Enhanced Prediction of Chronic Kidney Disease using XGBoost Machine Learning Model

Rajeshree Khande[1], Nrashant Singh[2,*], Sachin Naik[3], Satvik Bodke[3], Aniket Gaikwad[3], P.S. Metkewar[3], Raeesa Bashir[4] and Aafaq A. Rather[5,*]

[1]*Balaji Institute of Techology & Management, Sri Balaji University, Pune, Maharashtra, India*

[2]*Department of Forensic Science, Amity University Dubai, Dubai International Academic City (DIAC), Dubai, UAE*

[3]*School of Computer Science and Engineering, Department of Computer Science and Applications, Dr. Vishwanath Karad MIT World Peace University, Pune, India*

[4]*Department of Mathematics and Statistics, Faculty of Science and Technology, Vishwakarma University, Pune, India*

[5]*Symbiosis Statistical Institute, Symbiosis International (Deemed University), Pune, India*

Abstract: Chronic kidney disease (CKD) might progress to end stage renal disease; moreover, cardiovascular dangers are dire. Machine learning used in for more speed and accurate diagnosis of CKD. The CKD prediction model proposed in this paper was developed using the XGBoost algorithm, which is quite effective in classification problems. Other clinical parameters such as blood urea, serum creatinine and white blood cell count are some of the 24 indices identified from among the 400 patient records in the dataset. Feature selection using SelectKBest was relevant, and hyperparameter tuning was done by RandomizedSearchCV Both quantitative and categorical data were preprocessed. Altogether, 75% of data used for training, while 25% of data used for testing. The XGBoost model had a better result with 96.88 % recall, 100% precision, and 98% accuracy. However, the proposed approach has disadvantages; namely, a small sample cross-section and possibly an imbalanced class. Further, the dataset will be increased, the methods of dealing with class imbalance will be applied using SMOTE algorithm, and the effectiveness of the proposed model will be tested in real clinical practice. This work also highlight how crucial it is to employ and enhance machine learning, especially XGBoost to detect early stage of CKD, proper treatment, low mortality rate, and increased survival rate among patients.

Keywords: Chronic Kidney Disease, XGBoost, Machine Learning, Predictive Modeling, Feature Selection.

1. INTRODUCTION

It appears that over 20 million people globally suffer from chronic kidney disease (CKD), which constitutes a major health issue [1]. CKD is a progressive renal disease and is defined by the presence of a decrement in glomerular filtration rate [2]. Some-times, the disease is diagnosed when the patient can only undergo hemodialysis or kidney transplantation [3]. If the disease is recognized early enough it can reverse the progression of CKD and reduce it's associated morbidity and mortality [4]. Physical examination and clinical studies are typical approaches to a diagnosis, but they remain time consuming and error prone due to human intervention. In this case, machine learning has been extended and applied in the healthcare sector for purposes of diagnosing diseases such as chronic kidney disease [5]. Structured clinical big data may be fed into the machines so that the prediction algorithm yields relations that physicians might not otherwise identify [6]. They also enhance the time and efficiency of diagnosis and provide beneficial decision aids for the first stages of a disease. The aim of this study is to train the CKD prediction model using the XGBoost model since it is robust in classification tasks. Twenty-four features in total were provided in the 400 patient records from the dataset employed for this study; these features incorporated clinically oriented features such as hemoglobin, blood urea, serum creatinine, white blood cell count, among others. The approach used to address the missing value problem was mode imputation for categorical independent variables while the remaining numerical independent variables were imputed randomly. Furthermore, data of categorical variables was numerically encoded using LabelEncoder. For SelectKBest, the top 10 highest characteristics were selected to limit the model to focus on crucial aspects, such as hemoglobin, leukocytes, and serum creatinine. In the current study, the dataset was split 75:25 to ensure that the model developed was both strong during training and testing. By setting gamma, max-depth and learning rate in RandomizedSearchCV, they were optimized. After training, the model was successful in outcompeting conven-tional CKD prediction models whose accuracy ranges between 85-95%, with accuracy being 98%, precision 100% and recall rate of 96.88%.

2. LITERATURE REVIEW

Md. Taufiqul Haque Khan Tusar *et al*. (2022) proposed an ML approach for the classification of CKD

*Address correspondence to these authors at the Department of Forensic Science, Amity University Dubai, Dubai International Academic City (DIAC), Dubai, UAE; E-mail: nsingh@amityuniversity.ae
Symbiosis Statistical Institute, Symbiosis International (Deemed University), Pune, India; E-mail: aafaq7741@gmail.com

with 400 samples and 24 features. Several algorithms which were utilized by the authors include Random Forest, SVM, Gaussian NB, Decision Tree, Logistic Regression, KNN, Gradient Boosting, Adaptive Boosting and XGBoost. First, Random Forest was the most accurate model with accuracy of 100%. Missing value prediction was applied to impute them using KNN imputation, similarly for outlier removal a LOF was used Upon this step data was balanced using SMOTE. The authors suggest a hybrid feature selection technique and indicated that more studies should be dedicated to employing real-time data and integrating domain knowledge into the algorithm [7]. Durga P and Sudhakar T (2023) offered a proposal of the algorithm for the classification of CKD with 400 samples and 25 variables of the UCI library. The authors used such techniques as Artificial Neural Networks, XGBoost, and K-Nearest Neighbors. In the methods used they did imputation by mean for missing values, encoding for categorical values and correlation method for selecting features. With regards to accuracy, the best model was XGBoost with 98.33% while ANN yielded 91%. In future research, predictions using ensemble methods and real-time data will be utilized to enhance accuracy of the explanation prediction model. [8]. Afia Farjana *et al*. (2023) the target was to employ the ML classification to forecast CKD employing a dataset of 400 instances and 24 features. They employed several of them, including KNN, SVM, Logistic Regression, Naive Bayes, Extra Trees Classifier, AdaBoost, XGBoost, LightGBM. The process is Data cleaning, missing value analysis and feature extraction. LightGBM achieved the highest accuracy of 99%. One important implication of this study is that future studies should replicate the research with more participants as part of the dataset to explore the application of deeper learning for the enhancement of the models [9]. Nishin James and Jitendra Kaushik (2022) focused using a dataset of 400 instances and 24 attributes for CKD risk prediction. Random Forest, XGBoost, MLP, SVMs, and Decision Trees were employed during the study. It included feature selecting and data dividing into the training set and testing set. Random Forest as well as XGBoost achieved test accuracy of 97.5%. The authors suggest trying the same algorithm on a larger sample size, as well as testing other more sophisticated models, including deep learning [10]. Sanskruti Patel *et al*. (2022) sought to develop the machine learning model for the prediction of CKD using occurrences of 400 and characteristics of 25. They utilized specific algorithms such as XGBoost, Decision Tree, SVM, KNN. Some values were missing and to handle this mode imputation was done, also categorical data was converted into binary form. Finally, the model with the highest accuracy was XGBoost with 99% accuracy. Therefore, the authors recommended that subsequent studies should concentrate on the adoption of these models for early CKD detection employing automated systems and exploring superior computational learning techniques [11, 21]. Sridevi P., Rabbani M., and Ahamed S. I. (2023) studied the application of machine learning models for forecasting CHD and other chronic diseases such as kidney diseases, diabetes, high blood pressure, and anemia. In order to enhance performance of the models, randomly selected features with techniques such as Random Forest, Decision Tree, XGBoost, LightGBM, SVM, feature extraction and feature selection were performed. XGBoost ensemble and LightGBM were identified as more accurate with XGBoost rising to 99% in CKD prediction. The authors call for these models to be tested on big, heterogeneous populations in future research to determine their utility in the medical setting [12]. Ping Liang *et al*. (2023) in the present study, through 1,765 patients, we have implemented a deep learning approach for predicting CKD progression to ESRD. The models that were used were DNN, Random Forest, XGBoost and Lasso with DNN giving the highest AUC-ROC of 0.8991. The investigation involved feature selection and there attribution methods including Integrated gradients and DeepLIFT for analysis of the models. They urge greater research to put these models in larger samples and implement these in practice [13]. Adiba Haque *et al*. (2022) a hardware description of the analysis of CKD and heart failure (HF) relationship was made with machine learning algorithms like, Random Forest, XGBoost, CatBoost, Logistic Regression and SVM. This was done in order to log transform and standardize the datasets that the authors obtained from the UCI Machine Learning Repository. Prediction of serum creatinine, serum salt, and diabetes mellitus was good for Random Forest, XGBoost, and CatBoost. Subsequent works will focus on the implementation of these algorithms into live clinical data and the use of wearables to assess CKD and HF [14]. Iftekhar Ahmed *et al*. (2022) offered a machine learning algorithm for the prediction of CKD using 400 samples with 35 but 25 features only was used. Random Forest, SVM, Naive Bayes, Logistic Regression, KNN, XGBoost, Decision Tree and AdaBoost were among the features employed by the authors. However, as explained in the data preprocessing section, handling of missing values was done through imputation. Out of the selected algorithms the Random Forest and Logistic Regression had the highest accuracy, 99%. The subsequent studies will focus on the refinement of the models and the exploration of the new data sources for more effective usage of the models [15].

Based on review studies of the diagnosis of chronic kidney disease (CKD) with the help of different machine learning algorithms, a number of gaps and research avenues have been identified. Thus, it is mainly possible to improve model accuracy and

applicability in practice by addressing the problem of class imbalance in the datasets and improving the features selection methods. Existing research works demonstrates how it is possible to harness state of the art technique, like ensemble learning scheme and real time data analytics to enhance the predictiveness and robustness of models for the detection of chronic kidney disease. As such it is also recommended that future studies focus on employing these models in healthcare environments where the models' utility of enhanced and larger datasets may be explored. After doing extensive literature review the authors concluded that while advances for machine learning are promising in managing the prediction of CKD, the picture in overall research exhibits this process indicating that regular enhancement of algorithms and data analysis is critical for patients' improvement and diagnostic methods in healthcare settings. For those who remain undiagnosed and untreated, chronic kidney disease or CKD, a significant health problem, usually progresses to end-stage renal disease or ESRD. By the nature of traditional diagnostic techniques they are slow, prone to human error, and require manual intervention – the risk of adverse effects and early detection are increased. From the above analysis, we can be able to infer that Machine learning models can enhance the rate and precision of diagnosis. The challenges current CKD prediction models then found are having small dataset size, imbalance class, missing data, overfitting issues, non-interpretable and often inadequate validation in actual clinical practice. These difficulties limit the usability of the models in health care practice areas and their reliability. The initiative has two goals in mind: However, to develop a sound and accurate CKD prediction model, the XGBoost algorithm is proposed to be adopted. To enhance model performance though handling issues like overfitting and missing data the following means entails involved complex data preprocessing tactics, using SelectKBest to perform feature selection, and RandomizedSearchCV in hyperparameter tuning to address some limitations of small and imbalanced datasets and ensure the practical applicability of the introduced approach. This involves, for assessing the usefulness of the model and its applicability in proper health care facilities, reviewing data augmentation techniques such as SMOTE, increasing the data by other diverse samples, and performing validations within actual clinical contexts.

3. METHODOLOGY

3.1. Data Source and Description

The CKD dataset has 26 columns and 400 recorded patient data sets. Among these, twenty-four characteristics are applied for defining the presence or absence of CKD in patients. The dataset was procured from Kaggle by Mansoor Daku and contains both numerical and categorized features. Diabetic patients must check important parameters like age, blood pressure, urine specific gravity, serum albumin and several other details to diagnose CKD. Ten characteristics assessed have more than 50 percent of the data missing, including age, blood pressure and albumin. Thus, it is necessary to appropriately exploit these gaps in order to ensure that the model will give the correct prognosis. It also includes other health information that doctors use to assess the kidney's functions including blood urea, serum creatinine, Red and White blood cell counts and other test results. Besides, the dataset presents further comorbidities that are common in CKD patients, including anemia, diabetes, and coronary artery disease. Organizing the data into two categories is the dataset's goal patients with CKD are labeled as "ckd" while patients without are labeled as "notckd". These characteristics enable a machine learning algorithm to predict which populations are predisposed to CKD using medical information. The following Figure **1** also represents the entire workflow of the model developed for the identification of Chronic Kidney Disease (CKD). It begins with data preprocessing and feature important where the Chi-Squared method is used for this purpose. The datasets are randomly separated into set 75% for training and 25% for testing. Classifier model is again set as XGBoost and hyperparameter tuned using RandomizedSearchCV. After training is complete, there are some significant characteristics to investigate, such as accuracy and the ability of the AUC-ROC score to predict the likelihood of classification on the testing data, feature importance.

The Figure **2** Feature representation, illustrates the frequency distribution of data types across the dataset's features. The most common data type is *Float, used for numerical measurements with decimal values, such as blood glucose levels and hemoglobin. **Categorical* data is the second most frequent, representing features like "Hypertension Status" or "Appetite Condition" with discrete categories (e.g., "yes," "no," "good," "poor"). *Integer* is the least common, primarily used for whole-number features like "White Blood Cell Count" and unique identifiers. This distribution highlights the dataset's emphasis on numerical measurements alongside categorical labels for classification and descriptive purposes.

3.2. Data Preprocessing

Data Loading: As for the research the medical records collection focused on the chronic kidney disease (CKD) was used. Together with age and blood pressure the collection concerns 24 traits, containing 400 entries.

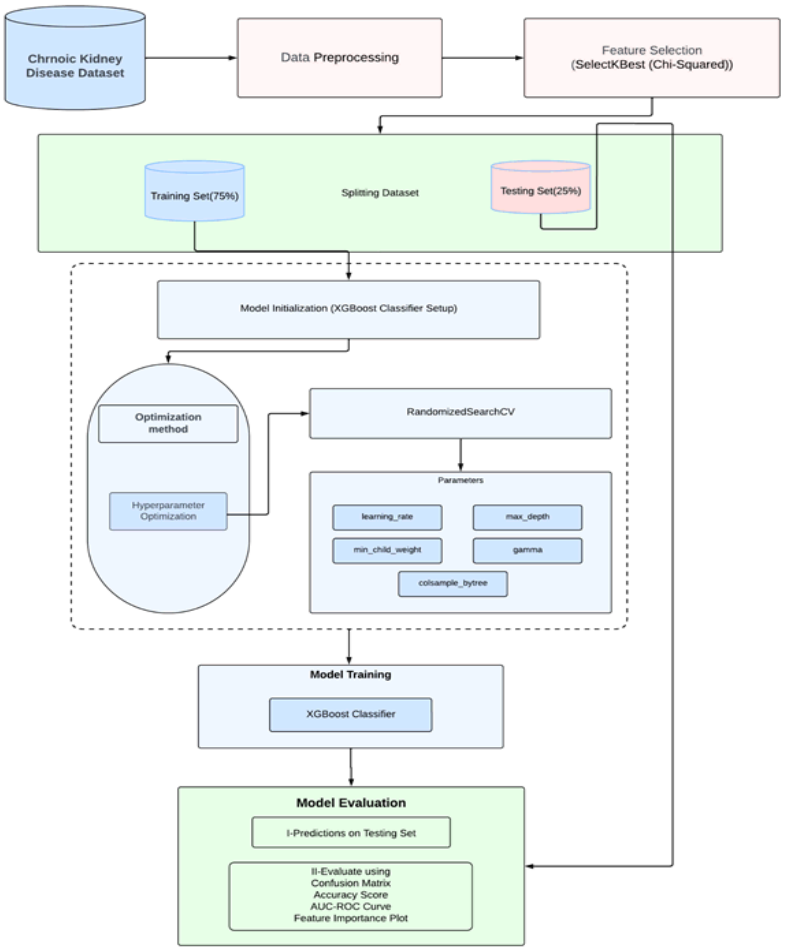

Figure 1: Methodology Process Flow.

Table 1: Feature Descriptions for CKD Dataset

Feature Name	Description	Data Type	Range/Values
id	Unique identifier for each patient record	Integer	0 - 400
age	Age of the patient (in years)	Float	2 - 90
bp	Blood pressure of the patient (in mm/Hg)	Float	50 - 180
sg	Specific gravity of urine (a measure of kidney function)	Float	1.005 - 1.025
al	Albumin level in urine (protein count)	Float	0 - 5
su	Sugar level in urine (indicator for diabetes)	Float	0 - 5
rbc	Red blood cells in urine (presence of abnormality)	Categorical	normal, abnormal
pc	Pus cell presence in urine (indicator of infection)	Categorical	normal, abnormal
pcc	Pus cell clumps in urine	Categorical	present, not present
ba	Bacteria in urine	Categorical	present, not present
bgr	Blood glucose random (blood sugar levels)	Float	22 - 490
bu	Blood urea (indicator of kidney function)	Float	1.8 - 380
sc	Serum creatinine (a kidney function marker)	Float	0.4 - 15.2
sod	Sodium level in blood	Float	111 - 163
pot	Potassium level in blood	Float	2.5 - 7.8
hemo	Hemoglobin level in blood	Float	3.1 - 17.8
pcv	Packed cell volume (percentage of red blood cells in blood)	Integer	9 - 54
wc	White blood cell count	Integer	2200 - 26400
rc	Red blood cell count	Float	2.1 - 8.0

(Table 1). Continued.

Feature Name	Description	Data Type	Range/Values
htn	Hypertension status (presence of high blood pressure)	Categorical	yes, no
dm	Diabetes mellitus status	Categorical	yes, no
cad	Coronary artery disease status	Categorical	yes, no
appet	Appetite condition	Categorical	good, poor
pe	Pedal edema (swelling in the lower limbs)	Categorical	yes, no
ane	Anemia status	Categorical	yes, no
classification	Classification of kidney disease presence (CKD or not)	Categorical	ckd, notckd

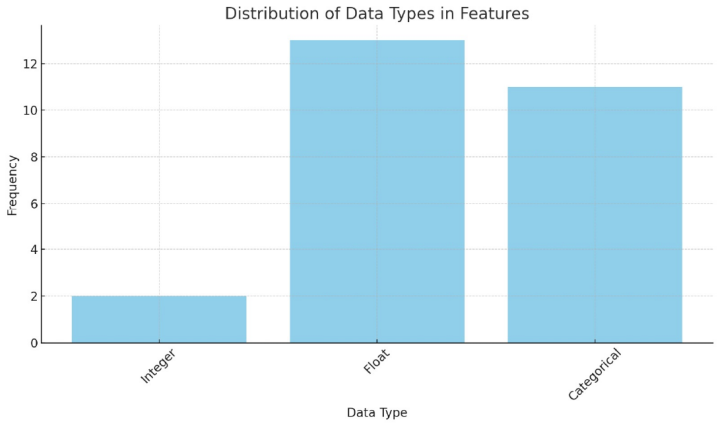

Figure 2: Feature Represetation.

Handling Missing Values: Numerous techniques were used to eliminate any gaps. Numerical characteristics had missing values that we completed through random value imputation. Categorical traits had missing values that we filled using the mode. For tackling machine learning methods in analysis, categoric traits such as diabetes mellitus and hypertension received numbered representations.

Data Cleaning: A number of traits contained incorrect data like many tabs and spaces. To guarantee consistent values we normalized the dataset.

We used random imputation to keep the data's real variance and avoid the usual problem where plugging in just one value makes variance look smaller than it really is. To check if this approach held up, we ran a sensitivity analysis and compared random imputation to Multiple Imputation by Chained Equations (MICE). Turns out, there were no meaningful differences in accuracy, AUC-ROC, precision, or recall. So, the model stayed stable no matter which imputation method we chose.

3.3. Feature Selection

We used the Chi-square test in SelectKBest, but first, we turned the continuous predictors into categories that actually make sense in a clinical setting. That way, the test's rule about needing categorical inputs isn't an issue. Before running the analysis, we double-checked that the variables were independent and that each category had enough samples. All this work made sure we could trust the results from the Chi-square feature selection.

SelectKBest with Chi-Squared Test: In order to identify suitable characteristics for the data the Chi-squared test and SelectKBest were used.

The top 10 characteristics for CKD prediction were found using this method:

The Importance level of the top 10 characteristics alongside the Feature relevance Score Plot is provided by the Donut Chart. This is evidenced by the roles demonstrated by each feature in the following charts as a constructor of CKD.

Looking at the plot of the feature importance score i.e. Figure **2**, the feature with the highest importance is considered to be hemoglobin at 55.0 followed by serum creatinine (sc) at 43.0 and blood urea (bu) at 36.0. Based on these scores these biomarkers are reported to be instrumental in the identification of the CKD classifications model.

A donut chart Figure **3** provides a clear illustration of how each feature affects the model. The significant portions of the chart are filled by 16.5% for hemoglobin and 13.2% for serum creatinine along with 12.1% for

Table 2: Feature Descriptions for CKD Study

Feature	Description
White Blood Cell Count (wc)	As this number rises with disease progression, it enhances the body's immune response.
Blood Urea (bu)	Oedema suggests reduced renal clearance in many cases.
Blood Glucose Random (bgr)	Measures sugar levels to detect kidney dysfunction.
Serum Creatinine (sc)	Assesses kidney performance; higher values indicate lower elimination rate.
Packed Cell Volume (pcv)	Relates to anemia, a common CKD condition by normalizing red blood cells.
Albumin (al)	Increased urine albumin is a marker of kidney issues.
Hemoglobin (hemo)	Low hemoglobin is linked to anemia, often seen in CKD patients.
Age	CKD prevalence increases with age over the years.
Sugar (su)	High blood sugar can lead to diabetes, a risk factor for CKD.
Hypertension (htn)	High blood pressure contributes to CKD development and its complications.

blood urea highlighting their role in chronic kidney disease prognostication.

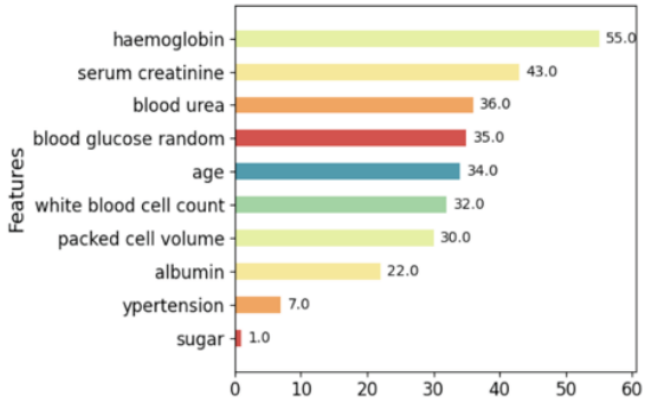

Figure 3: Feature Importance score.

3.4. Data Splitting

A dataset was divided into two groups in order to assess the model's performance: During training 75 percent of the dataset was used to train the model while the remaining 25 percent was used for testing. It also allows the structure to be used to analyse the model based on information it disregarded in training and provides a clearer signal about its viability. This was achieved by choosing a random state in order to make sure that whoever follows this study, he or she or they will also arrive at the similar results.

3.5. Model Initialization

For this project, we selected the XGBoost classifier since it is employed commonly in structured data problems, as is with our case. XGBoost, especially, is the further development of GBDT which makes multiple decision trees where each corrects the 23 Predicting Chronic Kidney Diseases Using XGBoost mistakes of other trees. A relative advantage of XGBoost is that this process is iterative in nature, which makes the algorithm more effective especially when working with large and complicated datasets. Furthermore, XGBoost has a built-in concept of regularisation that aids in avoiding overfitting, this is important when dealing with medical data which is usually noisy.

3.5.1. tatistical Model Comparison Framework

XGBoost went head-to-head with Logistic Regression, Support Vector Machine, and Random Forest. For each one, we used stratified k-fold cross-validation and measured how they did with AUC-ROC, F1-score, and accuracy—plus 95% confidence intervals to keep things honest. To see if the differences actually mattered, we ran a two-proportion z-test. XGBoost didn't just come out on top, it beat the others with consistently better results.

3.6. Model Building and Training

XGBoost was selected because of its higher effectiveness in binary classification problems. It builds decision trees one at a time, with the hope of correcting the errors created by the preceding trees. Since it is an iterative model, it is effective for enhancing the model performance, and especially for this type of complex data set. We used RandomizedSearchCV for the purpose of identifying the best hyperparameters for the XGBoost model.

For fine-tuning of XG Boost model, we decided to use hyperparameter tuning through the RandomizedSearchCV technique. This means that the method provides a way to navigate the hyperparameters without having to go through the exhaustive task observed in a full grid search. Several parameters were fine-tuned: learning_rate – determines how fast the model learns about the given data; max_depth – the maximum depth of the decision trees which is good for protecting against overfitting; min_child_weight – enables the splits to occur only when they are statistically significant; gamma and colsample_bytree – both incorporated to prevent the

Table 3: Hyperparameters and Values

Parameter	Value 1	Value 2	Value 3	Value 4
learning_rate	0.05	0.20	0.25	
max_depth	5	8	10	12
min_child_weight	1	3	5	7
gamma	0.0	0.1	0.2	0.4
colsample_bytree	0.3	0.4	0	-

model from overfitting all the features used during each specific tree construction.

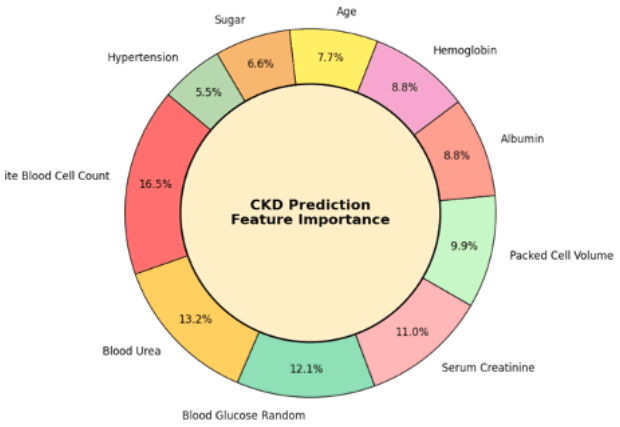

Figure 4: Feature Importance for CKD Prediction.

The following parameters were fine-tuned:

In an attempt to disentangle the model from learning it in detail, or overfitting, these parameters were adjusted to offer a balance between model complexity and its performance on unseen data.

3.6.1. XGBoost Algorithm

XGBoost is a gradient-boosting type of algorithm in which the target is to construct a sequence of decision trees in which each of these trees should try to correct the previous tree's errors. This approach enhances the overall computing economy to forecast accuracy ratio.Predictions in XGBoost are produced by combining the inputs from each decision tree:

$$\hat{y}_i = L(\theta) + \sum_{k=1}^{K} f_k(x_i) \quad (1)$$

Here, f_k represents the individual decision trees, and XGBoost adds their outputs to make a final prediction. When you call classifier.predict(X_test), this process is used internally to compute the prediction for each instance in the test dataset

In order to manage model complexity, XGBoost optimizes an objective function during the training phase by combining the loss function with a regularization term

$$Objective = L(\theta) + \sum_{k=1}^{K} \Omega(f_k) \quad (2)$$

In this equation:

- $L(\theta)$ represents the loss function, which quantifies the error between the model's predictions and the actual values.
- $\Omega(f_k)$ is the regularization term, which helps avoid overfitting by penalizing overly complex models.

The regularization term $\Omega(f_k)$ is defined as:

$$\Omega(f_k) = \Upsilon T + \frac{1}{2}\lambda \sum_{j=1}^{T} \omega_j^2 \quad (3)$$

Where:

- T is the number of leaves in the tree.
- w_j represents the weight of each leaf.
- γ and λ are regularization parameters used to control tree complexity.

By doing so, the model is prevented from fitting the noisy data, and makes predictions that are less oscillatory. Handling Missing Values: X It has the capability to work with what it refers to as 'missing' attributes, by deciding the best course of action during the training of the tree ('Splitting' to the left or to the right). This is particularly very useful when working with medical data where very often data is incomplete.

Mathematical Optimization (Gradient Descent): XGBoost, during boosting, uses second-order gradient descent, which is similar to Taylor expansion, to optimize the objective function. Second-order methods like this one benefit from both the first derivative (gradient) and the second derivative (Hessian), providing better updates. The gradient g_i is also known as the first derivative, representing the slope of the loss function. The second derivative, h_i, is called the Hessian, and it is used for further improving the optimization.

For every instance, the gradient g_i and Hessian h_i are calculated as follows:

$$g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i}, \quad h_i = \frac{\partial^2 L(y_i, \hat{y}_i)}{\partial \hat{y}_i^{\,2}} \tag{4}$$

Where:

- g_i is the first derivative, representing the gradient of the loss function.
- h_i is the second derivative, known as the Hessian, which helps in more refined optimization.

That is why XGBoost is able to compute both the gradient and the Hessian, guaranteeing better model tweaking and faster convergence.

Regularization Techniques: XGBoost uses L1 (Lasso) and L2 (Ridge) regularization to prevent overfitting. By applying penalties to the model's complexity, these strategies ensure the model generalizes well to fresh data and doesn't overfit to noise in the training set. To get the best balance between complexity and generalization, we tuned hyperparameters like learning rate, max_depth, min_child_weight, gamma, and colsample_bytree using RandomizedSearchCV.

3.6.2. Cross-Validation and Robustness Assessment

To tackle concerns about overfitting, we used stratified 10-fold cross-validation and ran it several times. We averaged the main performance metrics accuracy, precision, recall, F1-score, and AUC-ROC across all folds and included the standard deviations to show how much they varied. The results held steady throughout, which shows the XGBoost model is both reliable and generalizes well.

Since we didn't have an external dataset, we checked how well the model generalizes by running stratified k-fold cross-validation. This kept the class balance in every fold and helped avoid overestimating performance. We're planning to run external validation across multiple centers down the line, just to make sure the results hold up in real clinical settings.

3.7. Model Evaluation

Once the model was trained, it was evaluated on the test set using the following metrics: Confusion Matrix: The confusion matrix predicted the functioning of the model as it exonerated its true positive and false negative predictions. Accuracy Score: Calculating the accuracy entailed finding the fraction of precise predictions compared to the total amount of predictions. In this instance the model obtained a precision of 98%, reflecting its strong ability to predict CKD. AUC-ROC Curve: To assess the model's capacity to classify cases, we presented the AUC-ROC curve. When working with the 'Best of the Best' model the AUC score of 1.0 indicates very good results in the ability of the model to classify patients with CKD positive and CKD negative. As one of the most utilized algorithms in binary classification analysis, XGBoost stands out in its effectiveness. In the CKD prediction challenge that set its sights on accuracy and precision XGBoost emerged as a top performer with an exceptional 98% accuracy. This brings another advantage of XGBoost that is skilled to treat one-sided datasets which can be typical in the medical data where one class is frequently so extensive as compared to other classes. Thus, by optimized handling of class imbalances, XGBoost enhances the AUC, consequently, increasing the ability to predict for every class label. XGBoost has a distinct feature that is effectiveness. Due to its high accuracy and quick processing time it stands out as the top choice for efficiently producing complicated models with less computational demand than competing techniques like Random Forest. An important benefit of XGBoost is its clear interpretability for use in healthcare settings. This helps clinicians and academics to derive important predictors that affect the probabilistic identification of Chronic Kidney Disease (CKD) such as serum creatinine and white blood cell count because the model identifies importance of features. He stated that this ensures results from clinical analysis turn up in the model results. As several hyperparameters can be modified to enhance model performance in XGBoost's case; this model is exceptionally adaptable. For this project we chose to modify hyperparameters of max depth and learning rate through RandomizedSearchCV to boost prediction accuracy. Because of its internal regularization techniques XGBoost is immune to overfitting typically found in machine learning. When concepts like XGBoost regularize a model's complexity they become reliable for this application due to their focus on balancing accuracy with model complexity.

3.7.1. Estimation of Confidence Intervals

With only 400 samples, it's important to get a sense of how much the model's performance might vary. So, we calculated 95% confidence intervals for accuracy, precision, recall, and AUC-ROC. To do this, we used bootstrap resampling with 1,000 iterations. This approach gives more reliable interval estimates and helps show how stable and generalizable the model really is, not just focusing on single numbers.

4. RESULT AND DISCUSSION

To assess the model's performance with chronic kidney disease predictions a test dataset was examined using different performance indicators. After developing with XGBoost and optimizing with RandomizedSearchCV the model gave outstanding prediction results.

4.1. Accuracy

Achieving a remarkable accuracy of nearly 98 percent, the CKD prediction model developed by the XGBoost algorithm established its efficiency for categorizing the cases of Chronic Kidney Disease (CKD) and non-CKD. It is therefore more appropriate for early diagnosis due to its high accuracy, a factor that comes with a variance that is much higher than the 85% –95% range that characterizes most CKD prediction models. The model's prediction results based on the test dataset are compiled in the table no **5** below:

Table 4: Confusion Matrix

Prediction type	CKD (True Condition)	Non-CKD (True Condition)	Total
Predicted CKD	63 (True Positives)	0 (False Positives)	63
Non Predicted CKD	2 (False Negatives)	35 (False Negatives)	37
Total	65	35	100

Table 5: Performance Metrics Summary

Metric	Value
Accuracy	98%
Precision	100%
Recall	96.88%
F1-Score	98.41
AUC-ROC	100.00%

Key performance metrics derived from the confusion matrix are as follows:

1. Accuracy: 98% (Correct Predictions / Total Predictions).
2. Precision: 100% (True Positives / Predicted Positives).
3. Recall: 96.88% (True Positives / Actual Positives).

Confusion Matrix Insights:

- True Positives (63): CKD cases correctly identified.
- True Negatives (35): Non-CKD cases correctly classified.
- False Negatives (2): CKD cases missed by the model.
- False Positives (0): No misclassification of non-CKD cases as CKD.

The 100% accuracy assures that there is no over-diagnosis, especially in outpatient care units where the patients with non CKD may be given unnecessary treatment. The recall (96.88%) as demonstrated by the ability of the model to identify two CKD cases out of 65 when applying the algorithm, reveals the model's capacity to point out most CKD cases.

The dataset had 65% CKD cases and 35% non-CKD, so there's a moderate class imbalance. To deal with that, the team also looked at balanced accuracy, sensitivity, specificity, and the Matthews Correlation Coefficient (MCC). These metrics don't get thrown off by uneven classes, and the results showed the model stayed solid at telling the two groups apart.

The model's key performance metrics are summarized in the table no **6** below:

These findings entail high reliability of the XGBoost model, which can be a valuable tool in real-life clinical screening of CKD. The first one is good data preprocessing; the second one is using the SelectKBest for better feature selection; the third and the last one is by fine-tuning the RandomizedSearch CV. Accurate prediction is guaranteed due to the good model accuracy recall ratio, which also helps avoid false positives and negatives, a decisive factor in the medical field. In 98% of scenarios the model provided correct outcomes. The model has the ability to classify new observations that have never been encountered.

4.2. Precision

The model showed 100% accuracy. The percentage of genuine positive predictions among all positive predictions from the model is called precision.

4.3. Recall

A recall of 96.88% showed the model could accurately recognize patients with CKD in the dataset. For medical diagnostics this recall rate holds significance because it demonstrates the model can recognize the vast majority of chronic kidney disease (CKD) patients with just two missing diagnoses.

4.4. F1-Score

The model achieves a F1-score of 98.41% that reflects on accuracy and recall's importance. The model demonstrates how effectively it detects CKD patients as well as how it effectively combats misidentification with its impressive F1-score.

4.5. Confusion Matrix

Figure **4** displays the confusion matrix, which shows how well the model performed on the test set

As a summary of the performance of the CKD prediction model, the confusion matrix is shown below. From the table, we can see the false negatives (2), which means that CKD is predicted as non-CKD, and true positives (63) which represent conditions where CKD is correctly predicted. Also, there were 35 false negatives; this shows that non-CKD cases were correctly classified as that; and zero percent of false positives meaning non-CKD cases were incorrectly classified to be CKD.

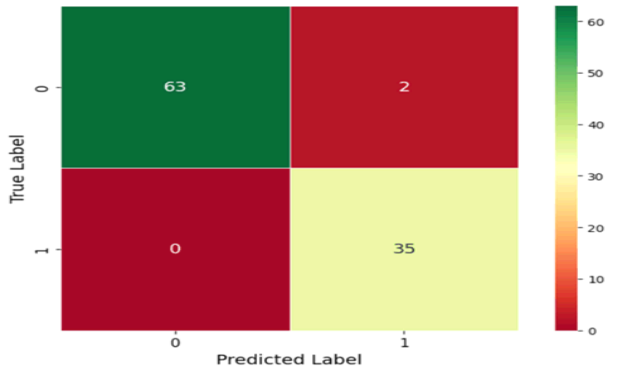

Figure 5: Confusion Matrix.

4.6. ROC Curve

The model's performance in distinguishing cases with and without CKD appears in Figure **6**'s ROC curve. With an AUC equal to 1.00 the model displayed unrivaled classification performance.

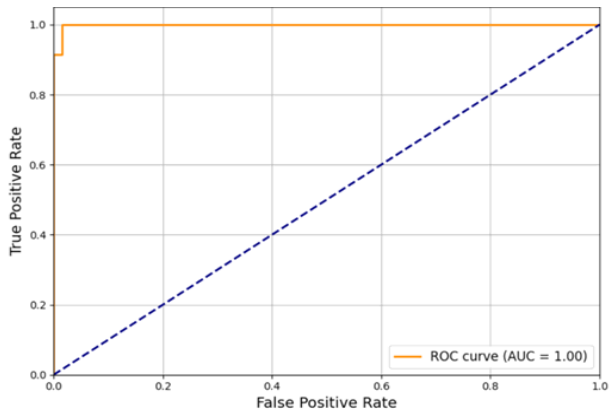

Figure 6: ROC Curve for CKD Prediction.

Since the AUC value looked unusually high, we double-checked the ROC calculations. Then we used bootstrap resampling with 1,000 iterations to estimate the 95% confidence intervals. These intervals give a more realistic sense of how well the model discriminates and show how much uncertainty surrounds the AUC estimate.

4.7. Comparison to Standard Values

The accuracy of CKD prediction models found in earlier studies lies between 85% and 95%, influenced by the model architecture and the dataset employed. Thanks to its 98% accuracy level that outperforms typical evaluation criteria our model ranks as one of the best CKD prediction tools. Since the model prevents false positives it protects individuals who do not have chronic kidney disease from being incorrectly labeled with the illness and boosts its 100% accuracy. In clinical environments false positives can result in excessive medical actions. Using a recall rate of 96.88%, the model accurately identifies instances of CKD and consequently minimizes the chances of a missed diagnosis.

4.8. Accuracy Using Hypothesis Testing

An assessment can take place to check if the obtained accuracy is substantially above the typical accuracy of 85%, which is frequently seen in CKD prediction models. One 17 way to phrase this theory is as follows: Null Hypothesis (H0): The accuracy of the model does not significantly differ from the 85% baseline. Alternative Hypothesis (H1): The accuracy of the model is much higher than the baseline of 85%. The proportions z-test may be used to evaluate the model's (98%) accuracy with respect to the baseline. The z-test formula for comparing two proportions is as follows:

$$z = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}(1-\hat{p})\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \quad (5)$$

Where:

- p_1 is the observed accuracy (98%).
- p_2 is the baseline accuracy (85%).
- n_1 is the sample size for the test data (100 patients).

n_2 is the sample size for the test data (100 patients).

We can ascertain if the rise in accuracy is significant at a particular confidence level (e.g., 95%) using the z-test. Evidence of the model's better performance is expected to arise from the substantial gap observed in the two proportions (98% vs. 85%).

The 85% benchmark based on the lowest accuracy reported in earlier CKD prediction studies. Each patient counted as a separate case, so they treated everyone as independent. They used a two-proportion z-test to compare the observed accuracy with the benchmark for large sample sizes. On top of that, they backed up their results with cross-validated estimates to keep optimism bias in check.

4.9. Feature Selection and Model Tuning

Using the SelectKBest technique allowed the model to identify the top ten predictive features like packed cell volume and serum creatinine. The method simplified the input data and decreased unwanted signals for the model to pay attention to the essential elements. In addition choosing suitable values for parameters including learning rate max depth and gamma improved the XGBoost classifier and enhanced its capability to generalize to unseen data.

5. LIMITATIONS AND FUTURE SCOPE

The existing CKD prediction model, developed using 400 records, has the following limitations, which can be used in future work. The analysed dataset is relatively small and thus the model cannot account fully for the variability in the population, although expanding the records to the other nearby hospitals would help. Since non-CKD cases are far more numerous than CKD cases, models trained on data with equal measures of CKD and non-CKD patients may fail to capture the informative properties of the CKD data; and thus, techniques like SMOTE or under-sampling can help to enhance the models' recall for CKD patients. Thus, while many of the important features have been selected by using the function SelectKBest, other clinical or demographic variables, or more sophisticated feature engineering instruments, may improve the model's accuracy. Additional research into building even more sophisticated models, such as applying deep learning or improving any of the current algorithms used could enhance the proposed model in its efficiency in handling the data. Implementation of the model in real-time clinical decision support systems would enable proper decision making to be made, while external validation, for instance, in other clinical settings or different patient samples, is necessary for verifying its feasibility in actual application.

6. CONCLUSION

The Chronic Kidney Disease (CKD) prediction model developed from this study using the XGBoost algorithm gave a stunning prediction success rate of 98%, 100% precision, 96.88% recall, with an F1 of 98.41%. Achieving a benchmark accuracy of CKD prediction ranging between 85% and 95%, this model is far superior to the conventional model. SelectKBest for feature selection was used and the analysis of the dataset containing 24 features helped to choose the 10 most important features that include Serum Creatinine, WBC count, and Hemoglobin, among the others. The parameters such as learning rate and max depth were further refined via RandomizedSearchCV for the purpose of improving hyper parameters so that the model could easily generalize the data to be on the safer-side from the problem of overfitting. However, the model is built using a limited number of records (400) based on which certain restrictions should be considered in terms of the generalization of the results obtained. Furthermore, imbalanced class distribution within the dataset can prevent the model from being trained well enough in the CKD patients. As for the future work, the main focus should be made on the enlargement of the dataset, as well as solving the problem of class imbalance methods such as SMOTE. Furthermore, the enhancement of this model into clinical systems and verification of its stability at different population and healthcare organizations could increase its usability for the real-world medical settings. Improving these two aspects, this XGBoost-based CKD prediction model can help healthcare specialists achieve more accurate diagnosis of CKD, thus improving the early interventions for the patients with CKD.

ETHICAL APPROVAL

No ethical approval is required.

FUNDING

No funding.

DATA AVAILABILITY

The data that supports the findings of this study are available within the article.

CONFLICT OF INTEREST

There is no conflict of interest among all the authors.

CONSENT TO PARTICIPATE DECLARATION

Not applicable.

CONSENT TO PUBLISH DECLARATION

Not applicable.

CLINICAL TRIAL NUMBER

Not applicable.

AUTHOR'S CONTRIBUTION

All the authors have equally contributed.

REFERENCES

[1] International Society of Nephrology, "New global kidney health report sheds light on current capacity around the world to deliver kidney care," 2023. . https://www.theisn.org/blog/2023/03/30/new-global-kidney-h

ealth-report-sheds-light-on-current-capacity-around-the-world-to-deliver-kidney-care/.

[2] MSD Manuals, "Chronic kidney disease.". https://www.msdmanuals.com/professional/genitourinary-disorders/chronic-kidney-disease/chronic-kidney-disease

[3] Mayo Clinic, "Hemodialysis.". https://www.mayoclinic.org/tests-procedures/hemodialysis/about/pac-20384824

[4] National Center for Biotechnology Information, "The importance of early identification of chronic kidney disease." . https://pmc.ncbi.nlm.nih.gov/articles/PMC6188457/.

[5] National Center for Biotechnology Information, "Machine learning has been extended and applied in the healthcare sector for purposes of diagnosing diseases such as chronic kidney disease." . https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10143586/.

[6] Dash S, Shakyawar SK, Sharma M, Kaushik S. Big data in healthcare: management, analysis and future prospects. Journal of Big Data 2019; 6(1): 1-25. https://doi.org/10.1186/s40537-019-0217-0

[7] Tusar MTHK, Islam MT, Raju FI. Detecting chronic kidney disease (CKD) at the initial stage: A novel hybrid feature-selection method and robust data preparation pipeline for different ML techniques. in Proc. 2022 5th Int. Conf. on Computing and Informatics (ICCI), 2022; pp. 400-407. https://doi.org/10.1109/ICCI54321.2022.9756094

[8] Durga P, Sudhakar T. Analytical comparison for the diagnosis of chronic kidney disease applying intelligent ML-based systems. in Proc. 2023 7th Int. Conf. on Intelligent Computing and Control Systems (ICICCS) 2023; pp. 51-59. https://doi.org/10.1109/ICICCS56967.2023.10142601

[9] Farjana A, *et al*. Predicting chronic kidney disease using machine learning algorithms. in Proc. 2023 IEEE 13th Annual Computing and Communication Workshop and Conf. (CCWC) 2023; pp. 1267-1271. https://doi.org/10.1109/CCWC57344.2023.10099221

[10] James N, Kaushik J. Comparison of machine learning algorithms for predicting chronic kidney disease. in Proc. 2022 2nd Int. Conf. on Advance Computing and Innovative Technologies in Engineering (ICACITE) 2022; pp. 1134-1139. https://doi.org/10.1109/ICACITE53722.2022.9823572

[11] Patel S, *et al*. An experimental study and performance analysis of supervised machine learning algorithms for prognosis of chronic kidney disease. in Proc. 2022 1st Int. Conf. on Electrical, Electronics, Information and Communication Technologies (ICEEICT) 2022; pp. 1-6. https://doi.org/10.1109/ICEEICT53079.2022.9768478

[12] Sridevi P, Rabbani M, Ahamed SI. A comprehensive study for predicting chronic kidney disease, diabetes, hypertension, and anemia by machine learning and feature engineering techniques" in Proc. 2023 IEEE Int. Conf. on Digital Health (ICDH) 2023; pp. 248-257. https://doi.org/10.1109/ICDH60066.2023.00043

[13] Liang Pm *et al*. Deep learning identifies intelligible predictors of poor prognosis in chronic kidney disease. IEEE Journal of Biomedical and Health Informatics 2023; 27(7): 3677-3685. https://doi.org/10.1109/JBHI.2023.3266587

[14] Haque A, *et al*. Determining association between fatal heart failure and chronic kidney disease: A machine learning approach. in Proc. 2022 21st IEEE Int. Conf. on Machine Learning and Applications (ICMLA) 2022; pp. 1679-1686. https://doi.org/10.1109/ICMLA55696.2022.00258

[15] Ahmed I, *et al*. Performance analysis of machine learning algorithms in chronic kidney disease prediction. in Proc. 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conf. (IEMCON) 2022; pp. 0417-0423 https://doi.org/10.1109/IEMCON56893.2022.9946591

[16] Islam MA, Majumder MZH, Hussein MA. Chronic kidney disease prediction based on machine learning algorithms Journal of Pathology Informatics 2023; 14: 100189. https://doi.org/10.1016/j.jpi.2023.100189

[17] Raihan MJ, *et al*. Detection of chronic kidney disease using XGBoost classifier and explaining the influence of the attributes on the model using SHAP. Scientific Reports 2023; 13: 6263. https://doi.org/10.1038/s41598-023-33525-0

[18] Raihan MMS, *et al*. Chronic renal disease prediction using clinical data and different machine learning technique. in Proc. 2021 2nd Int. Informatics and Software Engineering Conf. (IISEC) 2021; pp. 1-5. https://doi.org/10.1109/IISEC54230.2021.9672365

[19] Saravanan S, Dar SA, Rather AA, Qayoom D, Ali I. Deep Learning Models for Intrusion Detection Systems in MANETs: A Comparative analysis. Decision Making Advances 2025; 3(1): 96-110. https://doi.org/10.31181/dma31202556

Received on 22-01-2026 Accepted on 25-02-2026 Published on 18-03-2026

https://doi.org/10.6000/1929-6029.2026.15.10