, Sharmin F, and Dey SK. XML-GBM lung: an explainable machine learning-based application for the diagnosis of lung
46. Liao B, Liang J, Guo B, Jia X, Jiarong Lu, Zhang T, Sun R (2023) ILSHIP: an interpretable and predictive model for hypothyroidism. Comput Biol Med 154:106578
47. Zhang K, Xu P & Zhang J (2020) Explainable AI in deep reinforcement learning models: a SHAP method applied in power system emergency control. In: 2020 IEEE 4th Conference on Energy Internet and Energy System Integration (EI2) 711–716. https://doi.org/10.1109/EI250167.2020.9347147
48. Dikshit A, Pradhan B (2021) Interpretable and explainable AI (XAI) model for spatial drought prediction. Sci Total Environ 801:149797
49. Parsa AB, Movahedi A, Taghipour H, Derrible S, Mohammadian A (2020) Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. Accid Anal Prev 136:105405
50. Rubini L, Soundarapandian P, and Eswaran P (2015) Chronic_Kidney_Disease. UCI Machine Learning Repository. https://doi.org/10.24432/C5G020
51. Emmanuel T, Maupong T, Mpoeleng D, Semong T, Mphago B, Tabona O (2021) A survey on missing data in machine learning, vol 8. Springer International Publishing (**no. 1**)
52. Nijman S et al (2022) Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review. J Clin Epidemiol 142:218–229. https://doi.org/10.1016/j.jclinepi.2021.11.023
53. Murti DMP, Pujianto U, Wibawa AP and Akbar MI (2019) K-Nearest Neighbor (KNN) based missing data imputation. In: Proceeding - 2019 5th International Conference of Science and Information Technology Embrac. Ind. 4.0 Towar. Innov. Cyber Phys. Syst. ICSITech 2019, pp 83–88. https://doi.org/10.1109/ICSITech46713.2019.8987530.
54. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357. https://doi.org/10.1613/JAIR.953
55. Han H, Wang WY, Mao BH (2005) Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning, Lecture Notes of Computer Science. vol. 3644, no. PART I 878–887. https://doi.org/10.1007/11538059_91
56. Houssein EH, Sayed A (2023) A modified weighted mean of vectors optimizer for Chronic Kidney disease classification. Comput Biol Med 155:106691
57. Shanthakumari AS, Jayakarthik R (2023) Utilizing support vector machines for predictive analytics in chronic kidney diseases. Mater Today Proc 81:951–956
58. Rahman S, Irfan M, Raza M, Ghori KM, Yaqoob S, Awais M (2020) Performance analysis of boosting classifiers in recognizing activities of daily living. Int J Environ Res Public Heal. 17(3):1082. https://doi.org/10.3390/IJERPH17031082
59. Dogan A (2019) A weighted majority voting ensemble approach for classification. In: 2019 4th International Conference of Computer Science Engineering, pp 1–6. https://doi.org/10.1109/UBMK.2019.8907028
60. Singh J, Bagga S, Kaur R (2019) Software-based prediction of liver disease with feature selection and classification techniques. Procedia Comput Sci 2020(167):1970–1980. https://doi.org/10.1016/j.procs.2020.03.226
61. Liu X, Zhang X, Chen B (2025) Feature selections based on fuzzy probability dominance rough sets in interval-valued ordered decision systems. Int J Mach Learn Cyber. https://doi.org/10.1007/s13042-025-02562-8
62. Raihan MJ, Al-Masrur Khan M, Kee SH, Nahid AA (2023) Detection of the chronic kidney disease using XGBoost classifier and explaining the influence of the attributes on the model using SHAP. Sci Reports 13(1):6263
63. Moreno-Sanchez PA (2021) Development and evaluation of an explainable prediction model for chronic kidney disease patients based on ensemble trees. arXiv preprint arXiv:2105.10368

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.