

**A COMPARATIVE STUDY ON CHRONIC KIDNEY CANCER
PREDICTION USING SUPERVISED MACHINE LEARNING
ALGORITHMS**

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering degree in Computer Science and Engineering

By

VEGI SAI SRI LAKSHMI GAYATHRI (Reg.No - 40111412)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

Category-1 University by UGC

Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI - 600119

APRIL - 2024



INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Vegi Sai Sri Lakshmi Gayathri** (Reg - 40111412) who carried out the Project work entitled "**A COMPARITIVE STUDY ON CHRONIC KIDNEY CANCER PREDICTION USING SUPERVISED MACHINE LEARNING ALGORITHMS**" under my supervision from November 2023 to April 2024.

Internal Guide

Dr. A. SIVASANGARI, M.E., Ph.D

Head of the Department

Dr. L. LAKSHMANAN

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Joppalar Nagar, Rajiv Gandhi Salai,
Chennai - 600 119.

Submitted for Viva-voce Examination held on 16-04-2024

Internal Examiner

External Examiner

DECLARATION

I, Vegi Sai Sri Lakshmi Gayathri (Reg.No - 40111412), hereby declare that the Project Report entitled "A COMPARATIVE STUDY ON CHRONIC KIDNEY CANCER PREDICTION USING SUPERVISED MACHINE LEARNING ALGORITHMS" done by me under the guidance of Dr. A. SIVASANGARI, M.E., Ph.D. is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 13-04-2024

PLACE: Chennai


SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **The Board of Management** of **Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr.T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr.L.Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.A.SIVASANGARI, M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Early detection and appropriate treatment can halt or postpone the progression of this chronic condition until the point at which kidney transplantation or dialysis are the only options left for saving the patient's life. Consequently, it is crucial to have reliable tools for the early detection of chronic renal disease. In this paper we will be using supervised machine learning models to predict chronic kidney disease. We integrated the influencing factors of chronic kidney cancer collected a simple dataset, conducted data analysis and optimized the dataset. Accurate detection of the stages of CKD (Chronic Kidney Disease) is believed to be vital to minimize impacts of patient's health complications such as hypertension, anemia (low blood count), mineral bone disorder, poor nutritional health, acid base abnormalities, and neurological complications with timely intervention through appropriate medications. The five supervised machine learning algorithms. Xgboost, gradient boosting, support vector machine, logistic regression and random forest classifier were used to develop a kidney cancer prediction model. The predictive performance of these five algorithms were compared and discussed. The comparison of the models will give the best fit for the prediction of chronic kidney disease. The comparison of the models is based on the different metrics like Accuracy, precision, recall, F1 score etc. The experimental predicting the kidney cancer.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1	INTRODUCTION	1
	1.1 Background and Motivation	2
	1.2 Problem Identification and Objectives	4
2	LITERATURE SURVEY	5
	2.1 Overview of Existing Research and Literature	5
	2.2 Open Problems in Existing System	8
3	REQUIREMENTS ANALYSIS	10
	3.1 Feasibility Studies of the Project	10
	3.2 Requirement Specification	11
4	DESCRIPTION OF PROPOSED SYSTEM	15
	4.1 Selected Methodology or Process Model	15
	4.2 Architecture of Proposed System	19
	4.3 System Scalability	19
	4.4 Project Plan	20
5	IMPLEMENTATION DETAILS	22
	5.1 Development and Deployment Setup	22
	5.2 Testing	23
6	RESULTS AND DISCUSSION	25
7	CONCLUSION	30
	REFERENCES	31
	APPENDIX	34

A. SOURCE CODE	34
B. SCREEN SHOTS	43
C. RESEARCH PAPER	46
D. CERTIFICATE	57

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
4.1.1	Architecture Diagram	19
5.1	Comparison of Accuracy Between Algorithms	26
5.2	Comparison of F1 Score Between Algorithms	27
5.3	Comparison of Precision Between Algorithms	27
5.4	Comparison of Recall Between Algorithms	28
5.5	Comparison of Error Rate Between Algorithms	28
B.1	Importing Required Libraries	43
B.2	Heat Map of Null Values	43
B.3	Shows The Abbreviated Feature Names	44
B.4	Data Visualization	44
B.5	Heat Map of The Data	45

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
CKD	Chronic Kidney Disease
SVM	Support Vector Machine
Xgboost	eXtreme Gradient Boosting
LR	Logistic Regression

CHAPTER 1

INTRODUCTION

Spine, the area below the abdomen, and the area below the ribs are all home to the two bean-shaped kidneys. Each kidney measures between 4 and 5 inches in length, or the size of a huge fist. Your body's blood is filtered by your kidneys. They keep the proper electrolyte levels, eliminate waste, and regulate the body's fluid balance. About 40 times daily, all the blood in your body travels through them. The kidney receives blood and filters waste while adjusting salt, water, and minerals as needed. The body receives the filtered blood back. Urine is produced from waste. A kidney runs the danger of losing all of its functions if the blood flow is obstructed. This could lead to renal failure.

Dialysis and kidney transplantation are the only ways to preserve the lives of kidney cancer patients, therefore early detection of the disease can help prevent it from progressing to later stages. The early detection of kidney cancer can be predicted using machine learning methods. In this study supervised machine learning algorithms will be used to categorize whether learning algorithms namely Xgboost logistic regression, support vector machine, gradient boosting, random forest classifier. These algorithms will give different accuracies. The accuracy of each model is compared using different classification metrics and the best accurate model will be used for predicting the output.

Accurate detection of the stages of CKD is believed to be vital to minimize impacts of patient's health complications such as hypertension, anemia (low blood count), mineral bone disorder, poor nutritional health, acid base abnormalities, and neurological complications with timely intervention through appropriate medications.

1.1 BACKGROUND AND MOTIVATION

CKD is a major global public health concern since it is linked to an elevated risk for severe clinical outcomes. The prognosis for individual individuals is still inadequately known, despite the fact that it is widely known that CKD is independently linked to elevated risks for end-stage renal disease, cardiovascular events, and all-cause mortality. Making treatment decisions requires the application of clinically feasible methods for risk stratification of each outcome.

Patients with chronic kidney disease may benefit from more intensive management, such as higher doses of RAAS (renin- angiotensin-aldosterone system) inhibitors, anticoagulation therapy, and intensive blood pressure, blood glucose, urate, and lipid-lowering medications. These individuals maybe identified at high risk using renal prognosis predictive models. Furthermore, there is significant clinical and financial significance in determining which outpatient CKD patients should have rapid and through examinations.

By using these models, the majority of patients who have a risk of proteinuria of less than 1 g/24 hours can be categorized as low risk and may receive treatment only from their primary outpatient follow-up, while the majority of patients who have a risk of proteinuria greater than 1 g/24hours can be referred to urgent care by an inpatient management registration.

By using supervised machine learning algorithms, researchers can train models on a large dataset of patient information, including factors like age, gender, medical history, and various biomarkers. These models can then learn patterns and relationships within the data to make predictions about the likelihood of a patient developing chronic kidney cancer

Research has been done to see if commonly obtained laboratory tests that do not show proteinuria may be used to predict the course of renal disease. In patients with stages 3-5 of chronic renal disease, models that include age, sex, estimated GFR, albuminuria, serum calcium, serum phosphate, serum bicarbonate, and serum albumin may reliably predict the progression to kidney failure.

Artificial intelligence techniques have demonstrated efficacy in solving real-world issues in more recent times, such as rule-based and gold standard-oriented diagnosis and prognoses. We developed and analyzed nine prediction models utilizing statistical, machine learning, and neural network techniques with blood-derived outpatient clinical characteristics and demographic data in order to assist clinicians in choosing prediction tools for predicting the severity of CKD. We also developed an online tool for patient follow-up urine protein severity prediction based on the findings.

1.1.1 STAGES OF CKD

Early on in its course, CKD usually shows no symptoms. This is because the human body can usually adapt to a significant decline in kidney function. Until a normal test for another condition, such as a blood or urine test, finds a possible problem, kidney disease is frequently not detected until this point.

Treatment with medicine and continued observation with regular testing may help stop it from developing into a more severe form if it is found early on. In its most advanced phases, CKD Kidney disease can present with several symptoms if it is not detected early or if it worsens despite therapy

1.2 PROBLEM IDENTIFICATION AND OBJECTIVES

The challenge is to identify the best machine learning model for kidney cancer prediction. There are several potential causes of kidney cancer. One of the main concerns is determining which variables have the most influence on the onset of kidney cancer. This may be achieved by using a feature selection strategy. After applying the machine learning models to the selected characteristics and contrasting them using the categorization metrics, the most accurate machine learning model will be found. The primary objective of this paper is to identify the best machine learning method for predicting the diagnosis of kidney cancer.

A computer program that computes and infers task-related data and derives the properties of the relevant pattern is referred to as machine learning. This technique may be a viable way to diagnosis chronic kidney disease (CKD) since it can make accurate and affordable diagnoses of illnesses. With the advancement of information technology, it has evolved into a new class of medical instrument and has a wide range of potential applications due to the quick growth of electronic health records.

Machine learning has previously been used to the medical industry to diagnose a variety of diseases, assess the pertinent disease-related aspects, and identify the human body state. As a summary of the earlier CKD diagnostic models, we discover that the majority of them have very low accuracy or a limited applicability range due to the way they impute missing information. As a result, we suggested a way in this study to increase the CKD diagnostic models' applicability range. Simultaneously, the model's precision is significantly enhanced.

CHAPTER 2

LITERATURE SURVEY

Not just in the fields of mathematics and engineering, but also in the field of medicine, machine learning algorithms have become an extremely important instrument. Predictions are made frequently using machine learning algorithms so that speedy decisions can be made. Algorithms are mostly used to enhance performance and accuracy.

2.1 OVERVIEW OF EXISTING RESEARCH AND LITERATURE

Vijayarani and Dhayanand [1] collected the kidney function test (KFT) dataset from medical labs, research centers, and hospitals. 584 occurrences, 6 attributes, are present in the dataset. SVM and ANN algorithms were used. With 87.7%, ANN's accuracy was the highest.

With the help of 9 machine learning algorithms Xiao et al. [2] utilized the data from 551 patients. The linear model provided the maximum accuracy, according to their analysis of accuracy, ROC curve, precision, and recall.

On the CKD Dataset, Reshma et al [3]. 's feature selection technique was applied. The ACO approach was used to pick features. The feature selection meta heuristic algorithm is called ACO. It belongs to the Wrapper technique family. There were a total of 24 attributes accessible in their dataset. Twelve features were utilized to create the model after the feature selection technique was used. Using the Support Vector Machine Classifiers technique the model was created.

The deep learning method developed by Ma et al. was proposed for the early prediction of chronic kidney disease. The deep neural network was built on top of a heterogeneous modified artificial neural network algorithm. In order to build the model, ultrasound images were used. The Support Vector Machine, artificial neural network, and multilayer perceptron were three different classifiers that were utilized to compare the outcomes.

Padmanaban and Parthiban [5] used Naive Bayes and Decision trees to analyze the dataset after collecting the data from a diabetic research center in Chennai. They used the Weka tool to determine accuracy and came to the conclusion that the Naive Bayes classifier had the highest accuracy, 91%. The decision tree algorithm was found to be the best performing supervised machine learning algorithm out of the four used by the researchers in this work.

The data of patients with chronic kidney illness are used in this study's implementation of the Xgboost, logistic regression, support vector machine, gradient boosting, and random forest classifier methods to predict kidney cancer. The preprocessed dataset is used to implement the algorithms. To select the most effective algorithm, performance comparisons are made using classification metrics including f1-score, accuracy, and precision.

Data mining is the process of using specialized software to find hidden information in a large set of data. Data mining techniques are linked to each other and used in a wide range of places and situations. With data mining technologies, we can make predictions, sort the data, filter it, and put it into groups. The goal of the algorithm is to process a training set that has a collection of attributes and targets, and the objective describes how this should be done.

If the dataset is very big, data mining is a good way to find patterns in it. If the dataset is very small, however, we can still reach the same goal with the help of machine learning. Data analysis and pattern recognition are two further

capabilities of machine learning. Because there is such a wide diversity of health datasets, machine learning algorithms are the most appropriate method for enhancing the accuracy of diagnosis prediction.

Using information mining techniques, a variety of types of studies have been carried out in order to extract useful information from datasets pertaining to chronic kidney disease. This was done in order to cut down on the amount of time spent conducting the analysis, and in addition to that, it would increase the precision of the forecast with the assistance of the information mining categorization technique. Data mining is also applied in the treatment and diagnosis of a number of diseases and conditions. Using techniques for information accumulation, different kinds of work have been done to get useful information out of the dataset on chronic kidney disease.

Polat et al.²¹ offered directions that combined a total of 6 classifiers and 3 outfit measures. k-nearest neighbors (KNN), naive Bayes (NB), support vector machine (SVM), preference tables, random forest (RF), and J48 were some of the classifiers that were used. The authors of Polat et al.²¹ looked into a number of possible treatments for chronic renal disease by using the k-means algorithm and Apriori. A test that uses SVM, DT, NB, and KNN computations was developed in order to diagnose chronic kidney disease (CKD). Ani et al.⁴ modified many characterizations of computations, including DT, NB, Linear discriminant analysis (LDA) classifiers, randomized subspace, and KNN, as well as back propagation network (BPN). The decision tree (DT) and NB characterization approaches were employed in order to anticipate CKD and reduce the mortality rate that was caused by CKD.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

- **Dataset Selection and Preprocessing** - Gathering a comprehensive dataset with diverse patient profiles and relevant features. Handling missing data, outliers, and ensuring data quality. Balancing the dataset to address class imbalance.
- **Feature Engineering and Selection** - Identifying the most informative features related to chronic kidney cancer. Exploring domain knowledge and medical literature to guide feature selection. Applying dimensionality reduction techniques to handle high-dimensional data.
- **Supervised Machine Learning Algorithms** - Comparing various algorithms such as decision trees, support vector machines, and random forests. Evaluating their performance metrics, including accuracy, precision, recall, and F1-score. Assessing the scalability and computational efficiency of the algorithms.
- **Model Interpretability and Explainability** - Exploring methods to interpret and explain the predictions made by the models. Utilizing techniques like feature importance, SHAP values, or LIME to provide insights to clinicians and patients. Ensuring transparency and trust in the prediction process.
- **Evaluation Metrics** - Exploring different evaluation metrics such as area under the ROC curve (AUC-ROC), precision-recall curve, and specificity. Understanding the strengths and limitations of each metric in the context of chronic kidney cancer prediction.
- **Ensemble Methods** - Investigating ensemble techniques like bagging, boosting, and stacking to improve prediction accuracy. Assessing the impact of combining multiple models and their potential benefits in handling complex data patterns.
- **Cross-Validation Strategies** - Implementing various cross-validation techniques such as k-fold cross-validation, stratified cross-validation and leave-one-out cross-validation. Ensuring robustness and generalizability of the

models by validating them on different subsets of the data.

- **Hyperparameter Tuning** - Experimenting with different hyperparameter settings for the machine learning algorithms. Utilizing techniques like grid search, random search, or Bayesian optimization to find the optimal hyperparameters.
- **Handling Imbalanced Data** - Investigating techniques like oversampling, under sampling, and SMOTE to address class imbalance in the dataset. Assessing the impact of different sampling strategies on model performance and generalizability.
- **Feature Importance Analysis** - Utilizing techniques like permutation importance, feature importance from tree-based models, or SHAP values to identify the most influential features in predicting chronic kidney cancer. Understanding the biological and clinical significance of these features.
- **Explainable AI Techniques** - Exploring advanced methods such as LIME (Local Interpretable Model-Agnostic Explanations) or SHAP (SHapley Additive explanations) to provide interpretable explanations for individual predictions. Enhancing the transparency and trustworthiness of the predictive models.
- **Transfer Learning** - Investigating the potential of leveraging pre-trained models on related tasks or datasets to improve prediction performance. Exploring transfer learning techniques like fine-tuning or feature extraction.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES OF THE PROJECT

A feasibility study is a comprehensive evaluation of a proposed project that evaluates all factors critical to its success in order to assess its likelihood of success. Business success can be defined primarily in terms of ROI, which is the amount of profits that will be generated by the project. In a feasibility study, a proposed plan or project is evaluated for its practicality. As part of a feasibility study, a project or venture is evaluated for its viability in order to determine whether it will be successful.

3.1.1 TECHNICAL FEASIBILITY

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system. As an exaggerated example, an organization wouldn't want to try to put Star Trek's transporters in their building—currently, this project is not technically feasible.

3.1.2 ECONOMIC FEASIBILITY

This assessment typically involves a cost/benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization.

3.1.3 Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether and how well the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

3.1.4 Scheduling Feasibility

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete. When these areas have all been examined, the feasibility analysis helps identify any constraints the proposed project may face, including:

1. Internal Project Constraints: Technical, Technology, Budget, Resource, etc.
2. Internal Corporate Constraints: Financial marketing, Export, etc.
3. External Constraints: Logistics, Environment, Laws, and Regulations, etc.

3.2 REQUIREMENTS SPECIFICATION

The Hardware and Software Requirements are described below :

3.2.1 Hardware Requirements

1. Processor: A modern processor with multiple cores, such as an Intel Core i5 or higher, to handle the computational workload efficiently.
2. Memory (RAM): Sufficient RAM, ideally 8GB or more, to accommodate the dataset and the memory requirements of the machine learning algorithms.
3. Storage: Adequate storage space to store the dataset, software tools, and any intermediate or final results generated during the study.
4. Graphics Processing Unit (GPU): While not always necessary, a dedicated GPU with CUDA support can significantly speed up the training process of

certain machine learning algorithms.

5. Internet Connectivity: A stable internet connection is required for accessing online resources, datasets, and any cloud-based machine learning platforms or libraries.

3.2.2 Software Requirements

1. Google Collab was used to create the models.
2. Python 3.7 was used to work with the dataset and apply the algorithms.
3. Machine Learning Libraries: Utilize machine learning libraries such as scikit-learn, Tensor Flow, or Keras to implement and evaluate different supervised learning algorithms.

3.2.3 Python

Python is a scripting language that is high-level, interpreted, interactive, and object-oriented. Python is intended to be extremely readable. It commonly employs English terms rather than punctuation, and it has fewer syntactical structures than other languages. The following are the features of python:

- Easy-to-learn - Python has a small number of keywords, a basic structure, and a well-defined syntax. This enables the pupil to swiftly learn the language.
- A broad standard library - The majority of Python's library is portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode - Python features an interactive mode that enables for interactive testing and debugging of code snippets.
- Portable - Python can operate on a broad range of hardware devices and has the same interface across them all.
- Extendable - The Python interpreter may be extended using low-level modules. These modules allow programmers to enhance or adapt their tools to make them more efficient.

- GUI Programming - Python can construct and port GUI programs to numerous system calls, libraries, and window systems, including Windows MFC, Macintosh, and Unix's X Window system.
- Scalable - Python provides a better structure and support for large programs than shell scripting.
- Easy-to-read - Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain - Python's source code is relatively simple to maintain.

3.2.4 Machine Learning

Machine learning is a branch of artificial intelligence that focuses on developing algorithms and models that allow computers to learn from data and make predictions or decisions without being explicitly programmed. It involves training a computer system to recognize patterns and make informed decisions based on examples or past experiences. In machine learning, a model is trained using a dataset that consists of input features and corresponding output labels or target values. The model learns from this data by identifying patterns and relationships, and then uses this knowledge to make predictions or decisions on new, unseen data.

There are different types of machine learning algorithms, including supervised learning, unsupervised learning, and reinforcement learning.

1. In supervised learning, the model is trained on labeled data, where the input features are paired with the correct output labels. The goal is for the model to learn the mapping between the input features and the corresponding labels, so it can accurately predict the labels for new, unseen data.
2. Unsupervised learning, on the other hand, involves training a model on unlabeled data, where the goal is to discover patterns or structures in the data without any predefined labels. This can be useful for tasks such as clustering or dimensionality reduction.

3. Reinforcement learning is a type of machine learning where an agent learns to interact with an environment and make decisions in order to maximize a reward signal. The agent learns through trial and error, receiving feedback in the form of rewards or penalties based on its actions.

Machine learning has a wide range of applications, such as image and speech recognition, natural language processing, recommendation systems, fraud detection, and much more. It's an exciting field that continues to evolve and has the potential to revolutionize various industries.

Data plays a crucial role in machine learning. It's like the fuel that powers the learning process. The more diverse and representative the data is, the better the model can learn and make accurate predictions. Data can come in various forms, like text, images, or numerical values.

Algorithms are the set of rules and procedures that the machine learning model follows to learn from the data. These algorithms help the model find patterns, make predictions, or make decisions based on the input it receives. There are various algorithms available, each suited for different types of problems and data.

Models are the result of the machine learning process. They are the trained representations of the patterns and relationships found in the data. Models can be as simple as a linear equation or as complex as a deep neural network with multiple layers. The goal is to create a model that can accurately generalize and make predictions on new, unseen data.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

The proposed system aims to develop a predictive model for chronic kidney cancer using various supervised machine learning algorithms. The system will collect clinical and demographic data from patients and utilize this information to train and evaluate different machine learning models. By comparing the performance of these algorithms, we can identify the most effective approach for predicting chronic kidney cancer. The ultimate goal is to improve the accuracy and efficiency of diagnosis and treatment for this disease. The system will involve collecting a dataset that includes clinical and demographic information of patients diagnosed with chronic kidney cancer. This dataset will be used to train and evaluate various supervised machine learning algorithms. These algorithms will learn patterns and relationships within the data to make accurate predictions about the presence or absence of chronic kidney cancer. The performance of each algorithm will be compared to determine which one yields the highest accuracy. This comparative study will help identify the most effective algorithm for predicting chronic kidney cancer, which can contribute to early detection and improved treatment outcome.

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

The project's whole work flow is provided by the system methodology, which aids in a better understanding of the project. This project's dataset was gathered from an online data source. The gathered data is then preprocessed to make it ready for any machine learning algorithm to use. The missing values are added to the dataset through additional processing. The features that are not very useful in determining the outcome can be eliminated using feature selection. The preprocessed dataset is used to implement the Xgboost, logistic regression, support vector machine, gradient boosting, and random forest classifier methods.

To determine the best fit algorithm, the classification metrics of each method are compared.

4.1.1 Dataset collection

The dataset was obtained from Kaggle and includes information on 830 patients with 26 distinct features, including age, sex, red blood cells, urea, sodium, potassium, and more. Both categorical and numerical data are present in the dataset. Using label encoding techniques, categorical features like "ckd" and "not ckd" are transformed into 0 and 1, respectively. Additionally, the dataset has some missing values that are later filled at random.

4.1.2 Data preprocessing

Data preprocessing is a data mining approach that transforms raw data into a format that is effective and valuable. The dataset we chose has a large number of initial missing values for each feature, which makes it difficult for us to develop models in later phases. For larger missing values, random sampling is used to fill them, while for smaller missing values, mean/mode sampling is used to fill them. Utilizing label encoding techniques, categorical data are transformed into numerical data once the missing values are filled in. Prior to the data being used by the models, the data is subjected to feature selection, during which some features are removed that don't improve the model's accuracy.

4.1.3 Algorithms

The machine learning techniques we employed for this research are supervised techniques. Algorithms used in supervised machine learning are those that are trained on well-labeled data. In this paper, we employed techniques to support vector machine, gradient boosting, Xgboost, logistic regression, and random forest classifier. Using the patient's information and the algorithms, we can determine whether there is a likelihood that kidney cancer will occur or not.

4.1.4 Logistic regression

A statistical method known as logistic regression is used in machine learning to solve issues involving binary categorization, such as predicting whether or not a consumer would purchase a product based on one or more input variables. Using a logistic function, the logistic regression model calculates the likelihood of the binary result, which has a range of 0 to 1. The projected probabilities are then categorized into one of the two classes by the model using a threshold value.

4.1.5 Support vector machine

The supervised machine learning algorithm Support Vector Machine (SVM) can be applied to both classification and regression applications. SVM seeks to identify a decision boundary that divides the data into distinct groups during classification, while during regression it seeks to identify a function that forecasts a continuous output variable.

4.1.6 Random forest classifier

A common ensemble learning approach for classification problems, where the objective is to predict the class of a data point based on its attributes, is the Random Forest Classifier. The accuracy and resilience of the model are increased by the random forest approach, which combines numerous decision trees.

4.1.7 Xgboost

XGBoost (eXtreme Gradient Boosting), is frequently used to tackle classification and regression issues. A strong classifier or regressor is produced by combining a number of weak decision trees using this effective ensemble learning technique. The gradient boosting framework serves as the foundation for XGBoost, which trains its models using this method. It functions by improving the prediction accuracy of the model by repeatedly training decision trees on the residuals of the prior trees.

4.1.8 Gradient boosting

A well-liked machine learning approach for classification and regression tasks is gradient boosting. It is an ensemble learning technique that combines several weak learners to produce a powerful learner (such as decision trees). The gradient boosting algorithm works by iteratively adding decision trees to the model, each of which is trained to predict the residuals (ie, the differences between the actual values and the predicted values) of the previous trees. In other words, each new tree is trained to correct the errors of the previous trees.

4.1.9 Classification metrics

Machine learning metrics are used to evaluate the model's performance. It is necessary to compare the models using various metrics after applying the machine learning models on the dataset. We must utilize the classification metrics in this case because we are performing categorization. Accuracy, recall, F1-score, false positive rate, false negative rate, and precision are a few categorization measures, the classification metrics' outcomes depend on TP, TN, FP, and FN.

- True positives (TP) are instances that have been definitively identified as having kidney cancer.
- False positives (FP) are cases that were mistakenly diagnosed as kidney cancer.
- The number of kidney cancer cases that were specifically identified as true negatives (TN).
- The precise number of cases correctly recognised as kidney cancer constitutes a false negative (FN).

4.2 ARCHITECTURE OF PROPOSED SYSTEM

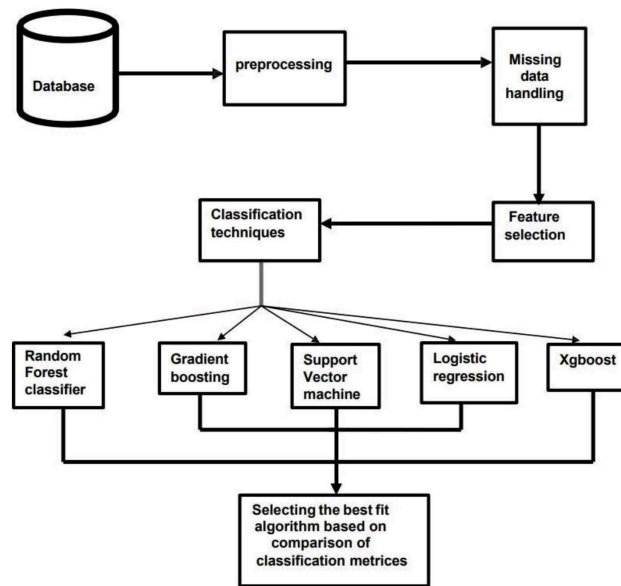


Fig. 4.2.1 Architecture Diagram

4.2 SYSTEM SCALABILITY

In this project, Google Collab was used to create the models, and Python was used to work with the dataset and apply the algorithms.

1. Data Preparation: Collect and preprocess the dataset, ensuring it is in a suitable format for machine learning algorithms.
2. Import Libraries: Import the necessary libraries such as scikit-learn, pandas, numpy, and matplotlib for data manipulation, model building, and evaluation.
3. Data Exploration: Perform exploratory data analysis to understand the dataset, visualize the features, and identify any missing values or outliers.
4. Feature Selection: Select relevant features from the dataset that are most informative for predicting chronic kidney cancer. You can use techniques like correlation analysis or feature importance ranking.

5. Model Selection: Choose the supervised machine learning algorithms you want to compare, such as logistic regression, decision trees, random forests, or svm.
6. Model Training and Evaluation: Split the dataset into training and testing sets. Train each model using the training set and evaluate their performance using appropriate evaluation metrics like accuracy, precision, recall, and F1- score.
7. Hyperparameter Tuning: Optimize the hyperparameters of each model using techniques like grid search or random search to improve their performance.
8. Model Comparison: Compare the performance of different models using the evaluation metrics and select the one that performs the best for chronic kidney cancer prediction. Implementation process, including the code, model performance results, and any insights gained during the project.
9. Final Model Testing: Once you have selected the best model, test it on the unseen testing data to assess its generalization ability.

4.4 PROJECT MANAGEMENT PLAN

Project management plan for the project on a comparative study of chronic kidney cancer prediction using supervised machine learning algorithms can be followed by the following steps:

1. Define Project Objectives: Clearly define the objectives of the project, such as evaluating the performance of different machine learning algorithms for chronic kidney cancer prediction.
2. Scope Definition: Determine the scope of your project, including the specific algorithms, datasets, and evaluation metrics you will use.
3. Identify Stakeholders: Identify the key stakeholders involved in the project, such as researchers, medical professionals, and potential users of the prediction model.
4. Develop a Project Schedule: Create a timeline with specific milestones and deadlines for each phase of the project, including data collection,

preprocessing, model training, evaluation, and reporting.

5. Resource Planning: Identify the resources required for your project, such as computing resources, software tools, and access to relevant datasets. Allocate resources accordingly to ensure smooth execution.
6. Risk Assessment and Mitigation: Identify potential risks and challenges that may arise during the project, such as data quality issues algorithm performance limitations. Develop strategies to mitigate these risks and have contingency plans in place.
7. Define the frequency and channels of communication, such as regular meetings or progress reports.
8. Quality Assurance: Define quality standards and procedures to ensure the accuracy and reliability of your project's outcomes. Perform regular checks and validations throughout the project lifecycle.
9. Documentation and Reporting: Document all project activities, including data preprocessing steps, algorithm implementations, evaluation results, and any insights gained. Prepare a final report summarizing the findings and conclusions.
10. Project Evaluation: Assess the success of project by comparing the achieved objectives with the initial project goals. Reflect on lessons learned and identify areas for improvement in future projects.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

1. Notebook Creation: Create a new notebook in Google Colab to work on your project.
2. Library Import: Import necessary libraries like 'xgboost', 'scikit-learn', and 'pandas' for working with the algorithms and handling data.
3. Dataset Loading: Load your dataset into the notebook using libraries like 'pandas' or by reading from a file.
4. Data Preprocessing: Perform preprocessing steps such as handling missing values, scaling features, or encoding categorical variables.
5. Train-Test Split: Split your data into training and testing sets using 'train_test_split' from 'scikit-learn'. This allows you to evaluate the performance of each algorithm on unseen data.
6. Algorithm Selection: Choose the algorithms you want to compare, such as XGBoosting, Gradient Boosting, SVM, Logistic Regression, and Random Forest Classifier.
7. Model Training: Create instances of each algorithm and train them on the training data using the respective fit methods.
8. Performance Evaluation: Evaluate the performance of each model on the testing data using appropriate evaluation metrics like accuracy, precision, recall, or F1- score. This will help you compare their predictive capabilities.
9. Hyperparameter Tuning: Experiment with different hyperparameter values for each algorithm to optimize their performance. Use techniques like grid search or random search to find the best hyperparameters.
10. Model Saving: Once you have selected the best-performing algorithm, save the trained model to a file using libraries like 'pickle' or 'joblib'.

11. Integration: Integrate the trained model into a web application or API if needed. You can use frameworks like Flask or Django to create a web interface for your model.
12. Documentation: Document your code and add comments to make it clear and understandable for others who might use or maintain your project.

5.2 TESTING

Once the models have been developed, the next step is to train them on the labelled dataset of physical activities. The development products of the given activity confirm the requirement of that activity. The dataset is divided into training and testing sets, with a larger portion of the dataset used for training the models. Once the models have been trained, they are tested on the testing set to evaluate their accuracy.

5.2.1 Data Collection

Verify Data Sources: Validate the reliability and accuracy of the chosen data sources by cross-checking with other trusted sources or conducting sample checks.

Data Integrity: Ensure that the collected data is complete, consistent, and free from errors or duplicates. This can be done through data profiling and data cleansing techniques.

5.2.2 Feature Engineering

Feature Validation: Validate the selected features by comparing them with domain knowledge or conducting statistical tests to ensure their relevance and usefulness in the model.

Transformation Validation: Validate the effectiveness of feature transformations by comparing the performance of models with and without the applied transformations.

5.2.3 Model Evaluation

Test-Train Split: Split the collected data into training and testing sets to evaluate the performance of the models on unseen data. The commonly used split ratios are 70:30 or 80:20.

Evaluation Metrics Validation: Validate the chosen evaluation metrics by comparing the results with domain experts' opinions or benchmark models in the same domain.

5.2.4 Model Interpretability

Feature Importance Validation: Validate the feature importance results by conducting sensitivity analysis or comparing them with experts' opinions.

Model Visualization Validation: Validate the interpretability of model visualizations by conducting user studies or obtaining feedback from domain experts.

5.2.5 Model Deployment

Testing APIs: Test the deployed model's APIs to ensure they are functioning correctly and providing accurate predictions.

Load Testing: Simulate high volumes of requests to test the scalability and performance of the deployed model, ensuring it can handle the expected load.

CHAPTER 6

RESULT AND DISCUSSIONS

In this study, the dataset was initially divided into numerical and categorical columns, and the null values in each column were removed. For removing null values in numerical and categorical columns, we used random value imputation and mode imputation, respectively, we have turned the non-numeric columns into numeric columns using scikit learn's label encoder class. The modified dataset is then divided into testing and training for the implementation of the five algorithms and is then compared using the classification metrics. The Random Forest algorithm achieved an accuracy of 96.3%, with precision, recall, F1-score, and error of 96.1%, 96.9%, 96.5%, and 3.6%, respectively. The logistic regression algorithm achieved an accuracy of 89.9%, with precision, recall, F1-score, and error of 95.6%, 84.6%, 89.7%, and 10.0%, respectively. The support vector machine algorithm achieved an accuracy of 59.4%, with precision, recall, F1-score, and error rates of 89.1%, 25.3%, 39.5%, and 40.0%, respectively. The gradient boosting algorithm achieved an accuracy of 93.5%, with precision, recall, F1-score, and error of 95.9%, 91.5%, 93.7% and 6.4%, respectively. The boost algorithm achieved an accuracy of 95.1%, with precision, recall, F1-score, and error rates of 96.8%, 93.8%, 95.3%, and 4.8%, respectively. According to the classification metrics random forest classifier algorithm is the best fit algorithm to predict the presence of chronic kidney cancer.

Algorithms	Random Forest	Support Vector Machine	Xgboosting	Logistic Regression	Gradient Boosting
Classification matrices					
Accuracy	96.3 %	59.4 %	95.1 %	89.9 %	93.5 %
F1 Score	96.5 %	39.5 %	95.3 %	89.7 %	93.7 %
Recall	96.9 %	25.3 %	93.8 %	84.6 %	91.5 %
Precision	96.1 %	89.1 %	96.8 %	95.6 %	95.9 %
Error rate	3.6 %	40.0 %	4.8 %	10.0%	6.4 %

Table 1: Comparison of performances of different algorithms based on classification metrices.

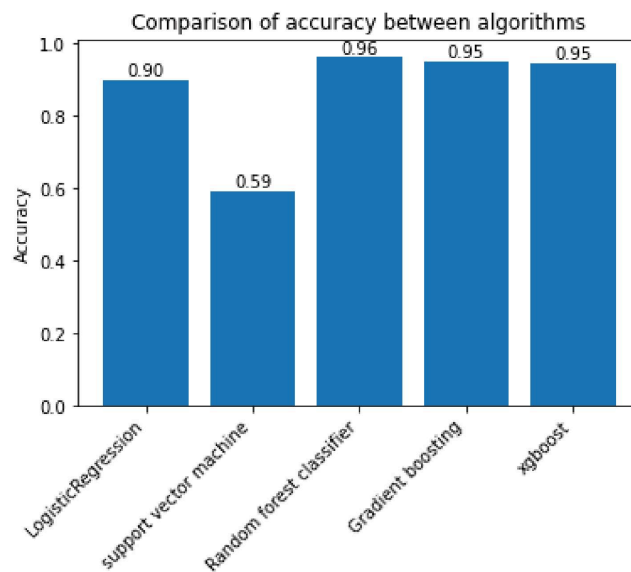


Fig 6.1 Comparison of accuracy between algorithms

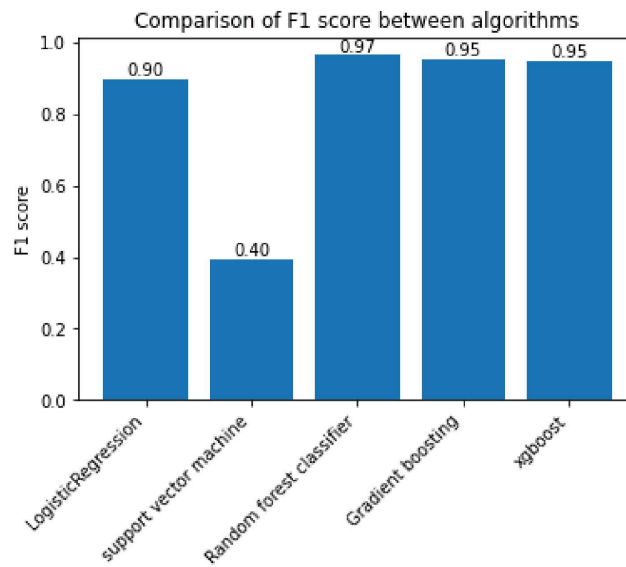


Fig 6.2 Comparison of F1 score between algorithms

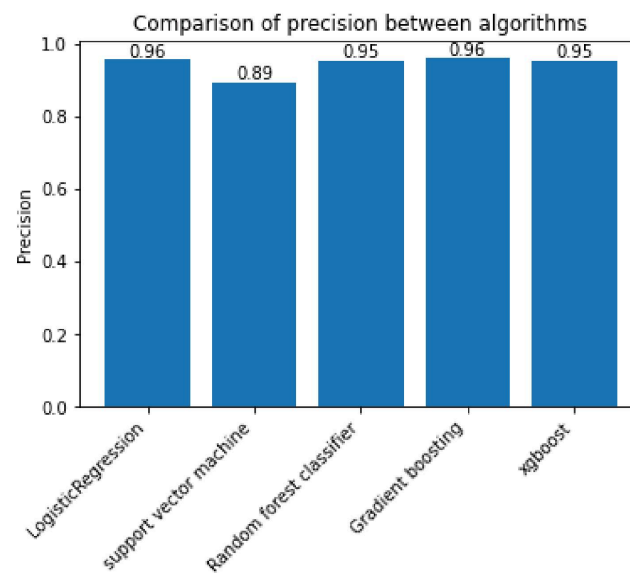


Fig 6.3 Comparison of precision between algorithms

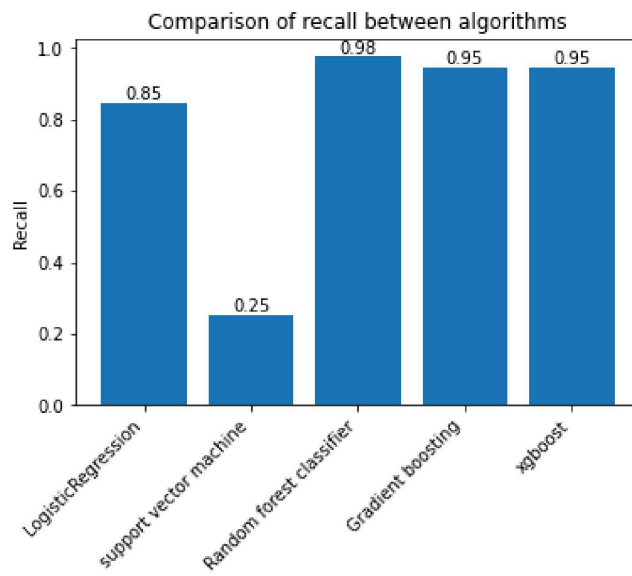


Fig 6.4 Comparison of recall between algorithms

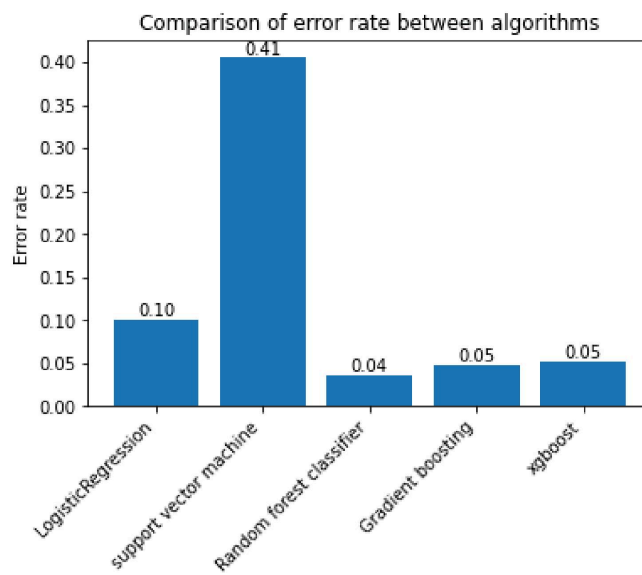


Fig 6.5 Comparison of error rate between algorithms

A. DISCUSSIONS

Chronic kidney disease is a global health threat and becoming silent killer in Ethiopia. Many people die or suffer severely by the disease mainly due to lack of awareness about the disease and inability to detect early. Thus, early prediction of chronic kidney disease is believed to be helpful to slow the progress of the disease. Machine learning plays a vital role in early disease identification and prediction. It supports the decision of medical experts by enabling them to diagnose the disease fast and accurately.

In this study, chronic kidney disease prediction has been carried out using machine learning techniques. The dataset was obtained from Kaggle and includes information on 830 patients with 26 distinct features, including age, sex, red blood cells, urea, sodium, potassium, and more. Both categorical and numerical data are present in the dataset. Using label encoding techniques, categorical features like "ckd" and "not ckd" are transformed into 0 and 1, respectively. Additionally, the dataset has some missing values that are later filled at random.

Data preprocessing is a data mining approach that transforms raw data into a format that is effective and valuable. The dataset we chose has a large number of initial missing values for each feature, which makes it difficult for us to develop models in later phases. For larger missing values, random sampling is used to fill them, while for smaller missing values, mean/mode sampling is used to fill them. Utilizing label encoding techniques, categorical data are transformed into numerical data once the missing values are filled in. Prior to the data being used by the models, the data is subjected to feature selection, during which some features are removed that don't improve the model's accuracy.

CHAPTER 6

CONCLUSION

Random forest classifier, out of the five algorithms, gives the best performance. The Support Vector machine is the least. The random forest and other supervised machine learning algorithms can be compared similarly in further research. If the dataset is larger or when better data is available study can also be done. This research can be utilized to create a website where people can provide information from test results and estimate their risk of developing kidney cancer.

There are additional conceivable evolutionary strategies that may be utilized to enhance the outcomes of the suggested classifiers. To identify CKD, SVM and CNN are used in this study. The performance of the used classifiers may also be assessed and contrasted with that of other classifiers already in use. Early diagnosis of CKD aids in both the prompt treatment of affected patients and the prevention of the disease's progression. The medical field requires prompt diagnosis and treatment of illnesses. In subsequent work, new classifiers can be employed and their effectiveness assessed in order to identify more optimal solutions for the goal function.

A. FUTURE WORK

This study used a supervised machine-learning algorithm, feature selection methods to select the best subset features to develop the models. It is better to see the difference in performance results using unsupervised or deep learning algorithms models. The proposed model supports the experts to give the fast decision, it is better to make it a mobile-based system that enables the experts to follow the status of the patients and help the patients to use the system to know their status.

REFERENCES

- [1] AbdElhafeez S, Bolignano D, D'Arrigo G, Dounousi E, Tripepi G, Zoccali C. Prevalence and burden of chronic kidney disease among the general population and high-risk groups in Africa: A systematic review. *BMJOpen*.2018;8:1.
- [2] B. Deepika. "Early prediction of chronic kidney disease by using machine learning techniques", *Amer. J. Comput. Sci. Eng. Survey*, vol. 8,no. 2, pp. 7, 2020,
- [3] Eroğlu K., Palabaş T. 2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO) IEEE; 2016. The impact on the classification 24 performance of the combined use of different classification methods and different ensemble algorithms in chronic kidney disease detection; pp. 512–516.
- [4] Fatima M., Pasha M. Survey of machine learning algorithms for disease diagnostic. *J Intel Learn Syst Appl*. 2017;9(01):1.
- [5] George C, Mogueo A, Okpechi I, Echouffo-Tcheugui JB, Kengne AP.Chronic kidney disease in low-income to middle-income countries: The case f increased screening. *BMJ Glob Heal*. 2017;2(2):1–10.
- [6] Gudeti B., Mishra S., Malik S., Fernandez T.F., Tyagi A.K., Kumari S. 2020 4th International Conference on Electronics,Communication and Aerospace Technology (ICECA) IEEE; 2020. A novel approach to predict chronic kidney disease using machine learning algorithms; pp. 1630–1635.
- [7] Heung M., Chawla L.S. Predicting progression to chronic kidney disease after recovery from acute kidney injury. *Curr Opin Nephrol Hypertens*. 2012;21:628–634.
- [8] Islam M., Shampa M., Alim T., et al. Convolutional neural networkbased marine cetaceans detection around the swatch of no ground in thebay of 31engal. *Int J Comput Digit Syst*. 2021;12:877–893.
- [9] Islam M.A., Akter S., Hossen M.S., Keya S.A., Tisha S.A., Hossain S. 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)

- IEEE; 2020. Risk factor prediction of chronic kidney disease based on machine learning algorithms; pp. 952– 957.
- [10] Islam M.A., Hasan M.R., Begum A. Improvement of the handover performance and channel allocation scheme using fuzzy logic, artificial neural network and neuro fuzzy system to reduce call drop in cellular
- [11] Islam, M.A., Akhter, T., Begum, A. Hasan, M.R., Rafi, F.S. Brain tumor detection from MRI images using image processing.
- [12] J. M. Pereira, M. Basto and A. F. D. Silva, “The logistic lasso and ridge regression in predicting corporate failure”, *Procedia Econ, Finance*, vol. 39, pp. 634-641, Jan. 2016.
- [13] J. Xiao, R. Ding, X. Xu, H. Guan, X. Feng, T., Sun, et al., “Comparison and development of machine learning tools in the prediction of chronic kidney disease progression”, *J. Transl. Med.*, vol. 17, pp. 119, Dec. 2019.
- [14] K. R. A. Padmanaban and G. Parthiban, “Applying machine learning techniques for predicting the risk of chronic kidney disease”. *Indian J. Sci. Technol.*, vol. 9, no.29, Aug. 2016.
- [15] Mahesh B. Machine learning algorithms-a review. *Int J Sci Res (IJSR)*. [Internet] 2020;9:381–386.
- [16] Mayan J.A, Arifa S, Pavithra R(2016) ,”Semantic based multilexical ranking technique for an effective search in protected cloud,2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICT), pp. 570-576.
- [17] Molla MD, et al. Assessment of serum electrolytes and kidney function test for screening of chronic kidney disease among Ethiopian Public Health Institute staff members, Addis Ababa, Ethiopia. *BMC Nephrol*.2020;21(1):494.
- [18] Polat H., Danaei Mehr H., Cetin A. Diagnosis of chronic kidney disease based on support vector machine by feature selection methods. *J Med Syst*. 2017;41:1–11.
- [19] Qin J., Chen L., Liu Y., Liu C., Feng C., Chen B. A machine learning methodology for diagnosing chronic kidney disease. *IEEE Access*.

2019;8:20991– 21002.

- [20] R.Ruiz-Arenas, “A summary of worldwide national activities in chronic kidney disease (CKD) testing, the electronic journal of the international federation of,” Clinical Chemistry and Laboratory Medicine, vol.28,no.4,pp.302–314,2017.
- [21] Rácz A., Bajusz D., Héberger K. Multi-level comparison of machine learning classifiers and their performance metrics. *Molecules*. 2019;24:2811.
- [22] Ravizza S., Huschto T., Adamov A., et al. Predicting the earlyrisk of chronic kidney disease in patients with diabetes using real-world data. *Nat Med*. 2019;25:57– 59
- [23] S. Vijayarani and S. Dayananda, “Kidney disease prediction usingSVM and ANN algorithms, *Int. J. Comput. Bus. Res.*, vol. 6, no. 2, pp. 1- 12, Mar. 2015.
- [24] Stanifer JW, et al. The epidemiology of chronic kidney disease in sub-Saharan Africa: A systematic review and meta-analysis. *Lancet Glob Heal*.2014;2(3):e174–81. Network. *J Eng Adv*. 2020;1:130–138.

APPENDIX

A.SOURCE CODE

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
from sklearn.metrics
import confusion_matrix, accuracy_score
# from google.colab import files
# uploaded = files.upload()
# import io
# df = pd.read_csv(io.BytesIO(uploaded['Kidney_disease.csv']))
df = pd.read_csv('kidney830 - Sheet1.csv')
df.head()
df.describe()
data = df
# data cleaning
for i in range(data.shape[0]):
    if data.iloc[i, 25] == 'ckd\t':
        data.iloc[i, 25] = 'ckd'
    if data.iloc[i, 20] in [' yes', '\t yes']:
        data.iloc[i, 20] = 'yes'
    if data.iloc[i, 20] == '\t no':
```

```

data.iloc[i,20]='no'
if data.iloc[i,21]=='\tno':
data.iloc[i,21]='no'
if data.iloc[i,16]=='\t?':
data.iloc[i,16]=np.nan
data.iloc[i,16]=='\t43':
data.iloc[i,16]='43'
if data.iloc[i,17]=='\t?':
data.iloc[i,17]=np.nan
if data.iloc[i,17]=='\t6200':
data.iloc[i,17]='6200'
if data.iloc[i,17]=='\t8400':
data.iloc[i,17]='6200'
if data.iloc[i,18]=='\t?':
data.iloc[i,18]=np.nan
if data.iloc[i,25]=='ckd':
data.iloc[i,25]='yes'
if data.iloc[i,25]=='notckd':
data.iloc[i,25]='no'
data.head()
feature_names=['id','Age (yrs)','Blood Pressure (mm/Hg)','Specific
Gravity','Albumin','Sugar','Red Blood Cells','Pus Cells','Pus Cell Clumps',
'Bacteria','Blood GlucoseRandom(mgs/dL)','Blood Urea (mgs/dL)',
'Serum Creatinine (mgs/dL)','Sodium (mEq/L)','Potassium(mEq/L)',
'Hemoglobin (gms)','Packed Cell Volume',
'White Blood Cells (cells/cmm)','Red Blood Cells(millions/cmm)',
'Hypertension','Diabetes Mellitus',
'Coronary Artery Disease',
'Appetite','Pedal Edema','Anemia','Chronic Kidney Disease']
data.columns=feature_names
data.head()

```

```

mistyped=['Packed Cell Volume',
'White Blood Cells(cells/cmm)','RedBlood Cells (millions/cmm)'] numeric=[]
for i in data.columns:
if data[i].dtype=='float64':

numeric.append(i) numeric=numeric+mistyped
for col in mistyped:
data[col]=data[col].astype('float')
data.info()
##extracting numaric andcategorical data
num_cols = [col for col in data.columns

if data[col].dtype !='object']
cat_cols= [col for col in data.columns
if data[col].dtype == 'object']
num_cols
for col in cat_cols:
print(f"{col} has {df[col].unique()} values\n")
print(len(num_cols))
print(len(cat_cols))
df.isna().sum()
sns.heatmap(df.isnull(),yticklabels=False,cbar=False,cmap='viridis')

# # filling null values, we will use two methods, random sampling for higher null
values

# # mean/mode sampling for lower null values
# def random_value_imputation(feature):
# random_sample = df[feature].dropna().sample(df[feature].isna().sum())
# random_sample.index = df[df[feature].isnull()].index
# df.loc[df[feature].isnull(),feature] =random_sample

# def impute_mode(feature):
# mode = df[feature].mode()[0]
# df[feature] =df[feature].fillna(mode)

```

```

def random_value_imputation(feature):

random_sample =df[feature].dropna().sample(df[feature].isna().sum())

random_sample.index = df[df[feature].isnull()].index
df.loc[df[feature].isnull(),feature] =random_sample

for col in num_cols:

random_value_imputation(col)

df[num_cols].isnull().sum()

def impute_mode(feature):

mode = df[feature].mode()[0]

df[feature]=df[feature].fillna(mode)

for col in cat_cols:

impute_mode(col)

df[cat_cols].isnull().sum()

sns.heatmap(df.isnull(),yticklabels=False,cbar=False,cmap='viridis')
sns.countplot(x='Chronic Kidney Disease',data=data)
plt.xlabel("yes")
plt.ylabel("no")
plt.title("Chronic Kidney disease")
plt.figure(figsize=(20,20))
# for i,feature in enumerate(cat_cols):
plt.subplot(4,3,i+1)
# sns.countplot(data[feature])

plt.figure(figsize=(30,20))
for i,feature in enumerate(num_cols):

```

```

plt.subplot(5,3,i+1)

data[feature].hist()

plt.title(feature)

plt.figure(figsize=(10,8))corr

df = data.corr()

sns.heatmap(corr_df,annot=True)

for col in cat_cols:
print('{} categories in the column- {}'.format(data[col].nunique(),col))
from sklearn.preprocessing
import LabelEncoderle=LabelEncoder()
for col in cat_cols:
data[col]=le.fit_transform(data[col])
data.head()
df.drop('id', inplace=True, axis=1)
df.head()
# from sklearn.feature_selection

import SelectKBest

# fromsklearn.feature_selection

import chi2
# ind_col=[col for col in df.columns if col!='Chronic Kidney Disease']
#dep_col='Chronic Kidney Disease'
# x=df[ind_col] # y=df[dep_col]


# ordered_rank_features=SelectKBest(score_func=chi2,k=20)
# ordered_feature=ordered_rank_features.fit(x,y)
# datascores=pd.DataFrame(ordered_feature.scores_,columns=["Score"])

```



```

# dfcolumns=pd.DataFrame(x.columns)

# features_rank=pd.concat([dfcolumns,datascores],axis=1)

# features_rank.columns=['Features','Score']

# selected_columns=features_rank.nlargest(10,'Score')['Features'].values

#X_new=df[selected_columns]

# X_new
X = df.drop(columns ='Chronic Kidney Disease' , axis=1)
Y = df['ChronicKidney Disease']
from sklearn.linear_model

import Lasso
from sklearn.feature_selection

import SelectFromModel
feature_model = SelectFromModel(Lasso(alpha=0.007))
feature_model.fit(X,Y)
cols=X.columns
selected_feature = cols[feature_model.get_support()]

print(len(selected_feature))

print(selected_feature)

cols=X.columns

selected_feature = cols[feature_model.get_support()]

print(len(selected_feature))

print(selected_feature)

# X = df.drop(columns ='Chronic Kidney Disease' , axis=1)

# Y =df['Chronic Kidney Disease']

# from sklearn.linear_model import Lasso
# from sklearn.feature_selection

```

```

import SelectFromModel

# feature_model = SelectFromModel(Lasso(alpha=0.005))
#feature_model.fit(X,Y)
# cols= X.columns
# selected_feature = cols[feature_model.get_support()]

print(selected_feature)
#from sklearn.model_selection

import train_test_split
X_train, X_test, y_train,

y_test = train_test_split(X, Y, train_size=0.7,random_state=123)
print('X_train :',X_train.shape)

print('X_test:' , X_test.shape)

print('y_train:',y_train.shape)
print('y_test :' , y_test.shape)
#from sklearn.linear_model

import LogisticRegression from sklearn.svm

import SVC

from sklearn.ensemble

import RandomForestClassifierfrom xgboost

import XGBClassifier
import xgboost
from sklearn.ensemble

import GradientBoosting

Classifiermodels=[ LogisticRegression(solver='liblinear'),SVC(),

```

```

RandomForestClassifier(), xgboost.XGBRFClassifier(max_depth=3,
random_state=1),GradientBoostingClassifier(max_depth=2, random_state=1)
]
models=[] models.append(('LogisticRegression',LogisticRegression(solver='liblinear')
))

models.append(('support vector machine',SVC()))

models.append(('Random forest classifier',RandomForestClassifier()))
models.append(('Gradient boosting',GradientBoostingClassifier(max_depth=2,
random_state=1)))
models.append(('xgboost',xgboost.XGBRFClassifier(max_depth=3,random_state=1)))
for name,model in models:
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

print(name)ac = accuracy_score(y_test, y_pred)
print('Accuracy score is ',ac)
print('='*50)

n = len(X)
n_train = int(n * 0.7)
n_test = n - n_train

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, train_size=0.7,
random_state=123)

for name,model in models:
y_pred = model.predict(X_test)

accuracy = sum(y_test == y_pred) / n_test

```

```

true_positives = sum((y_test == 1) & (y_pred == 1))
true_negatives = sum((y_test== 0) & (y_pred == 0))

false_positives = sum((y_test == 0) & (y_pred == 1))
false_negatives = sum((y_test == 1) & (y_pred ==0))

precision = true_positives / (true_positives + false_positives)

recall =true_positives / (true_positives + false_negatives)
f1 = 2 * (precision * recall) / (precision + recall)
error rate= 1-accuracy

print(name)

print("Accuracy:", accuracy*100)
print("Precision:", precision*100)

print("Recall:", recall*100)

print("F1 score:", f1*100)

print("Error rate:",errorrate*100)

print('='*50)

```

B.SCREENSHOTS

```
[ ] import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
from sklearn.metrics import confusion_matrix, accuracy_score
```

```
▶ # from google.colab import files
# uploaded = files.upload()
```

```
[ ] # import io
# df = pd.read_csv(io.BytesIO(uploaded['Kidney_disease.csv']))
```

Fig B.1 Importing Required Libraries

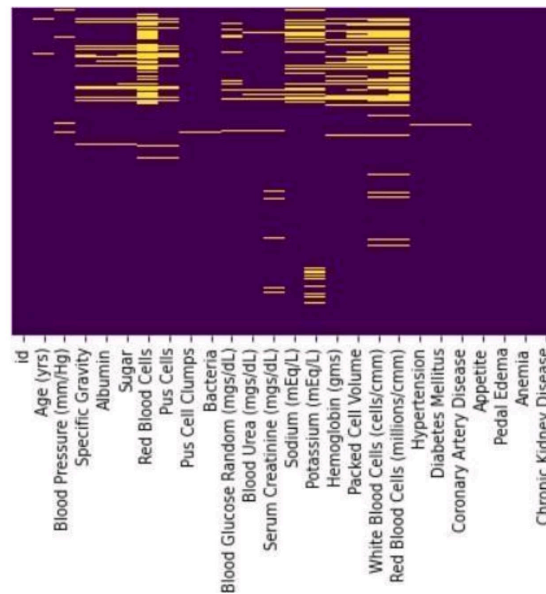


Fig B.2 Heat map of null values

```
feature_names=['id','Age (yrs)','Blood Pressure (mm/Hg)','Specific Gravity','Albumin','Sugar','Red Blood Cells',
               'Pus Cells','Pus Cell Clumps','Bacteria','Blood Glucose Random (mgs/dL)','Blood Urea (mgs/dL)',
               'Serum Creatinine (mgs/dL)','Sodium (mEq/L)','Potassium (mEq/L)','Hemoglobin (gms)','Packed Cell Volume',
               'White Blood Cells (cells/cmm)','Red Blood Cells (millions/cmm)','Hypertension','Diabetes Mellitus',
               'Coronary Artery Disease','Appetite','Pedal Edema','Anemia','Chronic Kidney Disease']
data.columns=feature_names
```

Fig B.3 Shows the abbreviated feature names

```
[ ] sns.countplot(x='Chronic Kidney Disease',data=data)
plt.xlabel("yes")
plt.ylabel("no")
plt.title("Chronic Kidney disease")
```

Text(0.5, 1.0, 'Chronic Kidney disease')

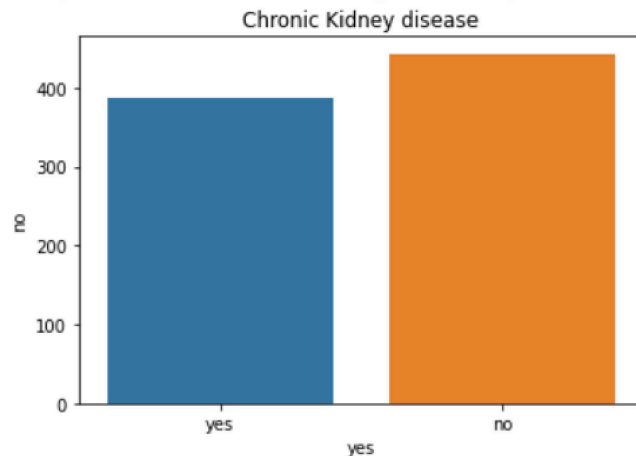


Fig B.4 Data Visualization

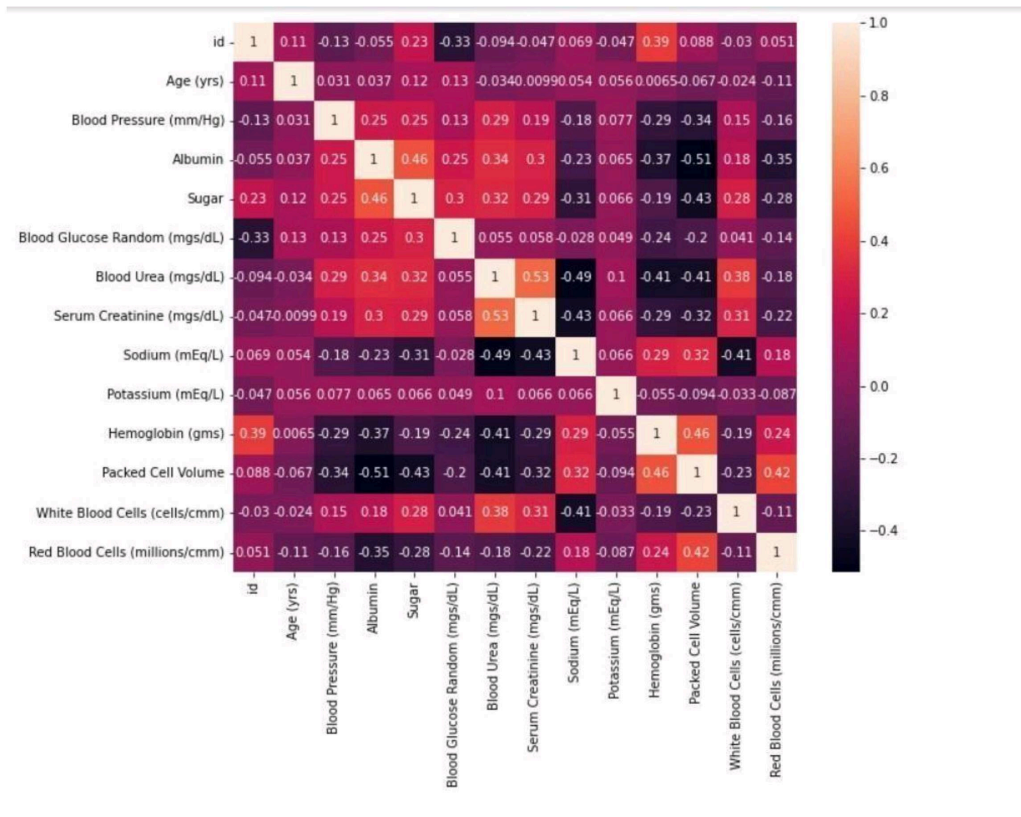


Fig B.5 Heat map of the data

C.RESEARCH PAPER

A Comparative Study On Chronic Kidney Cancer Prediction Using Supervised Machine Learning Algorithms

A.Sivasangari ^{1,a)}, Vegi Sai Sri Lakshmi Gayathri ^{2,b)}

¹ Professor, ² Student

^{1,2} School of Computing , Sathyabama Institute of Science and Technology, Tamilnadu, Chennai, India

^{a)}sivasangarikavya@gmail.com ^{b)}gayathrivegi875@gmail.com

Abstract. With early diagnosis along with suitable treatment, the chronic disease can be prevented from progressing further, delaying the point at which dialysis or kidney transplantation become the patient's sole options for survival. Therefore, having reliable tools is crucial for the early detection of chronic kidney disease. In this work, we will employ supervised machine learning models to predict chronic kidney disease. We collected a sample dataset, performed data analysis, and dataset refinement after combining the factors influencing chronic kidney cancer. A kidney cancer prediction model was created using five supervised machine learning algorithms: random forest classifier, logistic regression, support vector machine, gradient boosting, and Xgboost. The forecasting performance of these five algorithms was compared and assessed. A wide range of metrics, including F1 score, recall, accuracy, and precision, are employed to compare the models. The random forest algorithm is the most accurate strategy for predicting kidney cancer, owing to the experiment's statistics.

INTRODUCTION

Renal cell carcinoma is a term used to describe chronic kidney cancer, which is a tumorous growth originating from the cells lining the small tubes within the kidney. A diagnosis is difficult since the illness often shows no symptoms in its early stages. Consequently, reliable and accurate prediction models that can detect the illness at an early stage are needed.

Artificial neural networks, decision trees, random forests, and support vector machines are a few supervised machine learning methods that have developed into useful instruments for chronic kidney cancer prediction in recent times. These algorithms employ a range of statistical and mathematical method to examine huge datasets and detect patterns that can be utilized to identify individuals who are susceptible to the illness.

The use of supervised machine learning algorithms provides significant advantages over traditional statistical approaches in the prediction of chronic kidney cancer. These algorithms can, first, analyze enormous volumes of data and identify patterns that may not be visible to human observers. Second, they can process an extensive selection of input data types, such as binary, continuous, and categorical variables. Finally, the precise real-time predictions generated by these systems might find application in therapeutic environments.

This article aims to assess several supervised machine learning algorithms' predicting potential for chronic kidney cancer. A sizable collection of test results, medical histories, and patient data will be used in the study. Several supervised machine learning techniques will be used to the

training data to produce prediction models. One training set and one testing set will be included in the dataset. The testing set will be used to evaluate each algorithm's performance using a widerange of metrics, such as accuracy, precision, recall, and F1 score.

Finally, the goal of this research study is to provide light on how supervised machine learning algorithms may be used to anticipate chronic kidney cancer. The creation of an automated prediction system for chronic kidney cancer may benefit from the study's findings, which might result in an earlier diagnosis and better patient outcomes.

The National Kidney Foundation splits the five stages of chronic kidney disease (CKD) based on abnormal kidney function and reduced Glomerular Filtration Rate (GFR), a measure of kidney function. While stages 5 are believed to be end-stage renal failure, stages 1 and 2 are the least severe and show relatively few symptoms.

Tele Treatment is unavailable in most nations that are struggling, including Ethiopia. Because of this, renal failure and the repercussions are very difficult to cure in developing countries because of a shortage of resources, including expensive healthcare and facilities and physicians.

Machine learning is the process by which a computer software computes and infers task-related data and determines the distinctive features of the pertinent pattern. Given its ability to provide precise and affordably priced diagnostic services, this strategy could be a useful tool for the diagnosis of chronic kidney disease (CKD). The rapid expansion of electronic health records has led to the evolution of information technology into a new type of medical equipment with a widevariety of potential uses. In the past, machine learning has been applied in the field of medicineto diagnose a range of illnesses, evaluate relevant disease-related factors, and establish the status of the human body.

As a summary of the earlier CKD diagnostic models, we discover that many of them have very low accuracy or a limited applicability range due to the way they impute missing information. As a result, we suggested a way in this study to increase the CKD diagnostic models' applicability range. Simultaneously, the model's precision is significantly enhanced.

RELATED WORKS

Machine learning algorithms have become a vital tool not only in the domains of engineering and mathematics but also in medicine. Machine learning algorithms are often used to make predictions, enabling quick decision-making. The main purposes of algorithms are accuracy and performance improvement.

Of the four supervised machine learning algorithms that the researchers employed in this experiment, the decision tree approach proved to be the most effective.[1].

Hospitals, research centers, and medical labs provided the kidney function test (KFT) dataset that Vijayarani and Dhayanand [2] collected. Six characteristics and 584 occurrences make up the dataset. SVM and ANN algorithms were used. With an accuracy of 87.7%, ANN had the highest accuracy.

Nine machine learning methods and the data from 551 patients were employed by Xiao et al. [3]. The linear model provided the best degree of accuracy, according to their examination of accuracy, ROC curve, precision, and recall.

Charleonnann et al. [4] used data from Indians with Chronic Kidney Disease (CKD) to compare several predictive models, including K-nearest neighbors (KNN), support vector machine (SVM), logistic regression (LR), and decision tree (DT), in order to determine the best classifier for predicting chronic kidney disease. 98.3% classification accuracy and 0.99 sensitivity are demonstrated to be the highest offered by SVM.

Salekin and Stankovic [5] examined 400 datasets to assess classifiers including K-NN, RF, and ANN. For creating the model and selecting wrapper characteristics, five features were selected for the study. RF. S. records the best classification accuracy of 98% with an RMSE of 0.11. The "Prediction of Chronic Kidney Disease Using Machine Learning Algorithm" was developed by Tekale et al. [6] using a dataset of 400 occurrences and 14 features. Support vector machines and decision trees have been used by them. There are now merely 14 characteristics in the dataset as opposed to 25 after preprocessing. With an accuracy rating of 96.75%, SVM is acknowledged as a superior model. A self-learning knowledge-based system for the detection and treatment of the initial three stages of chronic renal illness was studied by Mohammed and Beshah [7] using machine learning. Using a small collection of data for this investigation, a prototype was created that enables the patient to query KBS to watch the advice execution. They used decision trees in order to construct the regulations. Reports indicate that 91% of the prototype's overall accuracy.

Rady and Anwar [8] evaluated probabilistic neural networks (PNN), multilayer perceptron (MLP), support vector machines (SVM), and radial basis functions (RBF) technologies to predict the stages of renal disease. For their experiment, the researchers worked with a small dataset that had few features. Based on the total classification accuracy percentage of 96.7%, the study's findings show that the Probabilistic Neural Networks technique yields the greatest results.

METHODOLOGY

To help with a better knowledge of the project, the system methodology provides the project's whole workflow. A web-based data source provided the dataset for this research. The collected data is subsequently prepared for usage by a machine learning algorithm through preprocessing. Through further processing, the missing values are added to the dataset. Using feature selection, features that are not particularly helpful in deciding the outcome can be removed.

The methods of logistic regression, random forest classifier, support vector machine, gradient boosting, and Xgboost are all implemented using the preprocessed dataset. The classification metrics of each approach are compared to identify the best- suited algorithm.

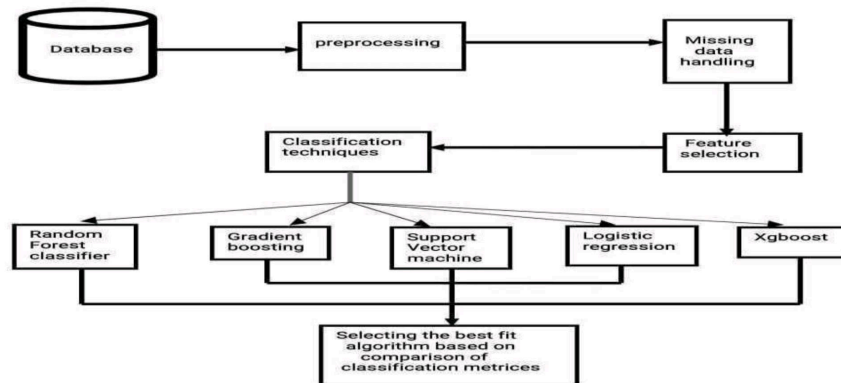


FIG 1: The methods used to determine the optimal algorithm for kidney cancer detection is depicted in the accompanying figure.

PROPOSED SYSTEM

Dataset collection

An online database from which we retrieved the data includes the medical records of 830 individuals, each of whom had 26 attributes. Python was utilized as the programming language and Google Collab was utilized as the model- building tool for this study.

Data Preprocessing

Raw data must be cleaned, transformed, and made ready for analysis, which is why data preparation is crucial. In order to guarantee precise and dependable findings, this stage involves eliminating mistakes, adding missing numbers, managing outliers, and managing variables. Preprocessing data helps researchers get more relevant insights from their data and enhances the quality of their data as well as the accuracy of their analysis.

A small number of null values are present in the dataset we gathered. We eliminated the null values using the subsequent techniques.

Random value imputation

A technique for dealing with missing data is random value imputation, which involves substituting randomly generated values from the distribution of the variable for the missing values. The randomly generated values can be chosen from a uniform or normal distribution, and this approach can be applied to numerical or continuous variables.

Mode imputation

One method for dealing with missing data is mode imputation, which involves substituting the missing values with the variable's most often occurring value. When a variable has a discrete numerical range or is categorical or nominal in nature, this strategy works well.

To exclude the null values from the dataset, we employed mode imputation for categorical columns and random value imputation for numerical columns.

Since many machine learning techniques need the conversion of category data into numerical variables, label encoding is crucial. This procedure gives each variable category a distinct numerical value, making it possible for the algorithm to identify and handle the data. Machine accuracy and performance may be greatly enhanced by label encoding, which is easy to use and effective.

Label encoding

Utilizing the label encoder class is necessary for label encoding. Quantitative variables may be created from category variables using the sci-kit-learn Label Encoder class, a preprocessing tool. To make it work, an integer value is assigned to each category in the variable. To find the distinct categories in the data, use the fit() function to create and fit the Label Encoder class to the category variable. Following that, the category variable may be changed into a numerical variable by using the convert() function.

Feature Selection

Feature selection is the process of selecting a subset of relevant characteristics to be utilized in machine learning models from a larger set of features. This is done to shorten calculation times, increase accuracy, and simplify the model. Feature selection is an important step in the machine learning pipeline that ensures accurate and productive analysis. This study uses lasso regularization for feature selection.

Lasso regularization

To minimize over fitting, Lasso regularization is a machine-learning approach that adds a penalty term to the cost function, thereby reducing the less significant attributes to zero. It

produces sparse models, which enhance the interpretability and performance of the model by retaining just the most crucial elements. Cross-validation may be used to adjust lambda, a hyper parameter that controls the degree of regularization.

Following feature selection, a total of eighteen features are selected from the group of twenty-six features. the subsequent eighteen attributes: Blood pressure (mm/hg), age (years), specific gravity, albumin, sugar, and Blood pressure, anemia, diabetes mellitus, appetite, pedal edema, blood urea (mg/dL), blood glucose random (mg/dL), blood creatinine (mg/dL), serum creatinine (mg/dL), sodium (mEq/L), and potassium (mEq/L), packed cell volume, red blood cells (millions/cmm), and blood pressure are more important measurements.

Algorithms

This study employed five supervised algorithms. After that, the training and testing datasets are created using the updated dataset from the earlier phases. The training dataset is used to put the algorithms into practice.

i. Logistic regression

In machine learning, problems requiring binary categorization-like forecasting a customer's likelihood of making a purchase based on one or more input variables are resolved by the statistical technique known as logistic regression.

The likelihood of the binary outcome, which has a range of 0 to 1, is determined by the logistic regression model using a logistic function. Using a threshold value, the model then groups the predicted probability into one of the two classes.

ii. Support vector machine

Applications for both regression and classification can use the supervised machine learning algorithm SVM. In regression, SVM looks for a function that predicts a continuous output variable, but in classification, it looks for a decision border that separates the data into discrete groups.

iii. Gradient boosting

Gradient boosting is a popular machine learning technique for applications involving classification and regression. It's an ensemble learning method that creates a powerful learner (like decision trees) by combining several weak learners. Iteratively adding decision trees to the model is done via the gradient boosting approach.

Every tree is trained to predict the residuals of the previous trees that is, the differences between the actual and projected values. Stated differently, each new tree is educated to correct errors caused by its predecessors.

iv. Random Forest Classifier

For classification issues, where the goal is to predict a data point's class based on its properties, the Random Forest Classifier is a popular ensemble learning technique. The random forest strategy increases the model's robustness and accuracy by combining many decision trees.

v. Xgboost

Regression and classification problems are commonly addressed with XGBoost, short for eXtreme Gradient Boosting. Using this efficient ensemble learning method, several weak decision trees are combined to create a powerful classifier or regressor. XGBoost trains its models by employing this technique, and it is based on the gradient boosting framework. Recurrent decision tree training using the residuals of the previous trees, it increases the model's prediction accuracy.

Classification metrics

The model's performance is monitored and assessed by machine learning measures. After the machine learning models are applied to the dataset, a comparison of the models using different metrics is required. Since we are conducting categorization in this instance, we must apply the classification metrics. In this work, we employed the following categorization metrics: accuracy, precision, recall, f1-score, and error rate.

RESULTS

In this study, the dataset was initially divided into numerical and categorical columns, and the null values in each column were removed. For removing null values in numerical and categorical columns, we used random value imputation and mode imputation, respectively, we have turned the non-numeric columns into numeric columns using scikit-learn's label encoder class. The modified dataset is then divided into testing and training for the implementation of the five algorithms and is then compared using the classification metrics.

Algorithms	Random Forest	Support Vector Machine	Xgboosting	Logistic Regresion	Gradient Boosting
Classification Metrices					
Accuracy	96.3 %	59.4 %	95.1 %	89.9 %	93.5 %
F1 score	96.5 %	39.5 %	95.3 %	89.7 %	93.7 %
Recall	96.9 %	25.3 %	93.8 %	84.6 %	91.5 %
precision	96.1 %	89.1 %	96.8 %	95.6 %	95.9 %
errorrate	3.6 %	40.0 %	4.8 %	10%	6.4 %

Table 1: comparison of performances of different algorithms based on classification metrics.

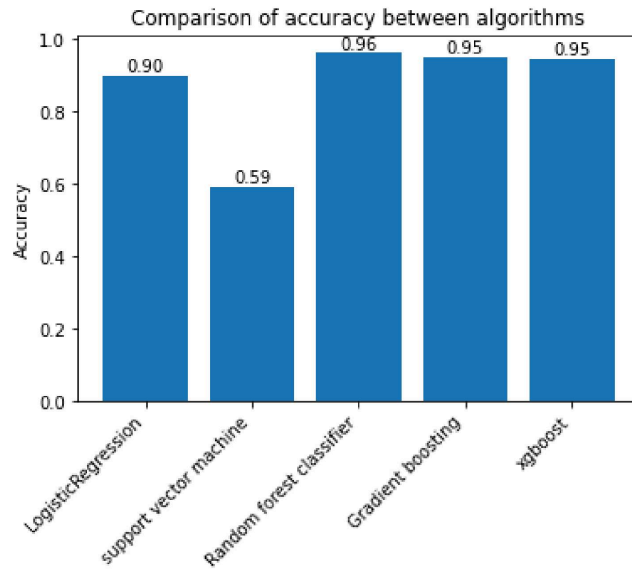


FIG 2: Comparison of accuracy between algorithms

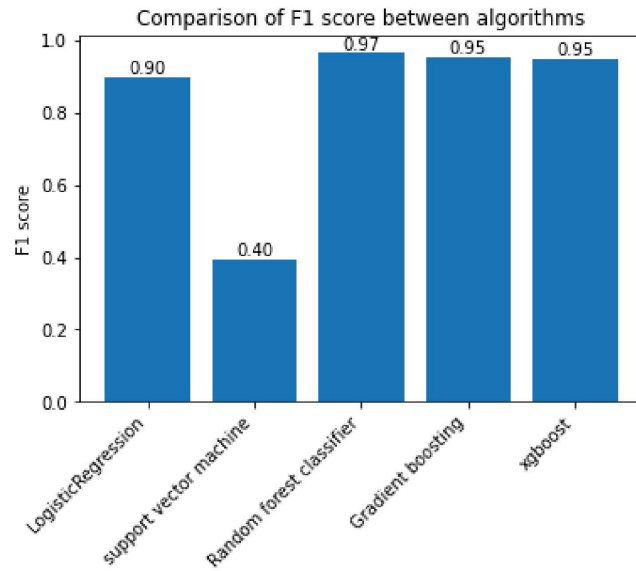


Fig 3: COMPARISON OF F1 SCORE BETWEEN ALGORITHMS

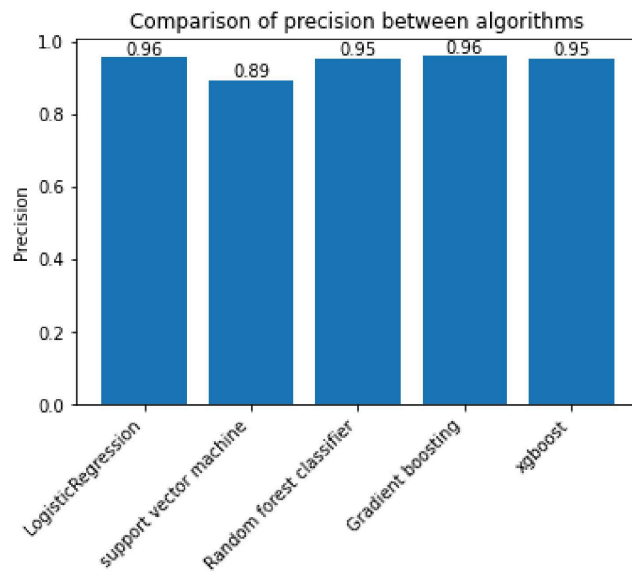


FIG 4: COMPARISON OF PRECISION BETWEEN ALGORITHMS

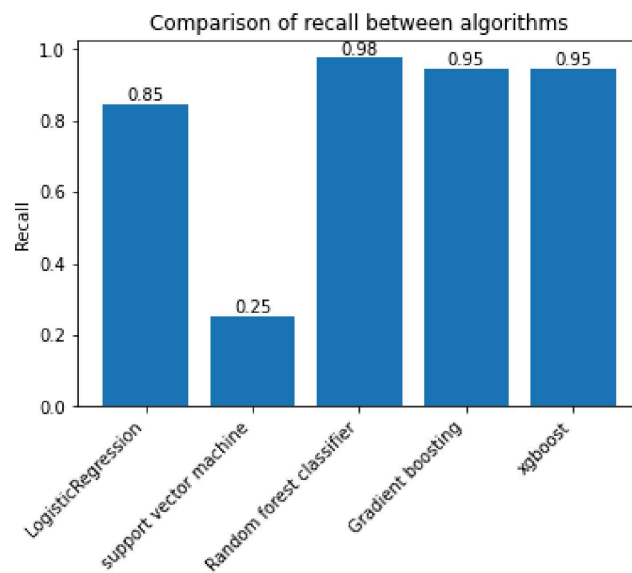


fig 5: COMPARISON OF RECALL BETWEEN ALGORITHMS.

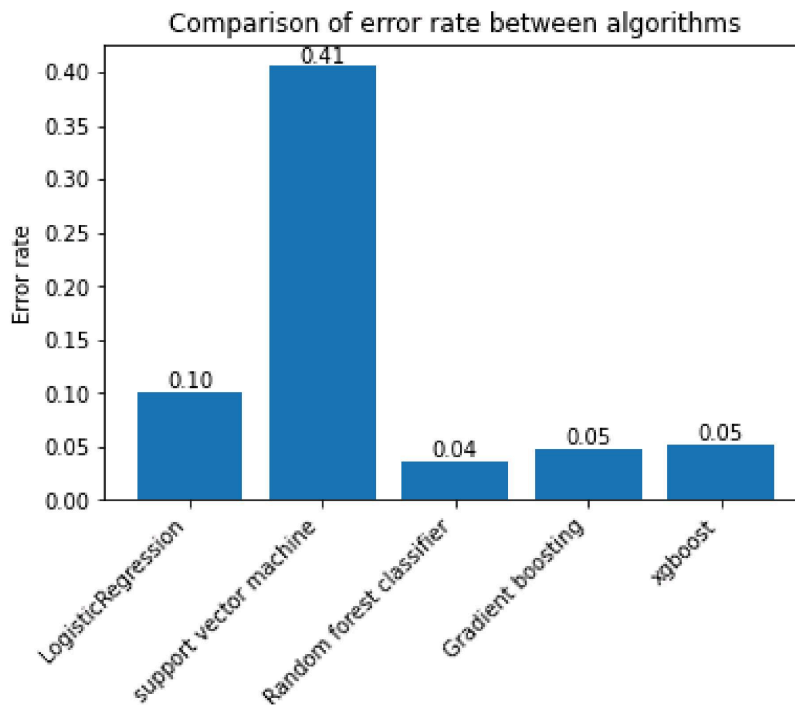


fig 5: COMPARISON OF ERROR RATE BETWEEN ALGORITHM

CONCLUSION AND FUTURE SCOPE

Of the five methods, the Random Forest Classifier performs the best. With support vector machines at the bottom. Similar comparisons between the random forest and other machine- learning algorithms can be made in future studies. A quick analysis can also be carried out if the dataset is larger or better data is provided. With the use of this research, a website allowing usersto input test data and estimate their risk of acquiring kidney cancer can be developed. It is possible to create a real-time system that offers the patient or end user the greatest possible communication.

REFERENCES

1. S.Vijayanil and S.Dayananda, "Kidney disease prediction using SVM and ANN algorithms', Int. J. Comput. Bus. Res., vol. 6, no. 2, pp. 1-12, Mar. 2015.
2. J. Xiao, R. Ding, X. Xu, H. Guan, X. Feng, T. Sun, et al., "Comparison and development of machine learning tools in the prediction of chronic kidney disease progression", J. Transl. Med., vol. 17, pp. 119,Dec. 2019.
3. J. M. Pereira, M. Basto, and A. F. D. Silva, "The logistic lasso and ridge regression in predicting

- corporate failure", *Procedia Econ. Finance*, vol. 39, pp. 634-641, Jan. 2016.
4. K. R. A. Padmanaban and G. Parthiban, "Applying machine learning techniques for predicting the risk of chronic kidney disease". *Indian J. Sci. Technol.*, vol. 9,no.29, Aug. 2016
 5. R. Ruiz-Arenas, "A summary of worldwide national activities in chronic kidney disease (CKD) testing, the electronic journal of the International Federation of," *Clinical Chemistry and Laboratory Medicine*, vol.28, no.4, pp.302-314,2017.
 6. George C, Mogueo A, Okpechi I, Echouffo- Tcheugui JB, Kengne AP. Chronic kidney disease in low- income to middle- income countries: The case increased screening. *BMJ Glob Heal*. 2017;2(2):1-10.
 7. Stanifer JW, et al. The epidemiology of chronic kidney disease in sub- Saharan Africa: A systematic review and meta-analysis. *Lancet Glob Heal*.2014;2(3):e174-81
 8. AbdElhafeez S, Bolignano D, D'Arrigo G, Dounousi E, Tripepi G, Zoccali C. Prevalence and burden of chronic kidney disease among the general population and high- risk groups in Africa: A systematic review. *BMJ Open*.2018;8:1.
 9. Molla MD, et al. Assessment of serum electrolytes and kidney function test for screening of chronic kidney disease among Ethiopian Public Health Institute staff members, Addis Ababa, Ethiopia. *BMC Nephrol*.2020;21(1):494.
 10. Eroğlu K., Palabaş T. 2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO) IEEE; 2016. The impact on the classification performance of the combined use of different classification methods and different ensemble algorithms in chronic kidney disease detection; pp. 512-516.
 11. Fatima M., Pasha M. Survey of machine learning algorithms for disease diagnostic. *J Intel Learn Syst Appl*. 2017;9(01):1.
 12. Gudeti B., Mishra S., Malik S., Fernandez T.F., Tyagi A.K., Kumari S. 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA) IEEE; 2020. A novel approach to predict chronic kidney disease using machine learning algorithms; pp. 1630-1635.
 13. Heung M., Chawla L.S. Predicting progression to chronic kidney disease after recovery from acute kidney injury. *Curr Opin Nephrol Hypertens*. 2012;21:628
 14. Islam M., Shampa M., Alim T., et al. Convolutional neural network-based marine cetaceans detection around the swatch of no ground in the bay of Bengal. *Int J Comput Digit Syst*. 2021;12:877- 893.
 15. Islam, M.A., Akhter, T., Begum, A. Hasan, M.R., Rafi, F.S. Brain tumor detection from MRI images using image processing.
 16. Islam M.A., Akter S., Hossen M.S., Keya S.A., Tisha S.A., Hossain S. 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) IEEE; 2020. Risk factor prediction of chronic kidney disease based on machine learning algorithms; pp. 952-957.
 17. Islam M.A., Hasan M.R., Begum A. Improvement of the handover performance and channel allocation scheme using fuzzy logic, artificial neural network, and neuro-fuzzy system to reduce call drop in a cellular network. *J Eng Adv*. 2020;1:130-138.
 18. Mahesh B. Machine learning algorithms-a review. *Int J Sci Res (IJSR)*. [Internet] 2020;9:381

19. Stanifer JW, et al. The epidemiology of chronic kidney disease in sub-Saharan Africa: A systematic review and meta-analysis. *Lancet Glob Heal.* 2014;2(3):e174–81. Network. *J Eng Adv.* 2020;1:130–138.
20. George C, Mogueo A, Okpechi I, Echouffo-Tcheugui JB, Kengne AP. Chronic kidney disease in low-income to middle-income countries: The case for increased screening. *BMJ Glob Heal.* 2017;2(2):1–10.
21. Gudeti B., Mishra S., Malik S., Fernandez T.F., Tyagi A.K., Kumari S. 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA) IEEE; 2020. A novel approach to predict chronic kidney disease using machine learning algorithms; pp. 1630–1635.

D.CERTIFICATE

