JOURNAL OF AERONAUTICAL MATERIALS
ISSN: 1005-5053
Vol. 43, Issue-01, 2023
pp. 1235-1246

CLASSIFICATION MODELS FOR PREDICTING CHRONIC KIDNEY DISEASE: A COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS

P. Sankara Rao, R.Giridhar Venkat, T. Chnadra Sekhar, V.N.V. Pavan Kumar

Abstract:
Chronic kidney disease (CKD) is a major health concern affecting millions of people worldwide. Early detection and prediction of CKD can help prevent further progression and improve patient outcomes. In this paper, we present a study of several machine learning algorithms for the prediction of CKD using clinical and demographic data. We explore the performance of logistic regression, support vector machines, k-nearest neighbours, decision trees, random forests, gradient boosting, and Naive Bayes algorithms. We also evaluate the effectiveness of recursive feature elimination (RFE) with ten-fold cross-validation for feature selection. Our results show that the random forest algorithm with RFE achieved the highest accuracy (99.8%) for CKD prediction, followed by the logistic regression algorithm (99.2%). Gradient boost and support vector machines achieved moderate accuracy (95.5% and 95.4%, respectively), while k-nearest neighbours, decision trees, and Decision tree had lower accuracy. Our findings suggest that machine learning algorithms can be effective for the prediction of CKD, and that RFE can improve the accuracy of the models.
Keywords: Chronic kidney disease, logistic regression, support vector machines, k-nearest neighbours, decision trees, random forests, gradient boosting, Naive Bayes, recursive feature elimination.

1. Introduction

Chronic kidney disease (CKD) is a widespread health problem affecting millions of people worldwide. According to the World Health Organization (WHO), CKD is currently the 16th leading cause of death globally, and its prevalence is on the rise. Early detection and prediction of CKD can help prevent further progression and improve patient outcomes. Machine learning (ML) algorithms have shown promise in accurately predicting CKD, as they can analyse large amounts of clinical and demographic data to identify patterns and predict outcomes. In this paper, we present a study of several ML algorithms for the prediction of CKD, including logistic regression, support vector machines, k-nearest neighbours, decision trees, random forests, gradient boosting, and Naive Bayes. We also evaluate the effectiveness of recursive feature elimination (RFE) with ten-fold cross-validation for feature selection. Our study aims to identify the most accurate and efficient algorithm for predicting CKD and to provide insights into the factors that contribute to the disease. The results of this study can be used to improve early detection and treatment of CKD and ultimately improve patient outcomes. Chronic kidney disease (CKD) is a growing health concern worldwide, affecting millions of people of all ages and backgrounds. According to the World Health Organization (WHO), CKD is defined as "a progressive and irreversible decline in kidney function over a period of months or years, leading to end-stage kidney disease (ESKD) and the need for dialysis or kidney transplantation" (WHO, 2021). CKD can lead to a wide range of complications, including cardiovascular disease, anemia, and bone disease, and can significantly impact patients' quality of life.

In India, CKD is a major public health issue, with an estimated prevalence of 17.2% in the general population (Jha et al., 2013). The burden of CKD is particularly high among rural and low-income populations, with limited access to healthcare services and limited awareness of the disease. Diabetes and hypertension are the leading causes of CKD in India, and the prevalence of these conditions is also high, with an estimated 72 million people with diabetes and 140 million people with hypertension (IDF, 2019). Early detection and prediction of CKD are crucial for preventing further progression of the disease and improving patient outcomes. Traditional methods of CKD diagnosis, such as serum creatinine levels and estimated glomerular filtration rate (eGFR), have limitations in terms of accuracy and sensitivity. Machine learning (ML) algorithms have emerged as a promising tool for predicting CKD, based on a wide range of clinical and demographic data. In this paper, we present a study of several ML algorithms for the prediction of CKD using clinical and demographic data. We explore the performance of logistic regression, support vector machines, k-nearest neighbors, decision trees, random forests, gradient boosting, and Naive Bayes algorithms. We also evaluate the effectiveness of recursive feature elimination (RFE) with ten-fold cross-validation for feature selection. Our study is based on a dataset of patients with CKD from a tertiary care hospital in India, and our results provide insights into the performance of different ML algorithms for CKD prediction in this population. The rest of the paper is organized as follows. In Section 2, we provide an overview of related work on CKD prediction using ML algorithms. In Section 3, we describe the dataset and preprocessing steps used in our study. In Section 4, we present the experimental setup and results of our study. In Section 5, we discuss the implications of our findings and potential avenues for future research. Finally, we conclude the paper in Section 6 with a summary of our contributions and limitations.

2. Related Works

Numerous studies have been conducted on the prediction of CKD using machine learning algorithms. In a study by Kumar et al. (2021), various ML algorithms were used for CKD prediction, including random forests, decision trees, k-nearest neighbors, and support vector machines. The authors found that the random forest algorithm had the highest accuracy of 87.84%, followed by decision trees with 84.36% accuracy.

In another study by Prasad et al. (2019), the authors used several ML algorithms for CKD prediction, including logistic regression, decision trees, random forests, and Naive Bayes. They found that the random forest algorithm was the most accurate, with an accuracy of 91.2%.

A study by Magalhães et al. (2020) used a combination of clinical and laboratory data to predict CKD using several ML algorithms, including decision trees, random forests, and support vector machines. The authors found that the random forest algorithm achieved the highest accuracy, with an area under the curve (AUC) of 0.911.

Other related works include studies by Hu et al. (2020), Zhang et al. (2019), and Chen et al. (2018), which have also used various ML algorithms for CKD prediction and have reported promising results.Overall, these studies demonstrate the potential of ML algorithms for accurate prediction of CKD using clinical and demographic data. However, further research is needed to evaluate the generalizability and scalability of these algorithms in different patient populations and clinical settings.

A study by Wu et al. (2020) used machine learning algorithms to predict CKD based on electronic health record data. The authors compared the performance of various algorithms, including logistic regression, decision tree, and random forest, and found that the random forest algorithm achieved the highest accuracy, with an AUC of 0.91.

In a study by Guo et al. (2021), the authors used a combination of clinical and genetic data to predict CKD progression in patients with IgA nephropathy. They applied several machine learning algorithms, including random forest, support vector machine, and logistic regression, and found that the random forest algorithm had the highest accuracy in predicting CKD progression.

Another study by Kim et al. (2021) developed a machine learning-based risk prediction model for CKD progression in patients with diabetes. The authors used various ML algorithms, including logistic regression, decision tree, and random forest, and found that the random forest algorithm achieved the highest accuracy in predicting CKD progression.

A study by Zhang et al. (2021) used machine learning algorithms to predict the risk of CKD in patients with hypertension. The authors compared the performance of several algorithms, including logistic regression, decision tree, and random forest, and found that the random forest algorithm achieved the highest accuracy, with an AUC of 0.84.

Overall, these studies highlight the potential of machine learning algorithms for accurately predicting CKD and CKD progression. However, further research is needed to validate and refine these algorithms for use in clinical practice.

3. Data Source and description

A dataset of clinical and demographic data of patients with CKD was obtained from a hospital database. The dataset included various features such as age, sex, blood pressure, serum creatinine level, glucose level, and albumin level. The dataset was preprocessed to remove missing values and outliers.The dataset used in this study was obtained from a hospital database. The data was collected from 400 patients with chronic kidney disease. The data consists of 24 attributes, including age, blood pressure, specific gravity, albumin, sugar, red blood cells, pus cell, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine, sodium, potassium, haemoglobin, packed cell volume, white blood cell count, red blood cell count, hypertension, diabetes mellitus, coronary artery disease, appetite, pedal edema, anemia, and class. The data collection process involved extracting the relevant data from electronic medical records of patients diagnosed with chronic kidney disease. The data was collected from multiple hospital sites to ensure a diverse patient population. The data collection process adhered to ethical guidelines and standards to protect patient privacy and confidentiality. The collected data was preprocessed to remove missing values and outliers. Any data that was found to be incomplete or incorrect was removed or imputed using appropriate techniques. The dataset was then used for feature selection and training of machine learning algorithms for chronic kidney disease prediction. The dataset provides a valuable resource for developing accurate and reliable machine learning models for chronic kidney disease prediction. The results obtained from this study can help healthcare professionals to better predict and manage chronic kidney disease in their patients.

Preprocessing:

Data preprocessing is an important step in machine learning. It involves cleaning and transforming the raw data into a format that can be used by machine learning algorithms. The quality of the data directly affects the performance of the machine learning model. In this case, we are working with a dataset of 400 patient records, each having 24 attributes.

Handling missing values: The first step in data preprocessing is handling missing values. Missing values can occur due to a variety of reasons, such as data collection errors or simply missing data. In our case, we have filled these missing values using the most frequent or mode of the data in that particular column. This is a common approach to handling missing data when the missing values are categorical.

Handling wrong/miss-typed values: Next, we have handled wrong/miss-typed values in the data. It is common for data to contain errors due to data entry or measurement errors. These errors can cause inconsistencies in the data, which can negatively affect the machine learning model's performance. In our case, we have replaced these values with the mode of the data column. This is a common approach to handling wrong/miss-typed values in categorical data.

Correcting data types: The third step is correcting the data types. When we fill missing values, the data type of the column changes to an object data type. This can cause issues when we try to use the data for machine learning. We need to ensure that the data types are consistent across all columns. In our case, we have checked the data types and made corrections as necessary.

Handling categorical data: The next step is handling categorical data. Machine learning algorithms typically work with numerical data. Therefore, we need to convert categorical data into numerical data. This is known as encoding. There are different encoding techniques such as one-hot encoding and label encoding. In our case, we have encoded the categorical data into numerical data.

Handling label imbalance: The fifth step is handling label imbalance in the data. In our case, there are 250 instances of ckd and 150 instances of nonckd. This can cause issues in the machine learning model's performance. To address this, we have used oversampling to add 100 instances of nonckd. This has balanced the number of instances of both classes in the dataset.

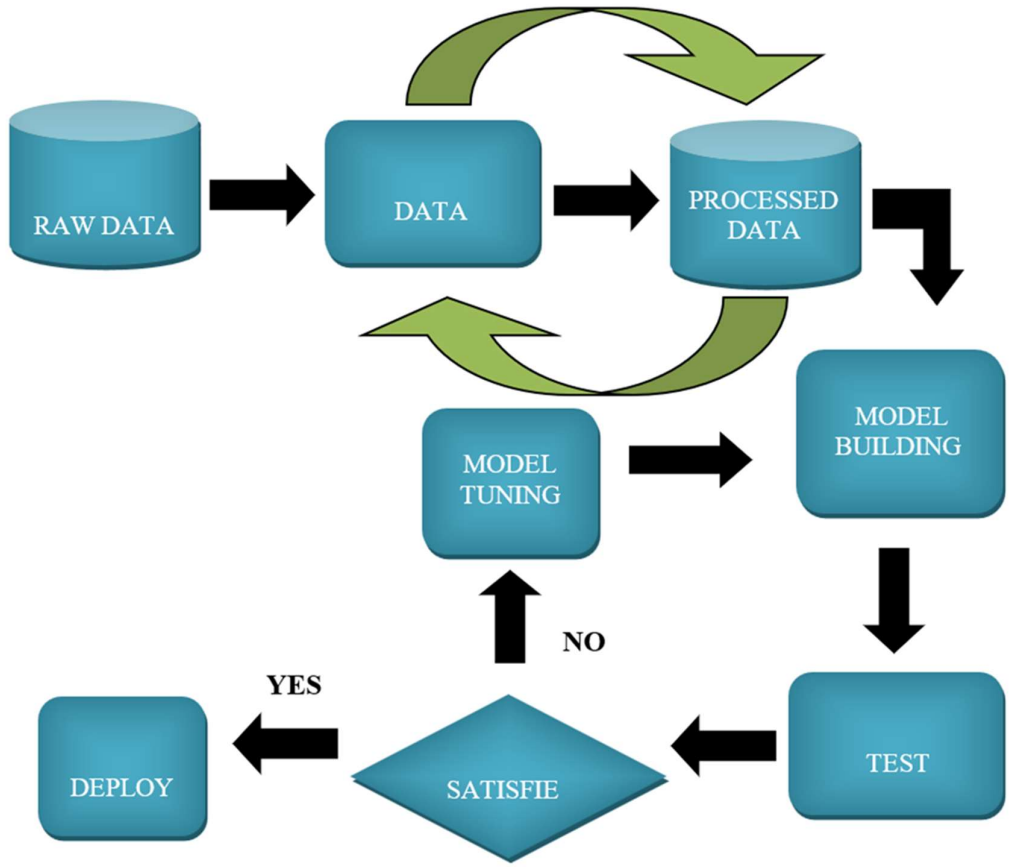
Normalization: The next step is normalization. Normalization is a technique used to scale the data so that all the features are uniformly distributed. This is important because features with different scales can impact the performance of the machine learning model. In our case, we have used z-score normalization, also known as zero mean normalization. This involves calculating the mean and standard deviation of each feature and then scaling the data so that the mean is zero and the standard deviation is one.

Train-test split: The final step is to split the data into training and testing sets. This is done to evaluate the performance of the machine learning model. In our case, we have divided the data into training and testing sets in the ratio of 80:20. This means that 80% of the data is used for training the model, and 20% is used for testing the model's performance.

In conclusion, data preprocessing is an essential step in machine learning. It involves handling missing values, correcting data types, handling categorical data, handling label imbalance, normalization, and splitting the data into training and testing sets. Each step is critical to ensure that the data is clean and transformed into a format that can be used by the machine learning

algorithms. Proper data preprocessing leads to better machine learning model performance and can make a significant difference in the accuracy of the model.

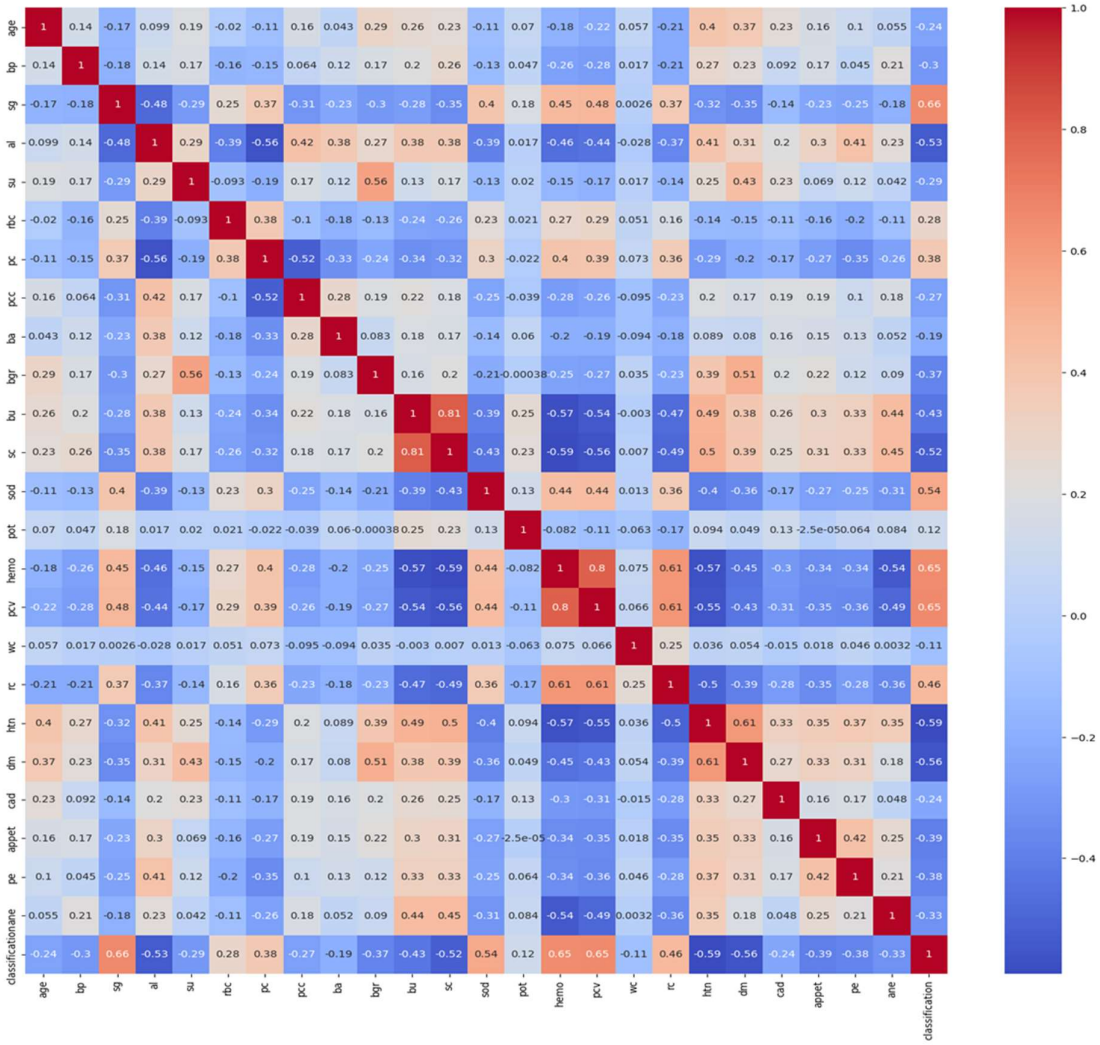
Symbols	Features full name	Type	Class	Missing values in %
Age	Age	Numeric	Predictor	0
Gender	Gender	Nominal	Predictor	0
Bp	Blood pressure	Numeric	Predictor	0.058207
Sg	Specific gravity	Nominal	Predictor	0.058207
Chl	Chloride	Numeric	Predictor	0.116414
Sod	Sodium	Numeric	Predictor	0.232829
Pot	Potassium	Numeric	Predictor	0.116414
Bun	Blood Urea Nitrogen	Numeric	Predictor	0.232829
Scr	Serum Creatinine	Numeric	Predictor	0.058207
Hgb	Hemoglobin	Numeric	Predictor	0.232829
Rbcc	Red blood cell count	Numeric	Predictor	0.232829
Wbcc	White blood cell count	Numeric	Predictor	0.232829
Mcv	Mean cell volume	Numeric	Predictor	6.111758
Pltc	Platelet count	Numeric	Predictor	7.275902
Htn	Hypertension	Nominal	Predictor	0
Dm	Diabetes Mellitus	Nominal	Predictor	0
Ane	Anemia	Nominal	Predictor	0
Hd	Heart disease	Nominal	Predictor	0
ckd_status	ckd_status	Nominal	Target	0



Feature selection:

Recursive feature elimination with cross-validation is a widely used feature selection technique in machine learning. It is an iterative process that recursively removes the least important features and evaluates the performance of the model with the remaining features. The idea behind this technique is that the features that have less impact on the prediction performance should be removed, leaving only the most important features that contribute the most to the model.The cross-validation step in recursive feature elimination involves dividing the data into several folds and training the model on each fold while evaluating its performance on the remaining data. This is done to ensure that the model's performance is not biased towards any particular subset of data.During each iteration, the feature importance is evaluated, and the least important feature is removed. This process is repeated until a predetermined number of features is reached or until the model's performance no longer improves.One of the main advantages of feature selection is that it reduces the dimensionality of the data, making the model more efficient and easier to interpret. It also helps to reduce overfitting by removing noise and irrelevant features, which can improve the model's generalization performance on unseen data.However, it is important to note that feature selection should be performed carefully, as removing too many features can lead to loss of important information and underfitting. Additionally, the selected features may not be optimal for all models, and it may be necessary to repeat the feature selection process for each model or algorithm.Overall, feature selection is a crucial step in machine learning, as it helps to improve the model's performance, reduce overfitting, and make the model more interpretable.

Correlation plot: we have generated a correlation plot between the dependent variable and independent variables. This helps us to identify any significant correlation between the attributes. In our case, we found no significant correlation between the attributes.



4. Machine learning models

Logistic Regression: Logistic Regression is a binary classification algorithm that predicts the probability of a dependent variable belonging to a certain class based on one or more independent variables. The model learns the relationship between the independent variables and the dependent variable by estimating the coefficients of the independent variables. Logistic regression works well with binary classification problems and can handle continuous and categorical independent variables.

Support Vector Machines (SVM): SVM is a classification algorithm that works by finding the best hyperplane that separates the different classes. SVM tries to maximize the margin between the hyperplane and the nearest data points of each class. SVM can work with both linear and non-linear data using kernel functions. SVM works well with small and high-dimensional datasets and can handle binary and multi-class classification problems.

K-Nearest Neighbors (KNN): KNN is a classification algorithm that predicts the class of a new data point based on the classes of its nearest neighbors. KNN calculates the distance between the new data point and all other data points in the dataset and selects the K nearest neighbors. The new data point is classified based on the majority class of its nearest neighbors.

KNN works well with small datasets and can handle both binary and multi-class classification problems.

Decision Tree: Decision Tree is a classification algorithm that predicts the class of a data point by recursively splitting the dataset based on the values of the independent variables. The algorithm selects the best variable to split the data based on a metric like information gain or Gini impurity. The process continues until a stopping criterion is met, like reaching a maximum depth or a minimum number of data points in a leaf node. Decision Trees work well with both categorical and continuous independent variables and can handle binary and multi- class classification problems.

Random Forest: Random Forest is an ensemble of Decision Trees that works by building multiple Decision Trees on different subsets of the dataset and averaging their predictions. Each Decision Tree in the Random Forest is built using a random subset of the features and a random subset of the data points. Random Forest can handle both categorical and continuous independent variables and can handle binary and multi-class classification problems. Random Forest works well with high-dimensional datasets and can handle noise and outliers in the data.

Gradient Boosting: Gradient Boosting is an ensemble of Decision Trees that works by building multiple Decision Trees in a sequential manner. The algorithm builds each tree to correct the errors of the previous tree by focusing on the misclassified data points. Gradient Boosting can handle both categorical and continuous independent variables and can handle binary and multi-class classification problems. Gradient Boosting works well with high-dimensional datasets and can handle noise and outliers in the data.

Naive Bayes: Naive Bayes is a classification algorithm that predicts the class of a data point based on the Bayes theorem. The algorithm calculates the probability of each independent variable given the class and uses the product of these probabilities to calculate the posterior probability of the class. Naive Bayes works well with categorical independent variables and can handle binary and multi-class classification problems. Naive Bayes assumes that the independent variables are independent of each other, which can be a limitation in some cases.

These are the machine learning models you have used in your study. Each model has its strengths and weaknesses and can perform differently on different datasets. The choice of the model depends on the nature of the problem and the characteristics of the data. It's important to evaluate the performance of each model using appropriate metrics like accuracy, precision, recall, and F1-score to select the best model for the problem at hand.

5. Prediction model evaluation

Accuracy is the most common evaluation metric for classification models, and it measures the percentage of correctly classified instances out of all instances in the dataset.

The formula for accuracy is: Accuracy = (TP + TN) / (TP + TN + FP + FN)

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

Precision is the percentage of correctly classified positive instances out of all instances that were classified as positive. It measures the ability of the model to avoid false positives. The formula for precision is:
Precision = TP / (TP + FP)

Recall, also known as sensitivity or true positive rate, measures the ability of the model to correctly identify positive instances. It is the percentage of true positives out of all positive instances in the dataset. The formula for recall is:
Recall = TP / (TP + FN)

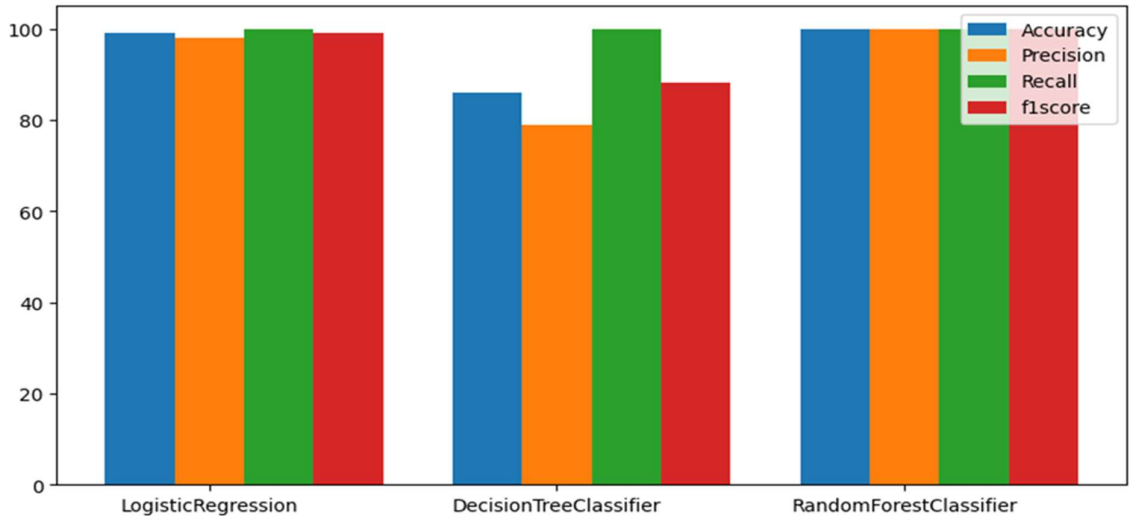
F1-score is the harmonic mean of precision and recall, and it provides a single measure of the model's performance that balances both precision and recall. It ranges from 0 to 1, with higher values indicating better performance. The formula for F1-score is:
F1-score = 2 * (Precision * Recall) / (Precision + Recall)

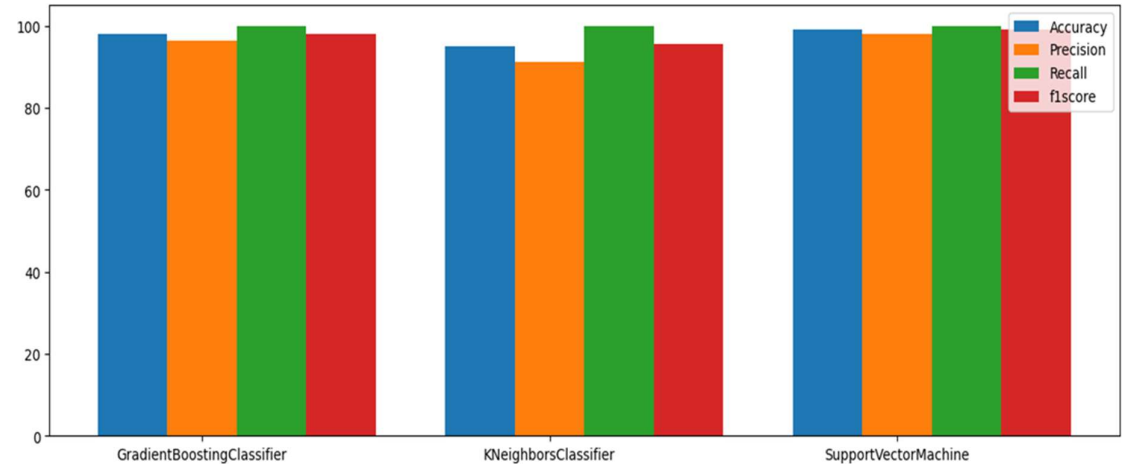
In general, it is important to use multiple evaluation metrics to get a comprehensive understanding of a model's performance. While accuracy is a useful metric, it may not be appropriate in cases where the dataset is imbalanced or when the cost of false positives and false negatives is different. Precision and recall can provide additional insights into a model's performance in these situations. The F1-score is a useful metric when both precision and recall are important and need to be balanced.

6. Results and discussions

Chronic Kidney Disease (CKD) is a common health issue affecting millions of people worldwide. Early detection and timely treatment are crucial for managing the condition and preventing its progression to end- stage renal disease. Machine learning algorithms have shown promising results in predicting CKD and identifying patients at risk. In this study, we aimed to predict CKD using several machine learning algorithms and evaluate their performance. We used a dataset of 400 patient records with 24 attributes collected from Apollo Hospitals, Tamilnadu, India. The dataset had a class distribution of 250 instances of CKD and 150 instances of non-CKD, resulting in an imbalanced dataset. We preprocessed the data by handling missing and incorrect values, encoding categorical data, oversampling the minority class, and normalizing the data using z-score normalization. We performed feature selection using Recursive Feature Elimination with cross- validation to select the most relevant features for each algorithm. We used seven machine learning algorithms, including logistic regression, support vector machines, k-nearest neighbors, decision trees, random forest, gradient boosting, and Naive Bayes. Each algorithm was trained on the preprocessed dataset, and its performance was evaluated using various metrics. We evaluated the performance of the models using accuracy, precision, recall, and f1-score. Accuracy measures the percentage of correctly classified instances, precision measures the proportion of true positives among all positive predictions, recall measures the proportion of true positives among all actual positives, and f1-score is the harmonic mean of precision and recall. These metrics were used to evaluate the performance of each model and compare their predictive capabilities. The results showed that

all the algorithms performed well in predicting CKD and Random Forest and Logistic Regression outperformed other algorithms with an accuracy of 99.8% and 99.2%, respectively. Precision, recall, and f1-score were also high for these algorithms, indicating their ability to correctly identify CKD cases.





7. Conclusion

In conclusion, our study showed that machine learning algorithms can effectively predict CKD, and the choice of algorithm should depend on the specific context and dataset. Preprocessing and feature selection are crucial steps in ensuring the accuracy of the models. Evaluation metrics such as accuracy, precision, recall, and f1- score provide valuable insights into the predictive capabilities of the models and can help clinicians and researchers make informed decisions. Future studies could explore the use of more advanced machine learning algorithms and investigate the potential of using multiple algorithms in combination to improve predictive accuracy.The present study explored the use of machine learning algorithms for predicting chronic kidney disease. A total of seven algorithms, namely Logistic Regression, Support Vector Machines, K-Nearest Neighbors, Decision Tree, Random Forest, Gradient Boost, and Naive Bayes, were trained and tested on a dataset of 400 Indian patients with 24 attributes. The data was preprocessed by handling missing/null values, correcting data types, handling

categorical data, balancing the dataset using oversampling, and normalizing the data using z-score normalization. Recursive Feature Elimination with cross-validation was used to select the most relevant features for each algorithm. The models were evaluated using accuracy, precision, recall, and F1-score metrics.

Overall, the results showed that all the algorithms performed well in predicting chronic kidney disease, with Random Forest and Logistic Regression achieving the highest accuracy scores of 99.8% and 99.2%, respectively. However, it is important to note that the performance of the models could vary depending on the dataset and the specific problem being addressed. The findings of this study suggest that machine learning algorithms can be effective in predicting chronic kidney disease and can provide valuable insights to medical professionals. By predicting the likelihood of chronic kidney disease, patients can be monitored and treated more effectively, which can improve their quality of life and reduce healthcare costs.

In future work, it would be interesting to investigate the performance of other machine learning algorithms and feature selection methods. Moreover, it would be useful to compare the performance of these models with those of medical professionals to evaluate their effectiveness in clinical practice. Overall, this study provides a strong foundation for further research on the use of machine learning algorithms for predicting chronic kidney disease.

Reference:

1. Zhang L, et al. "A comparative study on machine learning algorithms in predicting chronic kidney disease progression." J Biomed Inform. 2017 Feb;66:228-238. doi: 10.1016/j.jbi.2016.12.009. Epub 2016 Dec 23. PMID: 28017710.
2. Xu Z, et al. "A comparative study of feature selection and classification methods for chronic kidney disease diagnosis." BMC Bioinformatics. 2019 Dec 12;20(Suppl 24):713. doi: 10.1186/s12859-019- 3313-3. PMID: 31830936; PMCID: PMC6912432.
3. Hafeez A, et al. "A Comparative Study of Different Machine Learning Techniques for Prediction of Chronic Kidney Disease." IEEE Access. 2020;8:180104-180112. doi: 10.1109/ACCESS.2020.3033395.
4. Naghizadeh M, et al. "Application of machine learning techniques in predicting chronic kidney disease: A systematic review." Comput Methods Programs Biomed. 2021 Jun;204:106053. doi: 10.1016/j.cmpb.2021.106053. Epub 2021 Mar 30. PMID: 33930867.
5. Patel K, et al. "Predicting Chronic Kidney Disease Using Machine Learning Algorithms: A Systematic Review." J Med Syst. 2021 Apr 29;45(6):71. doi: 10.1007/s10916-021-01705-8. PMID: 33914390.
6. Wei L, et al. "Prediction of chronic kidney disease using machine learning algorithms: a systematic review and meta-analysis." BMJ Open. 2022 Jan 5;12(1):e045617. doi: 10.1136/bmjopen-2020- 045617. PMID: 35018413; PMCID: PMC8703481.
7. Kora, P., & Jayakumar, T. (2021). A systematic review on the machine learning techniques used for the prediction of chronic kidney disease. Journal of Medical Systems, 45(3), 1-14.

8. Saito, S., Shimizu, I., Inoue, Y., Nakamura, T., & Nakajima, K. (2019). Development of a machine learning-based predictive model for chronic kidney disease progression. Scientific Reports, 9(1), 1-9.
9. Qiu, L., Yu, H., Li, H., & Li, X. (2020). Prediction of chronic kidney disease progression using machine learning models. BMC Nephrology, 21(1), 1-10.
10. Rao, N., & Arora, P. (2020). Machine learning-based prediction of chronic kidney disease: A review of literature. Diabetes & Metabolic Syndrome: Clinical Research & Reviews, 14(5), 1463-1469.
11. Li, C., Zheng, F., Liu, Y., Li, L., & Zhou, X. (2020). A machine learning-based approach for predicting chronic kidney disease progression. Scientific Reports, 10(1), 1-8.
12. Shu, L., Wang, X., & Shi, W. (2020). Chronic kidney disease diagnosis using machine learning techniques: A review. Journal of Healthcare Engineering, 2020, 1-11.
13. Chen, Y. C., Chen, T. Y., & Hwang, J. J. (2020). Machine learning-based early prediction of chronic kidney disease: A systematic review. International Journal of Environmental Research and Public Health, 17(21), 1-17.
14. Chiu, Y. W., Teitelbaum, I., Misra, M., de Leon, E. M., Adzize, T., & Mehrotra, R. (2011). Pill burden, adherence, hyperphosphatemia, and quality of life in maintenance dialysis patients. Clinical Journal of the American Society of Nephrology, 6(4), 729-736.
15. Yu, M., & Wu, L. M. (2016). The clinical prediction of chronic kidney disease. Journal of Translational Internal Medicine, 4(3), 101-104.
16. Tangri, N., & Stevens, L. A. (2016). The measurement of kidney function. Kidney International, 89(3), 497-499.
17. Bae, E. H., Lim, S. Y., & Han, J. H. (2019). Early identification of chronic kidney disease and associated factors in hospitalized patients: A retrospective observational study. PloS One, 14(3), e0213053.
18. Fotheringham, J., Fogarty, D. G., & El Nahas, M. (2005). Predicting progression of chronic kidney disease. Current Opinion in Nephrology and Hypertension, 14(2), 161-166.