# Effective Classification of Chronic Kidney Disease Using Extreme Gradient Boosting Algorithm

Ramya Asalatha Busi, Vasireddy Venkatadri Institute of Technology, India\*

M. James Stephen, Wellfare Institute of Science Technology & Management, India.

# ABSTRACT

With a high rate of morbidity and mortality, chronic kidney disease is a global health issue that also causes other diseases. Patients frequently overlook the condition because there aren't any evident symptoms in the early stages of CKD. An efficient and effective Extreme gradient boosting method for the early diagnosis of kidney illness has been proposed in this paper to explore the capability of various machine learning algorithms. DenseNet can extract a variety of features such as vector features. After that feature extraction phase, the data are fed into the feature selection phase. The features are selected based upon the Improved Salp swarm Algorithm (ISSA). The proposed CKD classification method has been simulated in PYTHON. Utilizing the CKD dataset from the UCI machine learning resources, the dataset is then tested. Sensitivity, accuracy, and specificity are the performance metrics used for the proposed CKD classification approach. The results of the experiments demonstrate that the proposed approach outperforms the present state-of-the-art method in classifying CKD.

#### **KEYWORDS**

Chronic Kidney Disease (CKD), CKD dataset, DenseNet, Improved Salp Swarm Algorithm (ISSA), Machine Learning, Classification, Outperform

## INTRODUCTION

A serious death and illness problem is enforced by CKD, often known as CKD (Ammirati, 2020). It is one of the non-communicable diseases with one of the fastest expanding epidemiologies. CKD is a condition where the kidneys lose their ability to filter blood, allowing the body's waste products to build up within and leading to other health issues (Henry & Lippi, 2020; Jankowski et al., 2021; Byrne & Targher, 2020). Because clean, pure blood aids in the improved functioning of the body's organs, it is extremely vital to maintain healthy kidney function. Over many years, this harm develops (Portolés et al., 2020). Kidney function decreases as damage increases, which is bad for the body. In developing and underdeveloped nations, it is increasingly becoming a serious hazard. Diseases like diabetes and high blood pressure are the main causes of its onset (FIDELIO-DKD Investigators et al., 2021; Guzzi et al., 2019; Bidin et al., 2019). In addition to obesity, heart disease and a family history of CKD, other risk factors contribute to CKD.

DOI: 10.4018/IJSI.315732

\*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

Testing may be the only way to determine whether the patient has the renal disease because, in its initial stages, CKD has no manifestations (Connaughton et al., 2019; Paik et al., 2022). Early identification of CKD in its initial phases can enable the patient to receive appropriate treatment and halt the development of ESRD (Zhuang et al., 2021). It is suggested that everyone with a risk factor for CKD, such as a family history of renal failure, high blood pressure, or diabetes, should be examined annually (Mihai et al., 2018). This illness is characterized by a gradual decline in renal function, which leads to a full loss of renal function in the end.

Early on, CKD does not manifest any overt symptoms. As a result, the disease might not be identified until the kidney has lost about 25% of its functionality (Kumar et al., 2022). Additionally, CKD affects the human body globally and has a high rate of morbidity and mortality. Cardiovascular disease may develop as a result (Han et al., 2020). A pathologic illness that progresses and cannot be reversed is CKD. Therefore, early detection and diagnosis of CKD are crucial for allowing patients to begin treatment and halt the disease's progression (Chen et al., 2019).

Diabetes, high blood pressure, and cardiovascular disease (CVD) are risk factors for CKD patients. Patients with CKD experience side effects, particularly in the late stages, which weaken the immunological and nervous systems (Sharma, 2018; Siraj, 2019). Patients may be in advanced stages in developing nations, necessitating dialysis or kidney transplants. Glomerular filtration rate (GFR), a measure of kidney function, is used by medical professionals to identify renal illness. Age, blood test results, gender, and other patient-related characteristics are taken into account while calculating GFR. Doctors can divide CKD into five stages based on the GFR value (Wang et al., 2019).

Machine learning describes a computer program that evaluates and extrapolates task-related data to determine the traits of the associated pattern (Gunasundari et al., 2018). This technology is capable of making cost-effective and accurate diagnoses of diseases, making it a potentially useful tool for CKD diagnosis (Calderon-Margalit et al., 2018). With the advancement of information technology, it has evolved into a new kind of medical instrument and has a wide range of potential applications.

The following are the contributions of the proposed research:

- The preprocessing stage includes checking for uneven data and estimating missing values as well as removing noise like outliers and normalization.
- Then the pre-processed data is given as the input into the DenseNet for feature extraction. In this process, the layers of DenseNet are utilized to extract the important features of the input data.
- The features are selected based on the Improved Salp swarm Algorithm.
- Then the selected features are passed into the classification phase. An extreme gradient boosting algorithm is used for classification. Whether the data is Ckd or not CKD.

The remaining portions of the paper are formatted as follows: The research on CKD classification will be covered in the part that follows. The suggested approach is explained in Section part 3. Portion 4 discusses the evaluation criteria and categorization techniques used. The research's findings are summarized in Part 5. In Part 6, comes to a close.

#### LITERATURE REVIEW

In this section, we review some existing Machine learning approaches for diagnoses of CKD.

Elhoseny et al. (2019) introduced DFS with the D-ACO algorithm for CKD. Before building the ACO-based classifier, the suggested intelligent system via DFS removes unnecessary or duplicate features. The effectiveness of the suggested algorithm is assessed utilizing a CKD dataset, and a comparison is also done with the other approaches. The suggested D-ACO algorithm exceeded the other approaches with increased categorization effectiveness in a number of ways when compared to the current approaches.

Jerlin Rubini and Perumal (2020) presented MKSVM and FFOA for disease classification. That is used to pick the best features. For the goal of classifying medical data, the processed and chosen features from the dataset are sent to the presented approach. The provided approach produces improved accuracy when compared to current approaches.

Ma et al. (2020) introduced the HMANN for the earlier detection and characterization of chronic renal failure on the IoMT platform. The suggested HMANN is categorized as an MLP and SVM using a Back Propagation (BP) technique. The strategy that is being shown helps to segment the renal image and eliminates noise. The suggested HMANN approach for kidney segmentation provides high accuracy while greatly lowering the time to outline the contour.

Chen et al. (2020) introduced the AHDCNN for the identification of kidney disease. The numerous sub-types of lesions in kidney cancer are distinguished from CT scans using a deep learning algorithm. First, the acquired data will be examined, along with any missing values. Utilizing the learning and activation mechanisms effectively is the best method to prevent kidney disease. These advances in machine learning provide a promising framework for finding clever solutions that can show their predictive relevance outside the context of kidney disease.

Linear regression (LR) and neural networks were introduced by Abdelaziz et al. (2019). (NN). Critical factors that have an impact on CKD are identified using LR. NN is employed to forecast CKD. Out of the twenty-four parameters that have an impact on CKD, thirteen are crucial, according to the trial data, and a hybrid intelligent model has a 97.8% accuracy rate.

ANN and LR were recommended by Ahmed and Alshebly (2019) for the prediction of chronic renal disease. According to the experimental findings, the ANNs classifier performs better than the LR method. According to the elements that had the biggest influence on the data of patients with chronic renal illness, the variables creatinine and urea are the most significant and effective variables when applying the two approaches. Using FCM clustering, which is efficient in mining complicated data with fuzzy correlations among members, Kunwar et al. (2019) demonstrated analysis and identification of Chronic Kidney Disease.

## **PROPOSED METHODOLOGY**

The proposed chronic kidney disease classification employing the Extreme gradient boosting algorithm is discussed in this section. The preprocessing processes are used to increase the classification efficiency even more. The raw data were passed through the preprocessing phase. Preprocessing tasks included checking for imbalanced data and approximating missing values as well as removing noise like outliers and normalization. Then the preprocessed raw data is fed into the feature extraction phase. The features can be extracted based on DenseNet. In this process, the layers of DenseNet are used to extract the important vector features of data. With the help of the Improved Salp swarm Algorithm, the features are selected, followed by classification is used. After the feature selection process, the Extreme gradient boosting algorithm is utilized to classify the CKD.

In our strategy, the work is processed based on four phases such as preprocessing, feature extraction, feature selection and classification. Figure 1 depicts the general structure of the proposed methodology.

Figure 1. Architecture diagram of proposed Methodology



# PREPROCESSING

The most crucial step in obtaining the required features and classification levels is preprocessing the data. The data's quality must be good to provide efficient performances. The dataset needed to be cleaned up during preprocessing because it had outliers and noise. Estimating missing values, removing noise like outliers, normalizing, and verifying for imbalanced data were all parts of the preprocessing step. When patients are undergoing tests, it is possible for some measures to be missed, leading to missing numbers. 158 occurrences in the dataset were complete, whereas the rest instances had missing values.

## **Missing Values**

The dataset comprised 158 cases that were fully finished, and the rest instances had missing values. The simplest way to deal with missing values is to remove the record, although this approach is problematic for small datasets. Instead of deleting records, we can apply algorithms to estimate the missing values. One the statistical measures, such as median, mean, and standard deviation, can be used to calculate the missing values for numerical features. But utilizing the mode technique, which substitutes the missing value with the most frequent value of the characteristics, it is possible to evaluate the missing values of nominal features.

# FEATURE EXTRACTION

A robust diagnostic model cannot be built since the vector characteristics must be extracted to exclude features that are irrelevant and unhelpful for prediction. The pre-processed data is then fed into the DenseNet as input so that features can be extracted. The vector features of the data are extracted in this step using the DenseNet layers. A feed-forward neural network like DenseNet ensures maximal information flow across layers by directly connecting each layer to all succeeding layers. The dense block, transition layer, GAP, and convolutional layer are the primary components of the DenseNet structure.

Every layer passes its feature maps to all succeeding layers and receives extra input from all earlier layers. Concatenation is used to merge the resulting feature maps from the previous layer with those from the current layers. Each layer of the network is connected to all of the successive layers,

and these networks are known as dense nets. Compared to conventional CNNs, this model requires fewer parameters. Additionally, it lessens the overfitting issue that smaller malware training sets have.

## **Dense Block**

Considering the input data  $x_0$  that the suggested convolutional network processes. Every layer of the network, which consists of N layers, performs a nonlinear transformation called  $F_n$  (.). Assume layer n is made up of the feature maps from all layers of convolution that came before it. Cascaded feature maps from layers 0 to n - 1 from the input data are shown  $x_0, \ldots, x_{n-1}$ . As a result, this structure is connected to an N layer network via N (N + 1)/ 2 links. The n<sup>th</sup> layer's output can be calculated using:

$$x_{n} = F_{n}\left( \left[ x_{0}, \dots, x_{n-1} \right] \right)$$
(1)

while  $F_n$  (.) is the composites functional of Batch Normalization (BN)- Rectified Linear (ReLU) Units,  $[x_0, ..., x_{n-1}]$  is a fusion of feature maps generated from 0 to n – 1 layer, and  $x_n$  is the present nth layer. ReLU, 3 × 3 convolutions, and BN are the subsequent processes in the transition layer (Conv). If the dimensions of the feature maps alter, the fusion procedure is not practical. The layers with various feature map sizes are consequently down sampled. Among two adjacent Dense Conv blocks, transition layers made up of 1×1 Conv and 2 2 ×average pooling operations are provided. Seven by seven Conv blocks with a stride of two make up the first Conv layer.

BN is a widely accepted standard technique for achieving quick convergence and improved neural network classification capacity. The following output  $\hat{x}_r$  can be provided for a short batch of data from B:

$$x_{BN} = \frac{\hat{x}_r - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}$$
(2)

$$\hat{x}_r = \gamma x_{BN+\beta} \tag{3}$$

## **Transition Layer**

To speed up training and reduce the number of features, the transitional layer was used because dense connections enhanced network parameters. For every transitional layer in the tests, a  $1 \times 1$  Conv layer for lowering the data size and a  $2 \times 2$  average pooling were used. To guarantee the invariance of feature shift and scale, the input feature maps were divided into many, non-overlapping regions by the pooling layer, which then estimated the average value for each region.

As a result, the network processing was reduced while the important features were maintained, producing a more reliable model.

## Gap Layer

We processed the feature maps with the GAP after combining dense blocks and transitional layers, and then we fed the processed feature maps into the next Softmax layer for categorization selections. Whenever GAP sets the window to the same dimensions as the feature map, the featured maps can be easily recognized as probability maps for categorization. This emphasizes the correlations among

feature maps and categories. The training parameters were effectively decreased by generating the corresponding feature vector by averaging every feature map.

## Softmax Classifier

The localization of MI was made possible in the current work by the training of the Softmax classifier using the feature vectors generated by DenseNet. For the dataset  $\left\{ \left( x_{GAP}^{(i)}, y^{(i)}; i \in 1...N, y^{(i)} \in 0, ...k - 1 \right) \right\}$  where  $x^{(i)}$  GAP is the ith feature vectors of the input sample The categorization probability for every sample is provided by:

$$p(y^{(i)} = j \mid x_{GAP}^{(i)}; \theta) = \frac{e^{\theta_j^T} x_{GAP}^{(i)}}{\sum_{j=1}^k e^{\theta_j^T} x_{GAP}^{(i)}}$$
(4)

while  $p(y^{(i)} = j | x_{GAP}^{(i)}; \theta)$  denotes the likelihood that the GAP falls under the jth category, which is identical to the likelihood that the GAP falls under one of the several types of MI. The following provides the Softmax classification function:

$$h_{\theta}\left(x_{GAP}^{(i)}\right) = \begin{bmatrix} p(y^{(i)} = 1 \mid x_{GAP}^{(i)}; \theta \\ p(y^{(i)} = 2 \mid x_{GAP}^{(i)}; \theta \\ \vdots \\ \vdots \\ p(y^{(i)} = k \mid x_{GAP}^{(i)}; \theta \end{bmatrix} = \frac{1}{\sum_{j=1}^{k} e^{\theta_{j}^{T}} x_{GAP}^{(i)}} \begin{bmatrix} e^{\theta_{1}^{T} x_{GAP}^{(i)}} \\ e^{\theta_{2}^{T} x_{GAP}^{(i)}} \\ \vdots \\ e^{\theta_{2}^{T} x_{GAP}^{(i)}} \end{bmatrix}$$
(5)

while  $\theta$  is the parameter and  $\sum_{j=1}^{k} e^{\theta_{j}^{T} X_{GAP}^{(j)}}$  is the operation of the likelihood normalization.

## FEATURE SELECTION

The choice of features will have a big impact on how accurate and complex the classifier is during judgment. Feature selection is used to increase insight into the profusion of data while also shortening the computing time and complexity of the prediction model. The improved salp swarm method provides a basis for feature selection.

## Salp Swarm Algorithm (SSA)

The SSA is a heuristic algorithm that draws inspiration from the foraging behavior of slap swarms. In SSA, populations are split into two groups: leaders and followers. A leader location upgrade operator and a follower location upgrade operator make up the majority of SSA. The operator for updating the leader position is:

$$x_{1}^{t+1} = \begin{cases} x_{best}^{t} + c_{1} \left( \left( u_{b} - l_{b} \right) c_{2} + l_{b} \right) c_{3} \ge 0 \\ x_{best}^{t} - c_{1} \left( \left( u_{b} - l_{b} \right) c_{2} + l_{b} \right) c_{3} < 0 \end{cases}$$
(6)

$$c_1 = 2 \exp\left(-\left(\frac{4t}{t_{max}}()^2\right)()\right) \tag{7}$$

while ub and lb stand for the search space's upper and lower limits, correspondingly, c2 and c3 stand for two random values, and tmax signifies the number of iterations, xt best represents the location belonging to the greatest food source at the t-th iteration. One way to express the follower's position upgrade operator is as follows:

$$x_{i}^{t+1} = \frac{1}{2} \left( x_{i}^{t} + x_{i-1}^{t} \right), \left( i \ge 2 \right)$$
(8)

#### An Improved Slap Swarm Algorithm (ISSA)

Despite SSA having demonstrated its ability to be applied to real-world issues, the algorithm also has the drawback of being susceptible to local optimal solutions. In light of this, this work develops an ISSA by combining the chaotic local search (CLS) and Levy's flight (LF) strategies. The LF technique can be applied following Levi's distribution, which is usually assumed:

$$L(Z) \approx Z^{-1-\beta}, 0 < \beta \le 2 \tag{9}$$

$$Z = \frac{A}{|B|^{1/\beta}}, A \approx N(0, \sigma^2), B \approx N(0, \sigma^2)$$
(10)

$$\delta^{2} = \left\{ \frac{\Gamma\left(1+\beta\right)}{\beta \Gamma\left(\left(1+\beta\right)/2} \cdot \frac{\sin(\pi\beta/2)}{2^{(\beta-1)/2}} \right\}^{2/\beta}$$
(11)

z being the step size;  $\beta$  is the Levy index capable of regulating stability.  $A \approx N(0, \sigma^2)$  Denotes a sample taken from a Gaussian distribution, and its average and standard deviation are both zero and  $\delta^2$ , accordingly.  $\Gamma(\cdot)$  includes the Gamma function.

The CLS strategy is best described as:

$$v_{k+1} = 4v_k \left(1 - v_k\right), k = 1, 2, \dots n \left(v_1 \in \left(0, 1\right), v_1 \neq 0.25, 0.5, 0.75\right)$$
(12)

$$x_{new}^{t+1} = (1-S)x^{t+1} + S(l_b + (u_b - l_b)v_{k+1})$$
(13)

$$S = 1 - \left(\frac{t-1}{t}\right)^m \tag{14}$$

where n is the number of steps in a random local search. The shrinking speed is managed by m. To obtain ISSA, integrate the LF and CLS strategies into the SSA. The following stages can be used to divide up ISSA implementation:

Phase 1: Initialization of the population:

$$x^1 = \left(u_{\scriptscriptstyle b} - l_{\scriptscriptstyle b}
ight)rand + l_{\scriptscriptstyle b}$$

a random number, rand, is used.

**Phase 2:** Updating the positions of the leading salp and follower salp using Equations (11) and (13). **Phase 3:** Adopting the LF approach.

Upgrade the population using Equations (11) through (13), and then record the ideal response.

- **Phase 4:** CLS approach implementation. A chaotic local search will be conducted by the phase threeacquired optimal solution. There are 8000 steps in a CLS. It should be emphasized that once a superior outcome is discovered using the CLS technique, CLS is terminated.
- **Phase 5:** Ending the process. We additionally take into account two termination conditions to help ISSA converge to the global optimum:

Criteria 1: Completing the maximum number of iterations.

**Criteria 2:** Ensuring that the algorithm's objective function value changes by less than 10-6 after 50 iterations.

# CLASSIFICATION

The final step of a model, categorization, is to predict the label. The most popular machine learning approach, the Extreme gradient boosting algorithm, is summarized in this portion.

## **Extreme Gradient Boosting Algorithm**

The distributed gradient boosting algorithm known as XGBoost, sometimes known as extreme gradient boosting, has been developed to be very effective, adaptable, and portable. A group of classification or regression trees make up the decision tree ensemble-based XGBoost. It was established as an improved version of the gradient boosting technique and is a supervised machine learning approach based on ensemble learning. By aggregating the predictions of weak learners, the XGBoost algorithm uses additive approaches to create an effective learning approach. The XGBoost classifier avoids the overfitting issue and maximizes the use of computational resources in addition to its speed and great performance. These benefits come from the objective functions being made simpler so that they may be executed in parallel during the training phase and allow for the integration of regularization and predictive terms.

The first learner is fitted to the complete data according to the steps of the XGBoost algorithm. The second learner is then adjusted to include the previous learner's mistakes. Until a termination criterion is fulfilled, this process is continued, and when it is, the sum of all learners' predictions becomes the final prediction model. Equation (6) depicts the prediction procedure at the next stage:

$$f_{i}^{(t)} = \sum_{n=1}^{t} f_{n}\left(x_{i}\right) = f_{i}^{(t-1)} + f_{t}\left(x_{i}\right)$$
(15)

To begin with, the objective function is indicated as:

$$obj = \sum_{i=1}^{n} l\left(y_{i}, \hat{y}_{i}^{(t)}\right) + \sum_{i=1}^{t} \Omega(f_{i})$$
(16)

This equation has three variables: n, l, and  $\Omega$ , which stand for the number of trees, training loss function, and regularization term, respectively. The XGBoost increases the loss function to the second order and gets rid of all constants to accomplish the goal that has been set for step t. As a result:

$$L^{(t)} = \sum_{i=1}^{n} \left[ g_i f_t \left( x_i \right) + \frac{1}{2} h_i f_t^2 \left( x_i \right) \right] + \Omega \left( f_t \right)$$
(17)

while the definitions of the gi and hi are:

$$\begin{cases} g_{i} = \partial_{\hat{y}_{i}\left(t-1\right)\left(y_{i}, \hat{y}_{i}^{\left(t-1\right)}\right)} \\ h_{i} = \partial_{\hat{y}_{i}^{\left(t-1\right)}}^{2} l\left(y_{i}, \hat{y}_{i}^{-1}\right) \end{cases}$$
(18)

According to the decision rules for a particular tree, Ij is the instance set divided into the j-th leaf node. The score value for a tree's quality can be calculated using the formula (6). They also specified the point increase that results from splitting a leaf into two leaves:

$$Gain = \frac{1}{2} \left[ \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma$$
(19)

This equation is made up of the scores on the new left and right leaves, the original leaf's score, and the additional leaf's score after regularization. By scanning from left to right to obtain all feasible split options, we can quickly choose the best split by the highest Gain value:

$$\alpha\left(f\right) = \delta T + \frac{1}{2}\,\sigma\mu^2\tag{20}$$

while  $\sigma$  is the regularization parameter,  $\mu$  is the leaf node score vector, and  $\delta$  is the minimal loss required to further divide the leaf node.

# **RESULT AND DISCUSSION**

To illustrate the conclusion, using a benchmark dataset to compare the proposed method to existing methodologies in terms of sensitivity, specificity, precision and accuracy. The materials and metrics that were employed to achieve the intended results will be described in this paper. The proposed experiment's performance was evaluated in PYTHON using medical database. On an Anaconda navigator-equipped Windows 10 computer with 16 GB RAM and an Intel Ci7 64-bit processor, we trained and validate the proposed model. Tensor flow is used as the backend for all simulations, which are run on Keras.

# DATASET DESCRIPTION

The UCI, which was gathered from hospitals and donated, is where the CKD data set that was used in this work was found. 400 samples make up the data collection. Every sample in this data set has 24 predictive factors, including a categorical response variable. Every class has two values: non-CKD and CKD (example with CKD). 250 of the 400 samples are classified as having CKD, while 150 are classified as not having CKD. It is crucial to note that the data contains a significant number of missing values. Table 1 contains a list of each variable's specifics.

Variables	Explanation	Scale	Class	Missing Rate
Age	Age	Age in years	Num	2.25%
su	Serum creatinine	In mgs/dI	Num	4.25%
bp	Blood pressure	In mm/Hg	Num	3%
sod	Sodium	In mEq/L	Num	21.75%
al	Albumin	(0,1,2,3,4,5)	NOM	11.5%
pcv	Packet cell volume	-	Num	17.75%
pcc	Pus cell clumps	(present, not present)	NOM	1%
rbcc	Red blood cell count	In millions/cmm	Num	32.75%
ba	Bacteria	(present, not present)	NOM	1%
sg	Specific Gravity	(1.005,1.010, 1.015, 1.020, 1.025)	NOM	11.75%
bu	Blood urea	In mgs/dI	Num	4.75%
pc	Pus cell	(normal, abnormal)	NOM	16.25%
pot	Potassium	In mEq/L	Num	22%
rbc	Red Blood Cells	(normal, abnormal)	NOM	38%
hemo	Hemoglobin	In gms	Num	13%
su	Sugar	(0,1,2,3,4,5)	NOM	12.25%
htn	Hypertension	(yes, no)	NOM	0.5%
wbcc	White blood cell count	In cells/cumm	Num	26.5%

#### Table 1. Dataset description

Table 1 contniued on next page

#### Table 1 continued

Variables	Explanation	Scale	Class	Missing Rate
dm	Diabetes mellitus	(yes, no)	NOM	0.5%
cad	Coronary artery disease	(yes, no)	Class	0.5%
ane	Anemia	(yes, no)	Num	0.25%
pe	Pedal edema	(yes, no)	Num	0.25%
class	Class	(ckd, not CKD)	Num	0%
appet	appet	(good, poor)	Num	0.25%
bgr	Blood glucose random	In mgs/dI	Num	11%

## METRICS FOR EVALUATION OF THE MODEL

This stage involved assessing each technique's effectiveness to decide which could produce the best outcomes. Each approach used in this study was examined using the metrics of sensitivity, accuracy, and specificity from the confusion matrix. The True Positive is represented as TP, False Positive is represented as FP, True Negative is denoted as TN, and False Negative entries in the confusion matrix (FN).

#### Accuracy

The maximum number of positive outcomes divided by the maximum number of instances is used to calculate a model's accuracy:

$$Accuracy = \frac{TN + TP}{FP + TN + TP + FN}$$
(21)

## Precision

By evaluating the actual positive effects of the projected ones, checks the model's accuracy. The ratio of accurately predicted positive items to all predicted things is:

$$Prescision = \frac{TP}{TP + FP}$$
(22)

## Recall

Generated is the total number of actual positive values that the model noted and categorized as positive:

$$Recall = \frac{TP}{TP + FN}$$
(23)

# F1-Score

Precision and recall are two functions of the F1 score. A precise-recall balance is required, in which case the balance is determined:

$$F1 = 2 \times \frac{precision \times Recall}{Precision + Recall}$$
(24)

# **Evaluation Performances**

The performances can be compared with the existing approaches like SVM, KNN, PNN and decision tree.

## Table 2. Performance evaluation comparison with existing approaches

Classifiers	SVM	Decision Tree	KNN	Proposed
Accuracy	96.67%	99.17%	98.33%	99.29%
Precision	92%	98.79%	98.65%	99.17%
Recall	94.74%	98.68%	97.37%	98.97%
F1-Score	97.30%	99.34%	98.67%	99.65%

#### Figure 2. Accuracy performance over proposed with existing approaches



Performance evaluation comparison with existing approaches is shown in table 2. Comparison can be made with the approaches like SVM, decision tree, KNN. When differentiating the accuracy SVM gain 96.67%, the Decision tree yield 99.17%, KNN gains 98.33% and our proposed approach yield 99.29%. Accuracy performance over proposed with existing approaches is represented in figure 2.

Figure 3. Precision performance over proposed with existing approaches



When differentiating the precision seen in figure 3, SVM gain 92%, the Decision tree yield 98.79%, KNN gains 98.65% and our proposed approach yield 99.17%. Precision performance over proposed with existing approaches.

#### Figure 4. Recall performance over proposed with existing approaches



When differentiating the precision in figure 4, SVM gain 94.74%, Decision tree yields 98%, KNN gains 97.37% and our proposed approach yield 98.97%. Recall performance over proposed with existing approaches.

#### Figure 5. F1-Score performance over proposed with existing approaches



When differentiating the F1-score in figure 5, SVM gain 97.3%, Decision tree yields 99%, KNN gain 98.67% and our proposed approach yield 99.65%. F1-Score performance over proposed with existing approaches.

Name of the dataset	Accuracy (%)	Sensitivity (%)	Specificity (%)
Cleveland data	96.03	98.7	92.80
Hungarian data	93.19	91.48	96.22
Switzerland data	95.12	97.6	94.78
Proposed dataset	99.16	98.96	98.37

#### Table 3. Performance of the suggested technique with different datasets

#### Figure 6. Performance of the suggested technique with different datasets



Table 3 displays the estimated values that were derived for four dataset records. These indicators can be used to forecast how effectively a model for classifying medical data will be generated. Four datasets accuracy percentages are 96.03%, 93.19%, 95.12%, and 99.16%. The four datasets sensitivity values are 98.7%, 91.48%, 97.6%, and 98.96%. 92.80%, 96.22%, 94.78%, and 98.37% are the specificity values for the four datasets. Figure 6 shows the performance of the suggested technique with different datasets.



#### Figure 7. Training and Validation Accuracy

When comparing training and validation accuracy (table 4) with the existing approaches our proposed approach yield a greater accuracy which is depicted in figure 7.

Approaches	Accuracy	Execution Time
SVM	60.7%	0:00:40
RBF	87%	0:00:60
PNN	96.7%	0:00:22
Proposed	99.06%	0:00:11

#### Table 4. Accuracy and Execution Time comparison with existing approaches

#### Figure 8. Performance Evaluation of Accuracy



Performance Evaluation of Accuracy is represented in figure 8. The Overall performances can be compared with the existing approaches like SVM, RBF and PNN. 60.7% of accuracy in SVM, 87% of accuracy in RBF, 96.7% of accuracy in PNN and the proposed approach yield a greater solution which is 99.06%. Performance Evaluation of Execution time is shown in figure 9.

#### Figure 9. Performance Evaluation of Execution time



# CONCLUSION

The Extreme gradient boosting algorithm is described in this research for the classification of chronic kidney disease (CKD). On four scale datasets—the Cleveland dataset, the Hungarian dataset, the Switzerland dataset, and the CKD dataset—the anticipated work's outcome was evaluated by sensitivity, specificity, and accuracy. When differentiated from the existing RBF, PNN, and SVM, our proposed CKD categorization approach obtained the highest classification performance and also acquires maximum sensitivity and specificity. The suggested method thus achieves the kidney chronic dataset's maximum classification accuracy value of 99.06 percent. In future research projects, the clustering technique will be employed to increase classification accuracy and decrease instances of the wrong categorization.

# ACKNOWLEDGMENT

We declare that this manuscript is original, has not been published before and is not currently being considered for publication elsewhere.

# **CONFLICTS OF INTEREST**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

# FUNDING

Not applicable

# AVAILABILITY OF DATA AND MATERIAL

Not applicable

## CODE AVAILABILITY

Not applicable

# **AUTHORS' CONTRIBUTIONS**

The author confirms sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

# **ETHICS APPROVAL**

This material is the authors' own original work, which has not been previously published elsewhere. The paper reflects the authors' own research and analysis in a truthful and complete manner.

## REFERENCES

Abdelaziz, A., Salama, A. S., Riad, A. M., & Mahmoud, A. N. (2019). A machine learning model for predicting of chronic kidney disease based internet of things and cloud computing in smart cities. In *Security in smart cities: models, applications, and challenges* (pp. 93–114). Springer. doi:10.1007/978-3-030-01560-2\_5

Ahmed, R. M., & Alshebly, O. Q. (2019). Prediction and factors affecting of chronic kidney disease diagnosis using artificial neural networks model and logistic regression model. *Iraqi Journal of Statistical Sciences*, *16*(28), 140–159. doi:10.33899/iqjoss.2019.164186

Ammirati, A. L. (2020). Chronic kidney disease. *Revista da Associação Médica Brasileira*, 66(suppl 1), s03–s09. doi:10.1590/1806-9282.66.s1.3 PMID:31939529

Bidin, M. Z., Shah, A. M., Stanslas, J., & Seong, C. L. T. (2019). Blood and urine biomarkers in chronic kidney disease: An update. *Clinica Chimica Acta*, 495, 239–250. doi:10.1016/j.cca.2019.04.069 PMID:31009602

Byrne, C. D., & Targher, G. (2020). NAFLD as a driver of chronic kidney disease. *Journal of Hepatology*, 72(4), 785–801. doi:10.1016/j.jhep.2020.01.013 PMID:32059982

Calderon-Margalit, R., Golan, E., Twig, G., Leiba, A., Tzur, D., Afek, A., Skorecki, K., & Vivante, A. (2018). History of childhood kidney disease and risk of adult end-stage renal disease. *The New England Journal of Medicine*, *378*(5), 428–438. doi:10.1056/NEJMoa1700993 PMID:29385364

Chen, D. Q., Cao, G., Chen, H., Argyopoulos, C. P., Yu, H., Su, W., Chen, L., Samuels, D. C., Zhuang, S., Bayliss, G. P., Zhao, S., Yu, X.-Y., Vaziri, N. D., Wang, M., Liu, D., Mao, J.-R., Ma, S.-X., Zhao, J., Zhang, Y., & Zhao, Y. Y. et al. (2019). Identification of serum metabolites associating with chronic kidney disease progression and anti-fibrotic effect of 5-methoxytryptophan. *Nature Communications*, *10*(1), 1–15. doi:10.1038/ s41467-019-09329-0 PMID:30931940

Chen, G., Ding, C., Li, Y., Hu, X., Li, X., Ren, L., & Xue, W. (2020). Prediction of chronic kidney disease using adaptive hybridized deep convolutional neural network on the internet of medical things platform. *IEEE Access: Practical Innovations, Open Solutions*, 8, 100497–100508. doi:10.1109/ACCESS.2020.2995310

Connaughton, D. M., Kennedy, C., Shril, S., Mann, N., Murray, S. L., Williams, P. A., Conlon, E., Nakayama, M., van der Ven, A. T., Ityel, H., Kause, F., Kolvenbach, C. M., Dai, R., Vivante, A., Braun, D. A., Schneider, R., Kitzler, T. M., Moloney, B., Moran, C. P., & Hildebrandt, F. et al. (2019). Monogenic causes of chronic kidney disease in adults. *Kidney International*, *95*(4), 914–928. doi:10.1016/j.kint.2018.10.031 PMID:30773290

Elhoseny, M., Shankar, K., & Uthayakumar, J. (2019). Intelligent diagnostic prediction and classification system for chronic kidney disease. *Scientific Reports*, 9(1), 1–14. doi:10.1038/s41598-019-46074-2 PMID:31270387

Filippatos, G., Anker, S. D., Agarwal, R., Pitt, B., Ruilope, L. M., Rossing, P., Kolkhof, P., Schloemer, P., Tornus, I., Joseph, A., & Bakris, G. L.FIDELIO-DKD Investigators. (2021). Finerenone and cardiovascular outcomes in patients with chronic kidney disease and type 2 diabetes. *Circulation*, *143*(6), 540–552. doi:10.1161/CIRCULATIONAHA.120.051898 PMID:33198491

Gunasundari, S., Janakiraman, S., & Meenambal, S. (2018). Multiswarm heterogeneous binary PSO using win-win approach for improved feature selection in liver and kidney disease diagnosis. *Computerized Medical Imaging and Graphics*, 70, 135–154. doi:10.1016/j.compmedimag.2018.10.003 PMID:30366215

Guzzi, F., Cirillo, L., Roperto, R. M., Romagnani, P., & Lazzeri, E. (2019). Molecular mechanisms of the acute kidney injury to chronic kidney disease transition: An updated view. *International Journal of Molecular Sciences*, 20(19), 4941. doi:10.3390/ijms20194941 PMID:31590461

Han, X., Zhang, S., Chen, Z., Adhikari, B. K., Zhang, Y., Zhang, J., & Wang, Y. (2020). Cardiac biomarkers of heart failure in chronic kidney disease. *Clinica Chimica Acta*, *510*, 298–310. doi:10.1016/j.cca.2020.07.040 PMID:32710942

Henry, B. M., & Lippi, G. (2020). Chronic kidney disease is associated with severe coronavirus disease 2019 (COVID-19) infection. *International Urology and Nephrology*, *52*(6), 1193–1194. doi:10.1007/s11255-020-02451-9 PMID:32222883

Jankowski, J., Floege, J., Fliser, D., Boehm, M., & Marx, N. (2021). Cardiovascular disease in chronic kidney disease: Pathophysiological insights and therapeutic options. *Circulation*, 143(11), 1157–1172. doi:10.1161/ CIRCULATIONAHA.120.050686 PMID:33720773

Jerlin Rubini, L., & Perumal, E. (2020). Efficient classification of chronic kidney disease by using multi-kernel support vector machine and fruit fly optimization algorithm. *International Journal of Imaging Systems and Technology*, *30*(3), 660–673. doi:10.1002/ima.22406

Kumar, A., Sinha, N., Bhardwaj, A., & Goel, S. (2022). Clinical risk assessment of chronic kidney disease patients using genetic programming. *Computer Methods in Biomechanics and Biomedical Engineering*, 25(8), 887–895. doi:10.1080/10255842.2021.1985476 PMID:34726985

Kunwar, V., Sabitha, A. S., Choudhury, T., & Aggarwal, A. (2019). Chronic kidney disease using fuzzy C-means clustering analysis. *International Journal of Business Analytics*, 6(3), 43–64. doi:10.4018/IJBAN.2019070104

Ma, F., Sun, T., Liu, L., & Jing, H. (2020). Detection and diagnosis of chronic kidney disease using deep learning-based heterogeneous modified artificial neural network. *Future Generation Computer Systems*, 111, 17–26. doi:10.1016/j.future.2020.04.036

Mihai, S., Codrici, E., Popescu, I. D., Enciu, A. M., Albulescu, L., Necula, L. G., & Tanase, C. (2018). Inflammation-related mechanisms in chronic kidney disease prediction, progression, and outcome. *Journal of Immunology Research*, 2018, 2018. doi:10.1155/2018/2180373 PMID:30271792

Paik, J. M., Patorno, E., Zhuo, M., Bessette, L. G., York, C., Gautam, N., Kim, S. C., & Kim, S. C. (2022). Accuracy of identifying diagnosis of moderate to severe chronic kidney disease in administrative claims data. *Pharmacoepidemiology and Drug Safety*, *31*(4), 467–475. doi:10.1002/pds.5398 PMID:34908211

Portolés, J., Marques, M., López-Sánchez, P., de Valdenebro, M., Munez, E., Serrano, M. L., & Cuervas, V. (2020). Chronic kidney disease and acute kidney injury in the COVID-19 Spanish outbreak. *Nephrology, Dialysis, Transplantation*, *35*(8), 1353–1361. doi:10.1093/ndt/gfaa189 PMID:32871592

Sharma, S. (2018). A two stage hybrid ensemble classifier based diagnostic tool for chronic kidney disease diagnosis using optimally selected reduced feature set. *International Journal of Intelligent Systems and Applications in Engineering*, 6(2), 113–122. doi:10.18201/ijisae.2018642067

Siraj, M. (2019). A self-learning knowledge based system for diagnosis and treatment of chronic kidney disease. International Journal of Education and Management Engineering, 9(2), 44.

Wang, Y. N., Ma, S. X., Chen, Y. Y., Chen, L., Liu, B. L., Liu, Q. Q., & Zhao, Y. Y. (2019). Chronic kidney disease: Biomarker diagnosis to therapeutic targets. *Clinica Chimica Acta*, 499, 54–63. doi:10.1016/j.cca.2019.08.030 PMID:31476302

Zhuang, Y., Sun, J., & Liu, J. (2021). Diagnosis of chronic kidney disease by three-dimensional contrast-enhanced ultrasound combined with augmented reality medical technology. *Journal of Healthcare Engineering*, 2021, 2021. doi:10.1155/2021/5542822 PMID:33791081

*B.* Ramya Asalatha is a research scholar, Department of CSE, Andhra University Visakhapatnam. Her research interests include image processing, machine learning and deep learning.

James Stephen Meka is an Executive Council Andhra University Vishakapatanam, Principal WISTM Eng. College Vishakapatanam, Professor, Computer Science Engineering.