Chronic Kidney Disease Prediction using Multi-Task Learning Algorithm

I. ABSTRACT

Chronic Kidney Disease (CKD), also known as chronic renal disease, has become a serious public health concern with a constant upward trend. Chronic Kidney Disease (CKD) is a disorder that causes a gradual loss of kidney function over time as a result of a variety of diseases. A person can only survive for 18 days without their kidneys, resulting in a high demand for kidney transplants and dialysis. It's critical to have reliable tools for predicting CKD early on. In the prediction of CKD, machine learning technologies are efficient. This paper presents a strategy for predicting CKD status from clinical data that includes data preparation, a missing value management approach, collaborative filtering, and attribute selection. The additional tree classifier and random forest classifier are proven to have the highest accuracy and least bias among the 11 machine learning algorithms studied. The study also takes into account the practical issues of data gathering and emphasizes the need of applying domain knowledge when using machine learning to predict CKD status. This study is largely focused on determining the most appropriate classification method for diagnosing CKD based on the classification report and performance criteria. KNN, Decision Tree Classifier, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier, Stochastic Gradient Boosting, XgBoost, Cat Boost, Extra Trees Classifier are some of the techniques that have undergone empirical testing. When compared to other classification methods, the AdaBoost Classifier, Gradient Boosting Classifier, Stochastic Gradient Boosting, and Extra Trees Classifier produce better results and generate 98.33 percent accuracy.

Keywords - *Chronic kidney disease, Chronic renal disease, Machine learning, Classification algorithms, KNN, Decision Tree Classifier, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier, Stochastic Gradient Boosting, XgBoost, Cat Boost, Extra Trees Classifier.*

II. INTRODUCTION

Kidneys are two bean-shaped organs, each about the size of a fist. They are located just below the rib cage, one on each side of the spine. Every day, the kidneys filter about 120 to 150 quarts of blood to produce about 1 to 2 quarts of urine. The key function of the kidneys is to remove waste products and excess fluid from the body through the urine. The production of urine involves highly complex steps of excretion and reabsorption. This process is necessary to maintain a stable balance of body chemicals. The critical regulation of the body 's salt, potassium and acid content is performed by the kidneys and produces hormones that affect the function of other organs. For example, a hormone produced by the kidneys stimulates red blood cell production, regulates blood pressure and controls calcium metabolism etc.

Chronic Kidney Disease (CKD) is a generic condition for several diseases that affect the kidneys, and it generally means permanent and progressive damage to kidneys, until end-stage renal disease. It affects 12%-14% of people worldwide and its related care costs represent an important percentage of the total health expenditure. According to the Center for Disease Control and Prevention (CDC), Chronic Kidney Disease affects approximately 1 in 7 adults, or an estimated 30 million Americans, consisting in annual national care costing over 32 billion of dollars. Several are the possible causes of onset and rapid evolution of CKD, including diabetes, high blood pressure, or previous episodes in the family history. Prevention and early detection of CKD allow appropriate treatment and are the main factors against the disease, which however, in most of the cases, can only postpone the onset of complete kidney failure.

Chronic kidney disease (CKD) is a major issue worldwide which is a condition characterized by a gradual loss of kidney function over time, 14% of the world population suffer from CKD. Over 2 million people worldwide currently receive treatment with dialysis or a kidney transplant to stay alive, yet this number may only represent 10% of people who need treatment to live. Chronic kidney disease causes more deaths than breast cancer or prostate cancer. The stages of CKD are mainly based on the measured or estimated glomerular filtration rate (eGFR) which is based on creatinine level, gender, race and age. There are five stages of kidney functionality. The function is normal in stage 1 and minimally reduced in stage 2 but the majority of cases are at stage 3.

Data science is concerned with procedures and systems that are used to extract knowledge or insights from large amounts of structured or unstructured data. Today, data science has far-reaching inferences in many fields, both academic and applied research domains like image recognition, machine translation, speech recognition, digital economy on one hand and fields like healthcare, social science, medical informatics etc. Classification is a known method of data mining in the healthcare domain. It is used to predict the target class for each data point. The classification methods include Decision Tree, Random Forest, Support Vector Machine, K-Nearest Neighbour, Naïve Bayes etc. CKD generally develops slowly with few symptoms, and many people cannot discern that they have it until the disease is usually in its last stage. This disease kills a greater number of people each year compared to people with breast or prostate cancer. The prediction of the existence of this disease plays an important role in taking necessary preventive measures.

To predict positive CKD status and the stages of CKD machine learning can be used. Machine Learning grabs a major part of artificial intelligence when it comes to doing predictions from previous data using classification and regression methods. Application of machine learning methods to predict CKD has been explored based on multiple data sets. Among them, the dataset from UCI repository (referred to as UCI dataset hereafter) is identified as a benchmark dataset. Similar to most of the related work, this work considers the mentioned benchmark dataset. When analyzing clinical data related to CKD, if there are instances with missing attributes then the missing values handling method should be determined based on the randomness of the way they were missed. Moreover, the UCI data-set has 400 instances which are a comparatively small number of samples with 25 attributes. In this case, the data set may have redundant (highly correlated) features or the data set does not represent all possibilities. Thereby, this work identifies the limitations in handling missing values when analyzing CKD data, proposes a new method to handle missing values and presents the evaluation of different methods based on UCI dataset. Further, this work also highlights the importance of statistical analysis as well as the domain knowledge of the features when making a prediction based on clinical data related to CKD.

III. LITERATURE SURVEY

IV. METHODOLOGY

The following are the steps followed for the prediction of CKD.

1. **Acquire:** **The database for prediction of CKD is obtained from patient medical reports which are obtained from different laboratories in Tamil Nadu through UCI repository. There are 400 records with 25 various attributes related to kidney disease like age, blood pressure, specific gravity, albumin, sugar, red blood cells, pus cells, bacteria, blood glucose random, blood urea, serum creatinine, sodium, potassium, hemoglobin, packed cell volume, white blood cell count, diabetes mellitus, anemia.**

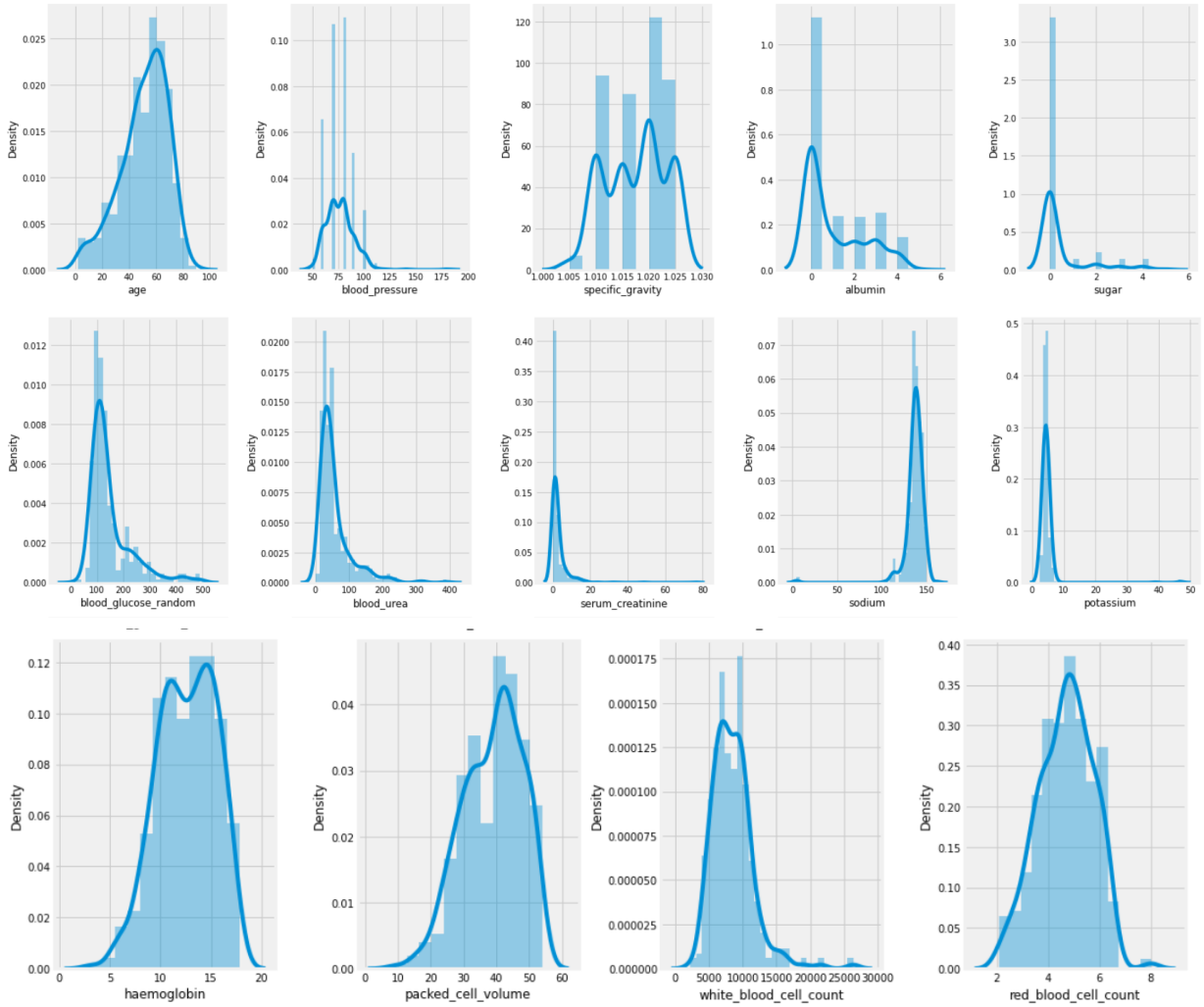
2. **Data preprocessing:** **The CKD dataset class consists of 'ckd' and 'nockd' as target labels. In order to replace strings with numbers, we assigned numbers like '1' and '0' respectively so that the algorithms can work on the data. Similarly, this process is done for the remaining attributes possessing strings. In the CKD dataset, there are many unknown values discovered. Initially, the strings are replaced with integers. Then the missing values in all the columns are filled with medians of that column respectively.**

   In the CKD dataset, there are many unknown values discovered. Initially, the strings are replaced with integers. Then the missing values in all the columns are filled with medians of that column respectively.

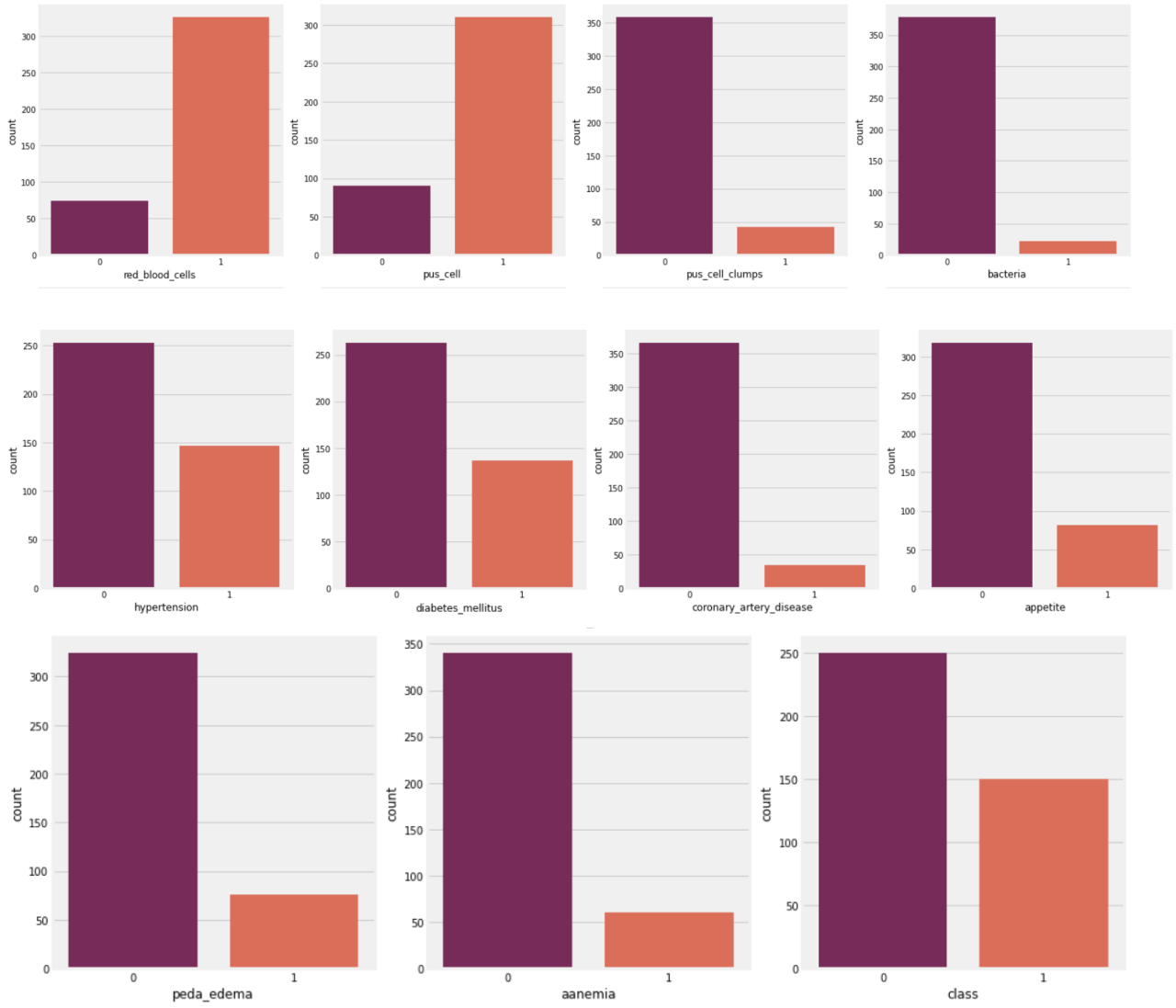
3. **Data Exploration:** **It is performed in the beginning of the data analysis, where a data analyst tries using various visual scrutiny techniques to comprehend what is in a dataset and the traits of the data, rather than through conventional data management systems.**

   (i) <u>Univariate analysis:</u> This function provides access to several approaches for visualizing the univariate or bivariate distribution of data, including subsets of data defined by semantic mapping and faceting across multiple subplots.

DistPlot: Shows the visual representation of the relationship between the top 14 attributes.



Distribution curve: The following graph displays the value distributions of all the attributes. If the data contains a class variable, distributions are conditioned on the class. For discrete values, the graph shows how many times each attribute value appears in the data. For continuous values, the attribute values are displayed as a function graph.



(ii) <u>Bi-variant analysis</u>: This analysis is based on the correlation. It is a mutual relationship between quantities. Correlation can help in predicting one quantity from another. The association between variables can be found using the describe function. The correlation can also be represented using heat maps. The following is the heat map for correlation among all the attributes in the dataset. The above heat map depicts the correlation of all the attributes. The correlation is observed using the scale where the intensity of the color determines the correlation and the value of the correlation.

4. **Feature Selection:** In order to abate the training time and the evaluation time and increase the accuracy of prediction feature selection is done. The model-based ranking is one of the methods where a classifier is fit to each feature and ranks the predictive power. This method selects the most powerful features individually but ignores the predictive power when features are combined. After exploring the CKD dataset using univariate feature selection, the features are obtained whose contribution is almost equal to the contribution of all the attributes together. The top features are 'pcv', 'hemo', 'sc', 'rbcc', 'sg'.

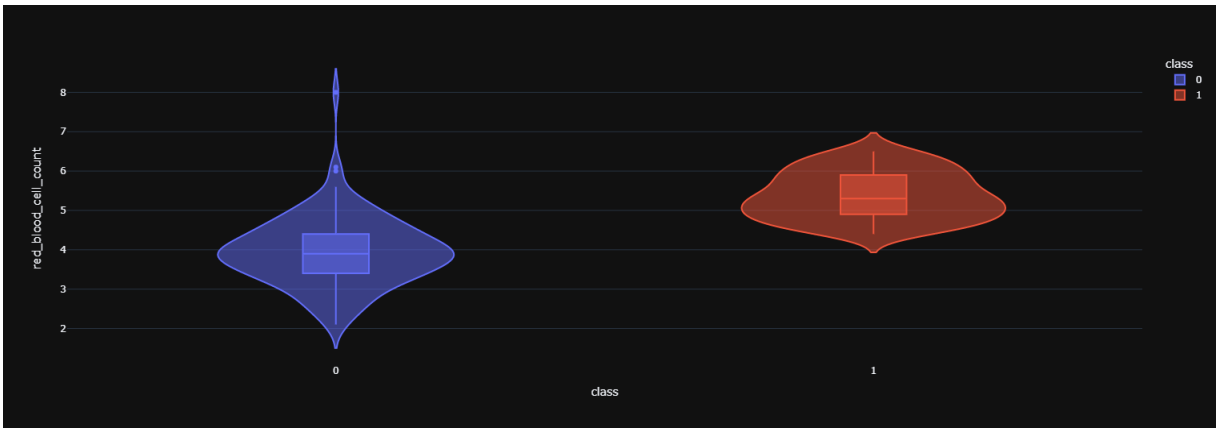
V. Exploratory Data Analysis (EDA):

Exploratory Data Analysis (EDA) is an approach to analyzing datasets to summarize their main characteristics, often with visual methods. EDA is used for seeing what the data can tell us before the modeling task.

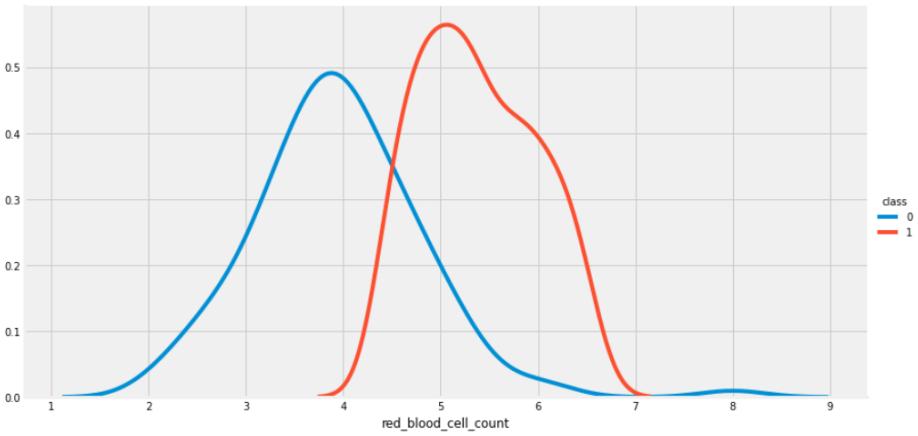
1. **Violin Plot:** It shows the distribution of quantitative data across several levels of one (or more) categorical variables such that those distributions can be compared.

2. **KDE Plot:** A kernel density estimate (KDE) plot is a method for visualizing the distribution of observations in a dataset, analogous to a histogram. KDE represents the data using a continuous probability density curve in one or more dimensions.

A. Red Blood Cell Count:

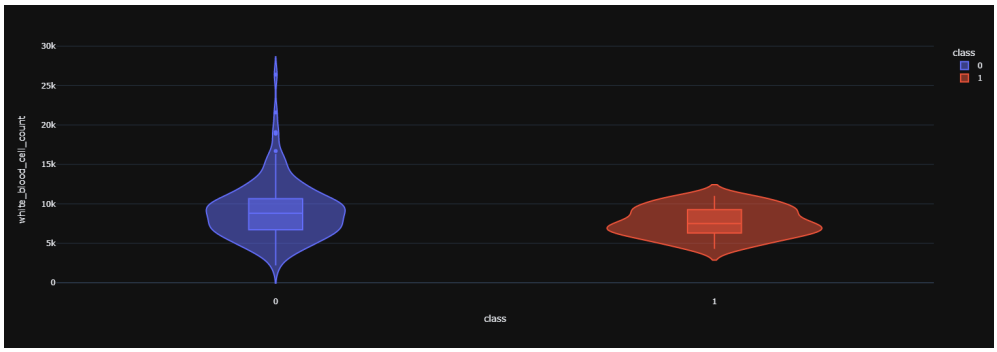


(Violin-Plot)

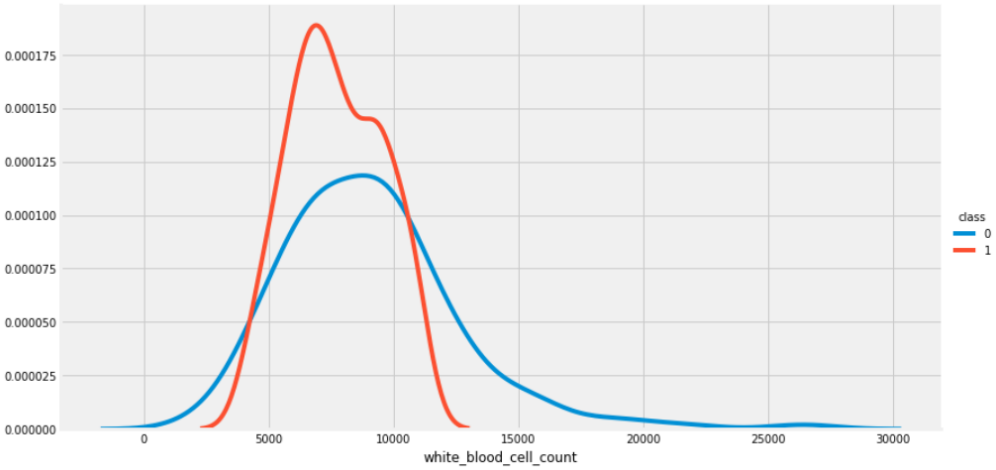


(KDE Plot)

B. White Blood Cell:

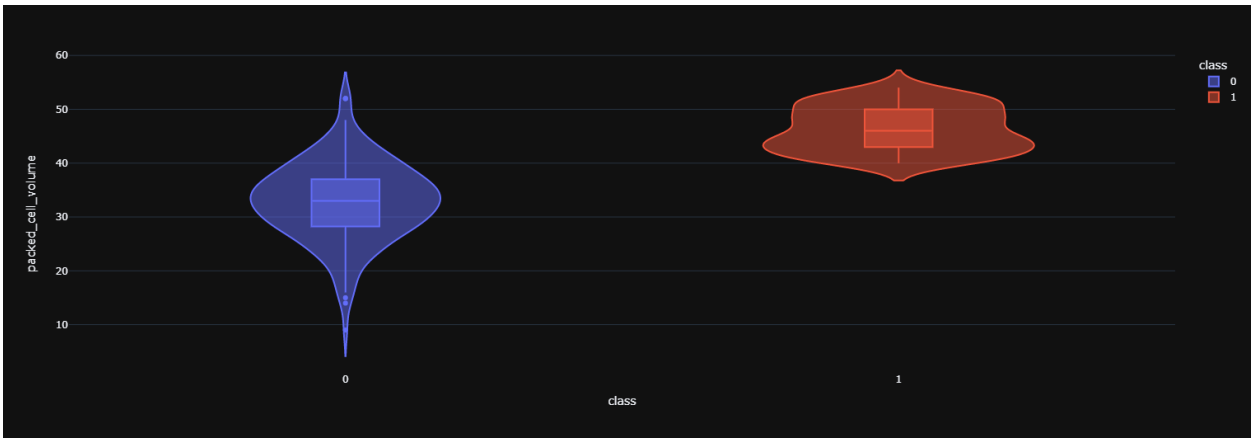


(Violin-Plot)

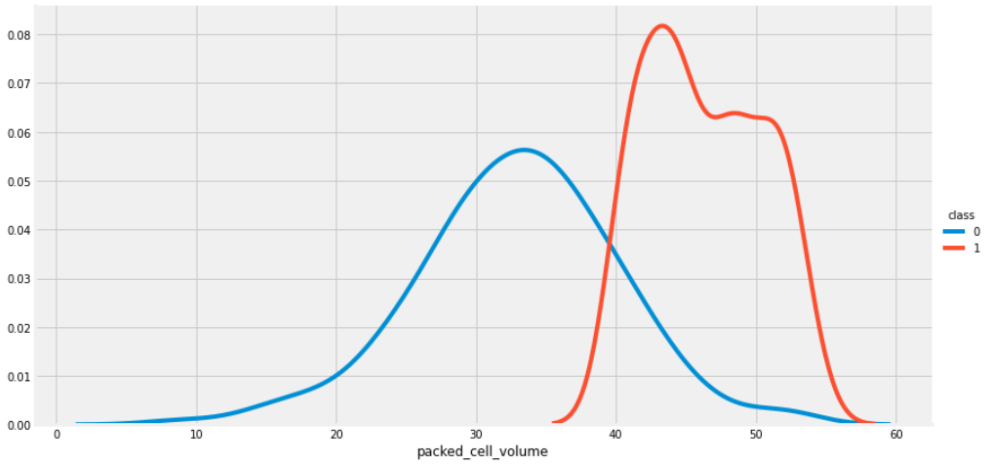


(KDE Plot)

C. Packed Cell Volume:

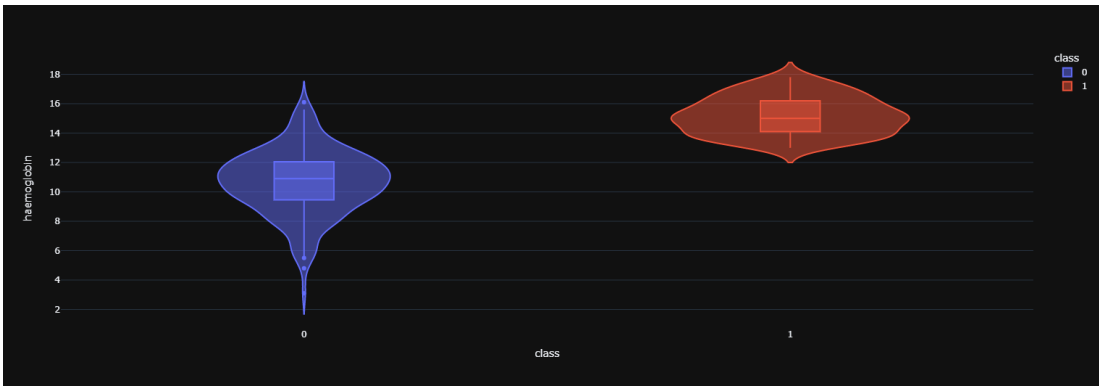


(Violin-Plot)



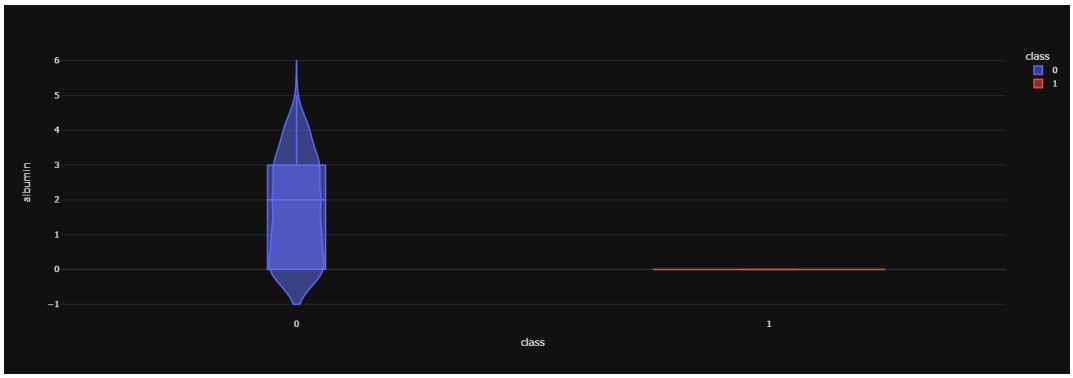
(KDE Plot)

D. Hemoglobin:

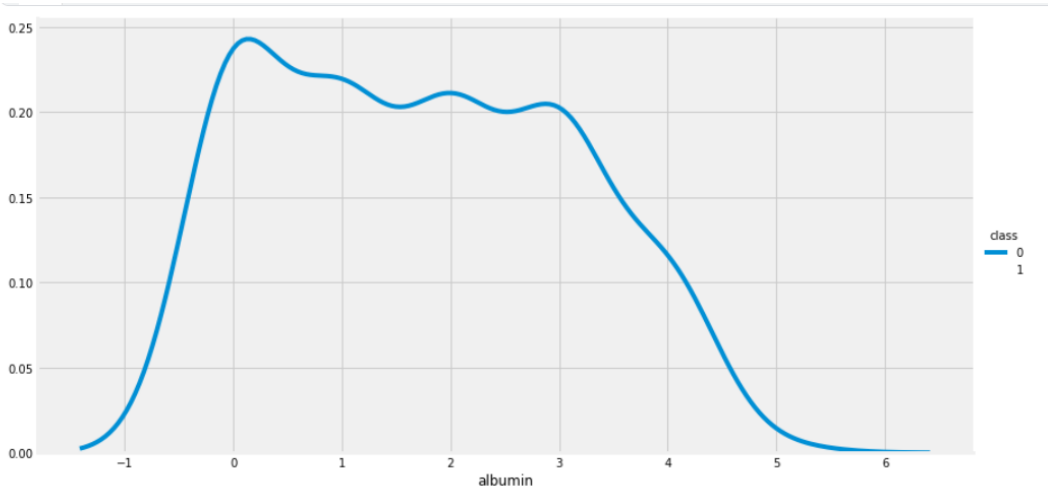


(Violin-Plot)

E. Albumin:

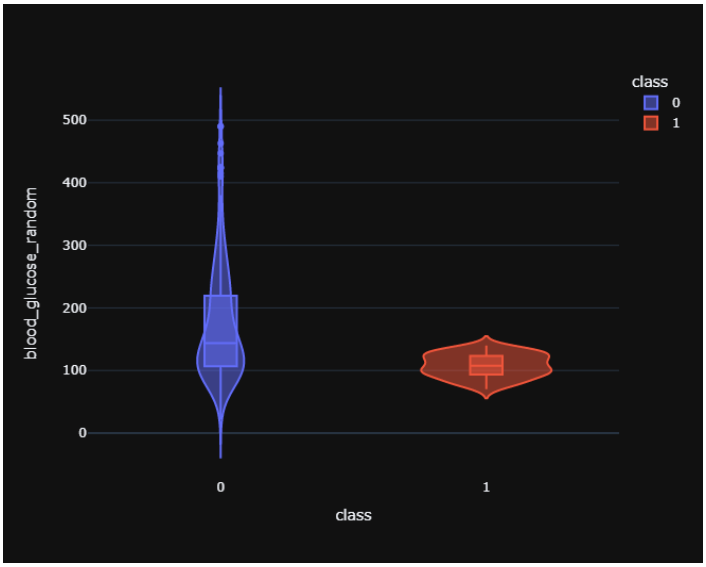


(Violin-Plot)

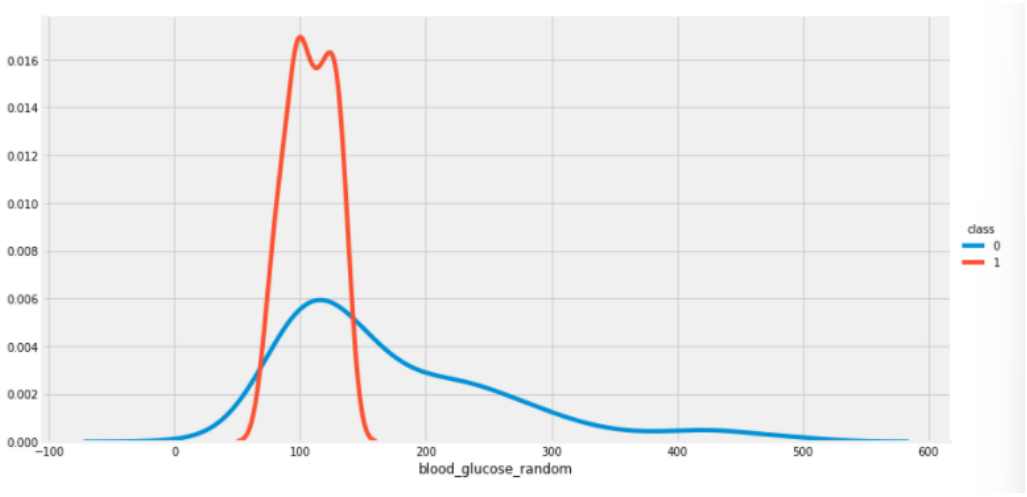


(KDE Plot)

F. Blood Glucose Random:

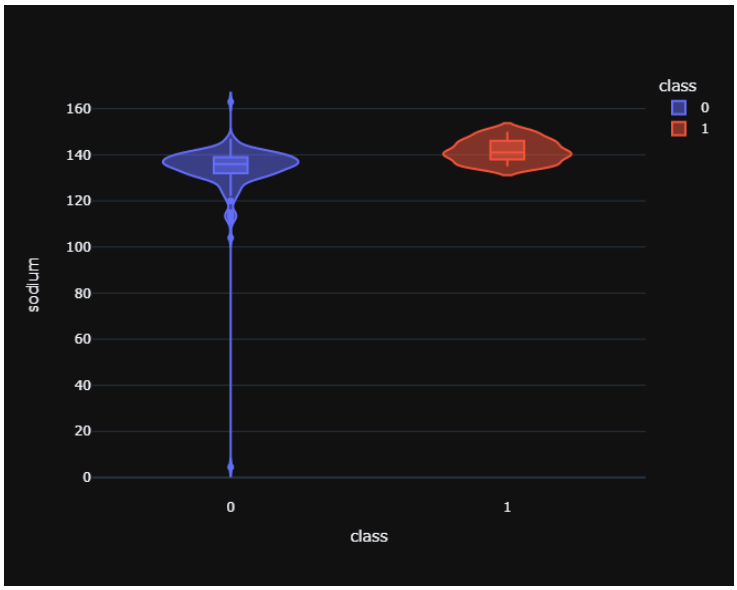


(Violin-Plot)

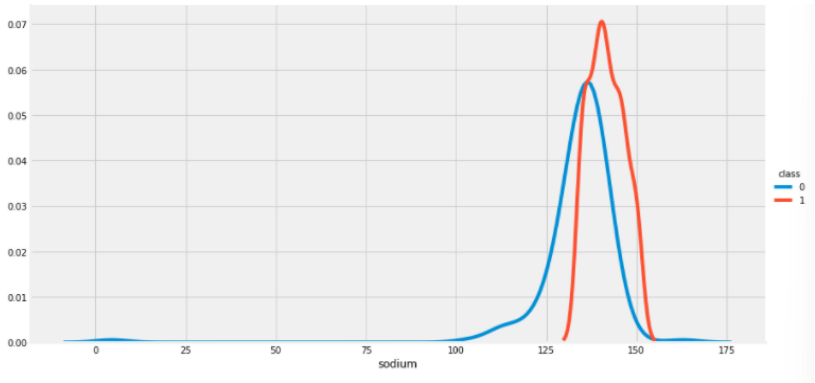


(KDE Plot)

G. Sodium:

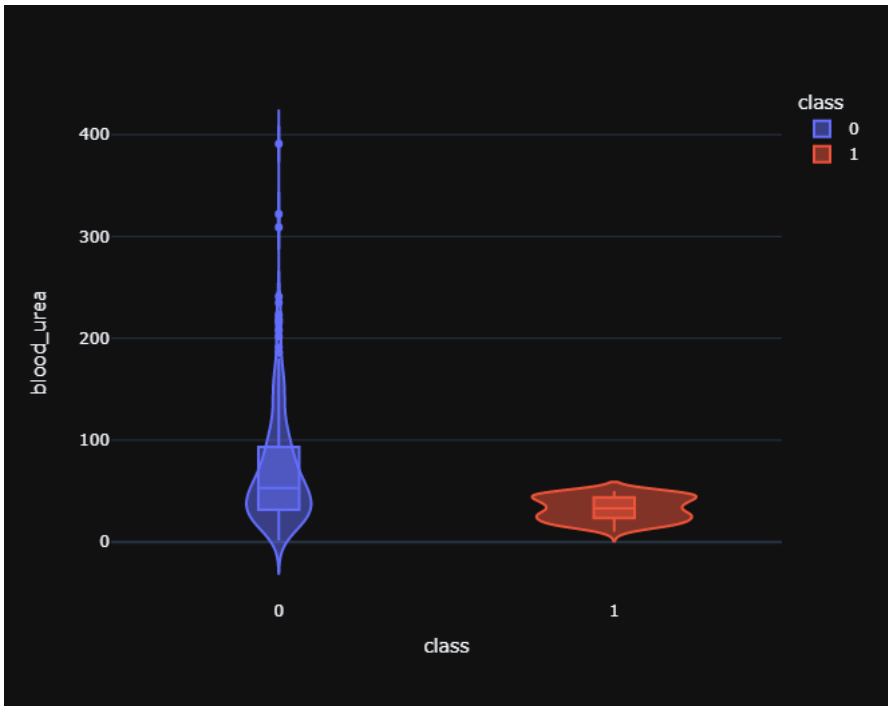


(Violin-Plot)

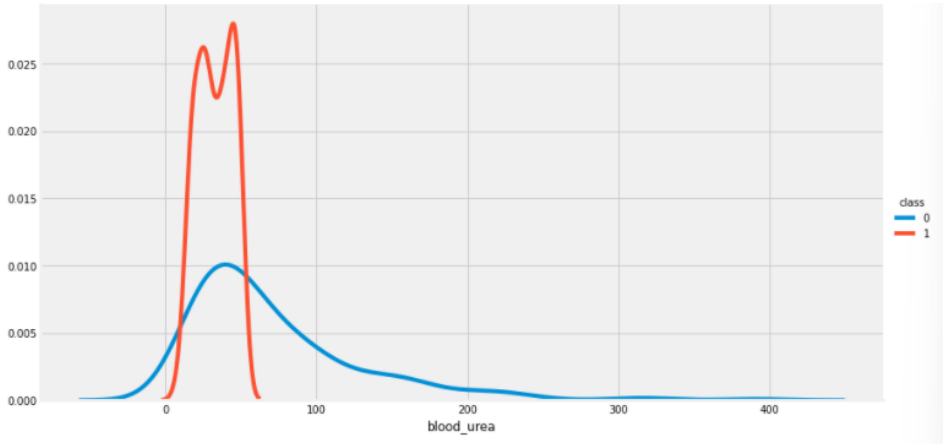


(KDE Plot)

H. Blood Urea:

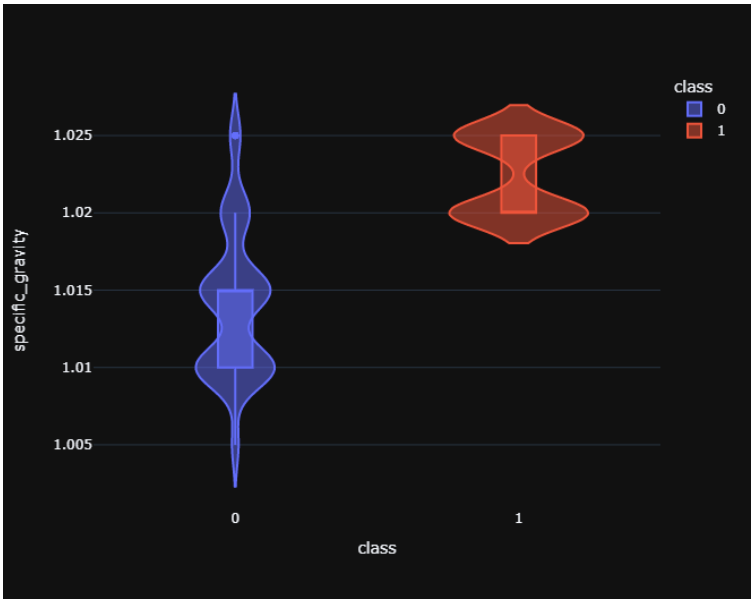


(Violin-Plot)

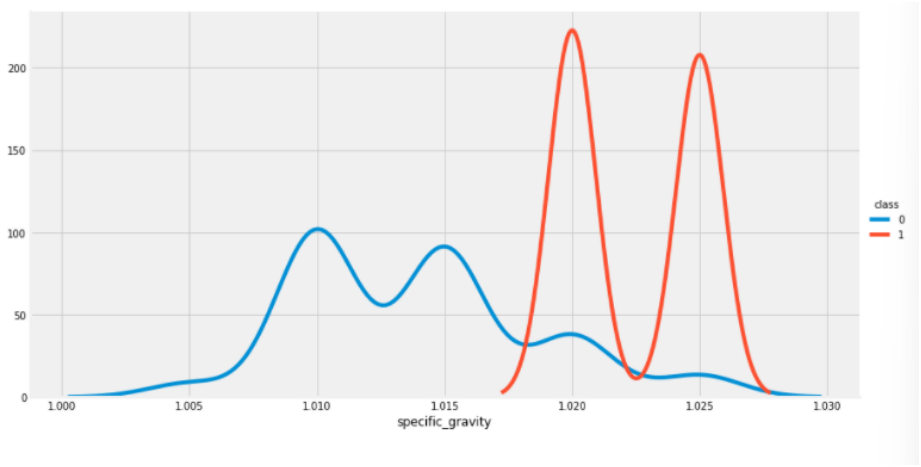


(KDE Plot)

I. Specific Gravity:



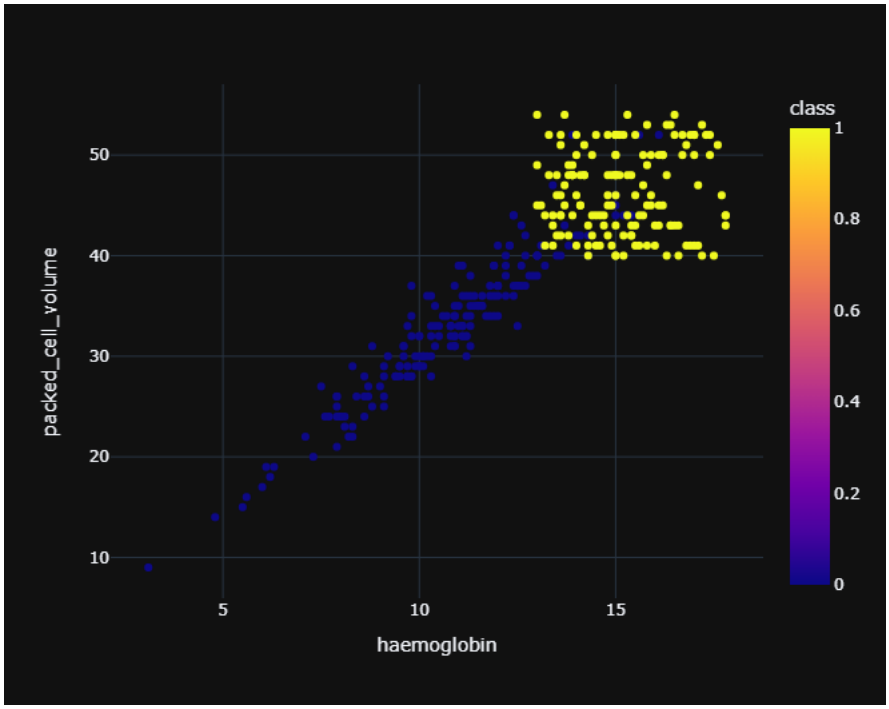
(Violin-Plot)



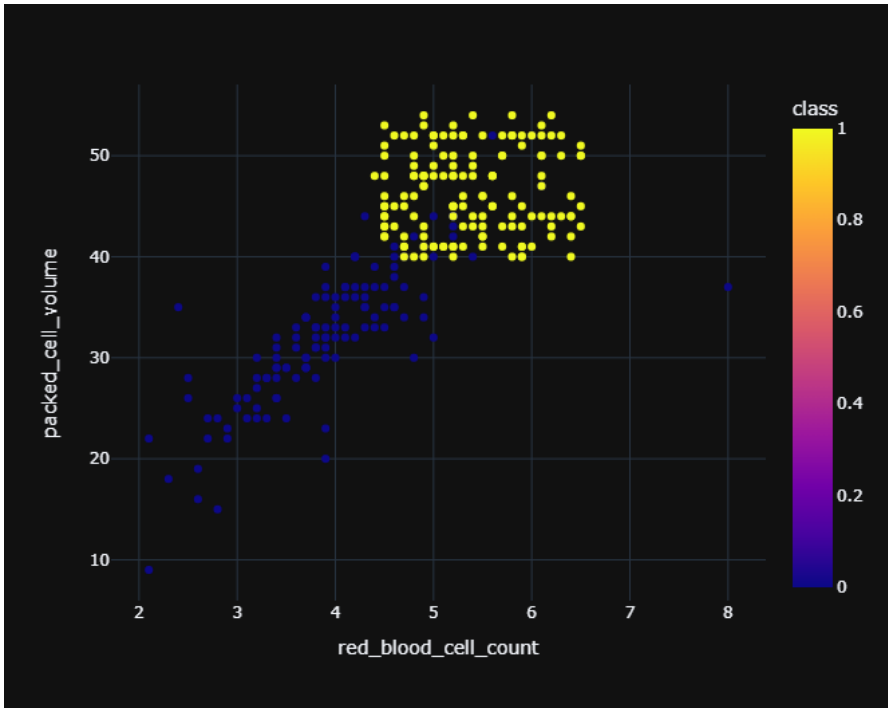
(KDE Plot)

VI. SCATTER: The relationship between x and y can be shown for different subsets of the data using the hue, size, and style parameters. These parameters control what visual semantics are used to identify the different subsets. It is possible to show up to three dimensions independently by using all three semantic types, but this style of plot can be hard to interpret and is often ineffective. Using redundant semantics (i.e. both hue and style for the same variable) can be helpful for making graphics more accessible.

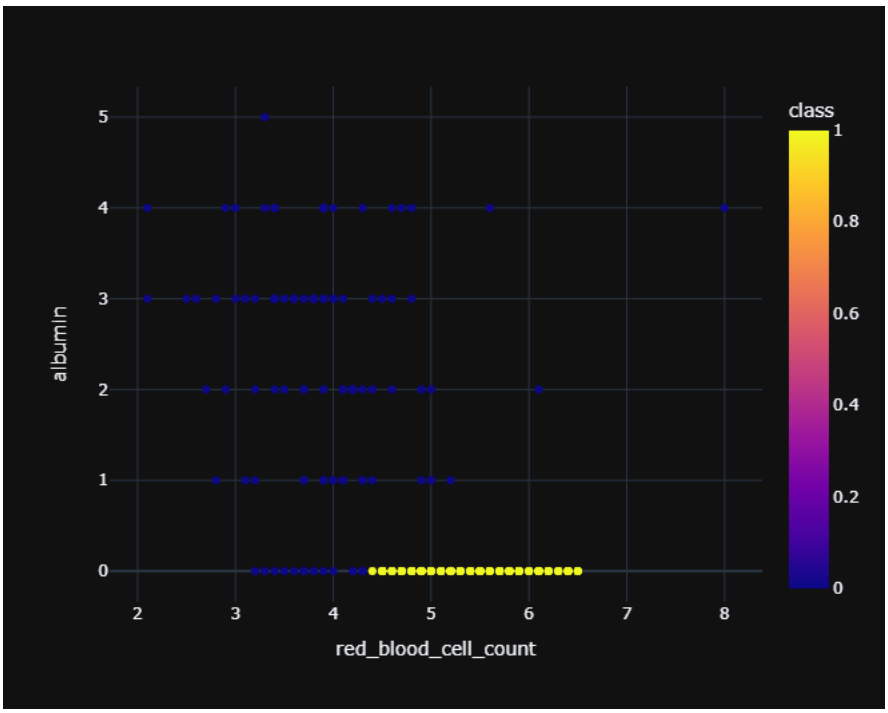
I. Haemoglobin & Packed Cell Volume:



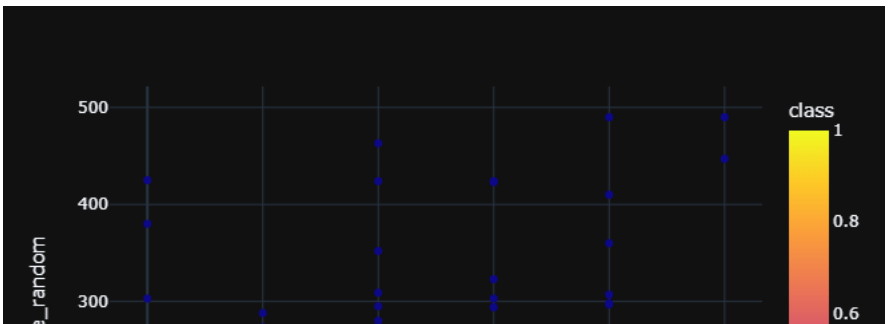
II. Red Blood Cell Count & Packed Cell Volume:



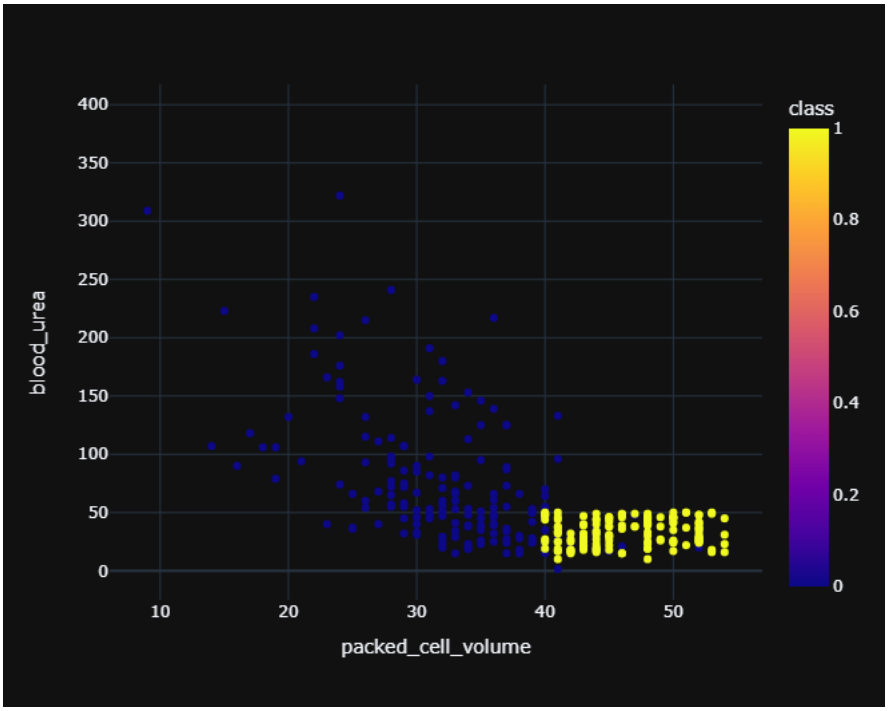
III. Red Blood Cell Count & Albium:



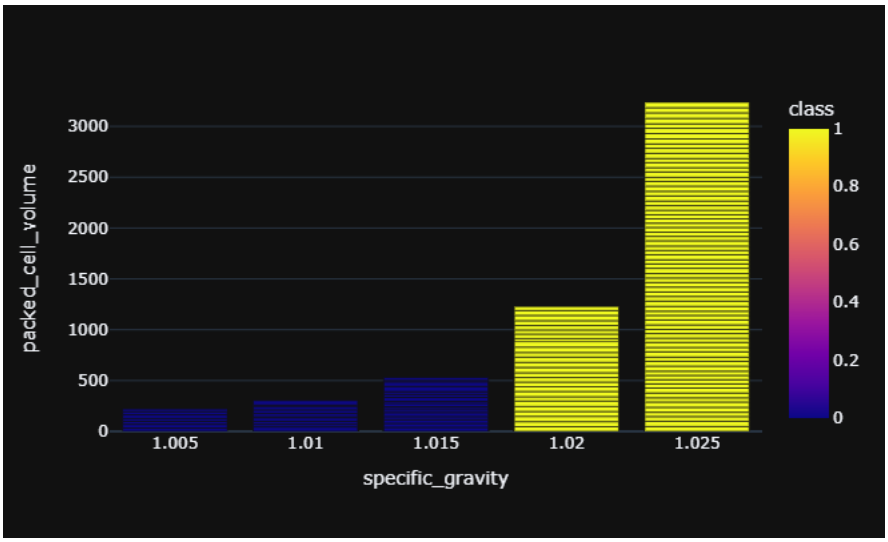
IV. Sugar & Blood Glucose Random:



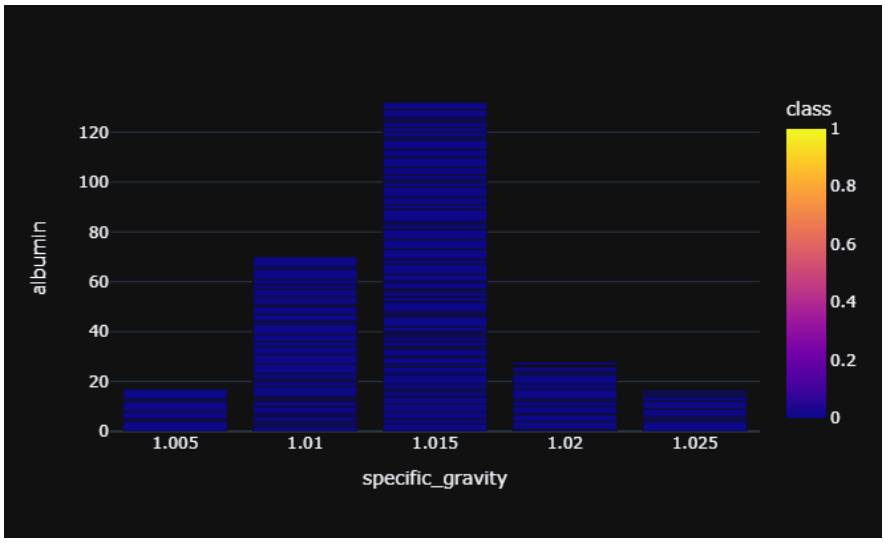
V. Packed Cell Volume & Blood Urea:



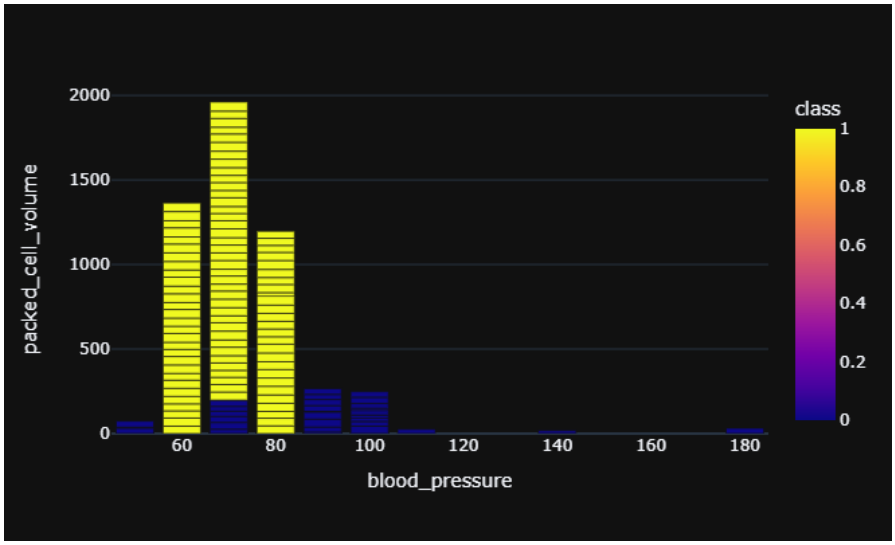
VI. Specific Gravity & Packed Cell Volume:



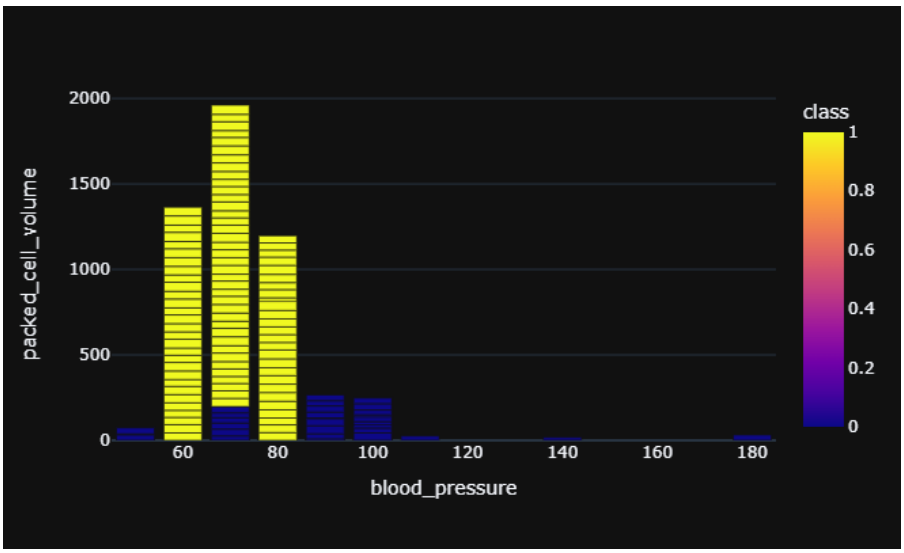
VII. Specific Gravity & Albumin:



VIII. Blood Pressure & Packed cell Volume:



IX. Blood Pressure & Haemoglobin:



VII. **Model selection:** In this paper, Nine classification algorithms are used Extra Tree Classifier, Cat Boost, XGBoost, Stochastic Gradient Boosting, Gradient Boosting Classifier, AdaBoost Classifier, Random forest Classifier, Decision Tree Classifier, KNN.

I. **Extra Tree Classifier:**

Extremely Randomized Trees Classifier(Extra Trees Classifier) is a type of ensemble learning technique which aggregates the results of multiple de-correlated decision trees collected in a "forest" to output it's classification result. In concept, it is very similar to a Random Forest Classifier and only differs from it in the manner of construction of the decision trees in the forest.

Each Decision Tree in the Extra Trees Forest is constructed from the original training sample. Then, at each test node, Each tree is provided with a random sample of k features from the feature-set from which each decision tree must select the best feature to split the data based on some mathematical criteria (typically the Gini Index). This random sample of features leads to the creation of multiple de-correlated decision trees.

To perform feature selection using the above forest structure, during the construction of the forest, for each feature, the normalized total reduction in the mathematical criteria used in the decision of feature of split (Gini Index if the Gini Index is used in the construction of the forest) is computed. This value is called the Gini Importance of the feature. To perform feature selection, each feature is ordered in descending order according to the Gini Importance of each feature and the user selects the top k features according to his/her choice.

```
Training Accuracy of Extra Trees Classifier is 1.0
Test Accuracy of Extra Trees Classifier is 0.975

Confusion Matrix :-
[[72  0]
 [ 3 45]]

Classification Report :-
               precision    recall  f1-score   support

           0       0.96      1.00      0.98        72
           1       1.00      0.94      0.97        48

    accuracy                           0.97       120
   macro avg       0.98      0.97      0.97       120
weighted avg       0.98      0.97      0.97       120
```

II. Cat Boost:

Catboost can be used for solving problems, such as regression, classification, multi-class classification and ranking. Modes differ by the objective function that we are trying to minimize during gradient descend. Moreover, Catboost has pre-build metrics to measure the accuracy of the model.

```
Training Accuracy of Extra Trees Classifier is 1.0
Test Accuracy of Extra Trees Classifier is 0.975

Confusion Matrix :-
[[72  0]
 [ 3 45]]

Classification Report :-
               precision    recall  f1-score   support

           0       0.96      1.00      0.98        72
           1       1.00      0.94      0.97        48

    accuracy                           0.97       120
   macro avg       0.98      0.97      0.97       120
weighted avg       0.98      0.97      0.97       120
```

III. XGBoost Classifier:

XGBoost is an implementation of Gradient Boosted decision trees. This library was written in C++. It is a type of Software library that was designed basically to improve speed and model performance. It has recently been dominating in applied machine learning. XGBoost models majorly dominate in many Kaggle Competitions.

In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. Weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.

```
Training Accuracy of XgBoost is 1.0
Test Accuracy of XgBoost is 0.95

Confusion Matrix :-
[[72  0]
 [ 6 42]]

Classification Report :-
              precision    recall  f1-score   support

           0       0.92      1.00      0.96        72
           1       1.00      0.88      0.93        48

    accuracy                           0.95       120
   macro avg       0.96      0.94      0.95       120
weighted avg       0.95      0.95      0.95       120
```

IV. Stochastic Gradient Boosting:

The word ‘stochastic‘ means a system or a process that is linked with a random probability. Hence, in Stochastic Gradient Descent, a few samples are selected randomly instead of the whole data set for each iteration. In Gradient Descent, there is a term called “batch” which denotes the total number of samples from a dataset that is used for calculating the gradient for each iteration. In typical Gradient Descent optimization, like Batch Gradient Descent, the batch is taken to be the whole dataset. Although, using the whole dataset is really useful for getting to the minima in a less

noisy and less random manner, but the problem arises when our datasets get big.
Suppose, you have a million samples in your dataset, so if you use a typical Gradient Descent optimization technique, you will have to use all of the one million samples for completing one iteration while performing the Gradient Descent, and it has to be done for every iteration until the minima are reached. Hence, it becomes computationally very expensive to perform. This problem is solved by Stochastic Gradient Descent. In SGD, it uses only a single sample, i.e., a batch size of one, to perform each iteration. The sample is randomly shuffled and selected for performing the iteration.

```
Training Accuracy of Stochastic Gradient Boosting is 1.0
Test Accuracy of Stochastic Gradient Boosting is 0.9666666666666667

Confusion Matrix :-
[[71  1]
 [ 3 45]]

Classification Report :-
              precision    recall  f1-score   support

           0       0.96      0.99      0.97        72
           1       0.98      0.94      0.96        48

    accuracy                           0.97       120
   macro avg       0.97      0.96      0.97       120
weighted avg       0.97      0.97      0.97       120
```

V. Gradient Boosting Classifier:

Gradient Boosting is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees).

```
Training Accuracy of Gradient Boosting Classifier is 1.0
Test Accuracy of Gradient Boosting Classifier is 0.975

Confusion Matrix :-
[[72  0]
 [ 3 45]]

Classification Report :-
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        72
           1       1.00      0.94      0.97        48

    accuracy                           0.97       120
   macro avg       0.98      0.97      0.97       120
weighted avg       0.98      0.97      0.97       120
```

VI. AdaBoost Classifier:

AdaBoost models belong to a class of ensemble machine learning models. From the literal meaning of the word 'ensemble', we can easily have much better intuition of how this model works. Ensemble models take the onus of combining different models and later produce an advanced/more accurate meta model. This meta model has comparatively high accuracy in terms of prediction as compared to their corresponding counterparts.

AdaBoost algorithm falls under ensemble boosting techniques, as discussed it combines multiple models to produce more accurate results and this is done in two phases:

1. Multiple weak learners are allowed to learn on training data.
2. Combining these models to generate a meta-model, this meta-model aims to resolve the errors as performed by the individual weak learners.

```
Training Accuracy of Ada Boost Classifier is 1.0
Test Accuracy of Ada Boost Classifier is 0.925

Confusion Matrix :-
[[70  2]
 [ 7 41]]

Classification Report :-
               precision    recall  f1-score   support

           0       0.91      0.97      0.94        72
           1       0.95      0.85      0.90        48

    accuracy                           0.93       120
   macro avg       0.93      0.91      0.92       120
weighted avg       0.93      0.93      0.92       120
```

VII. Random forest Classifier:

The Random forest or Random Decision Forest is a supervised Machine learning algorithm used for classification, regression, and other tasks using decision trees.
The Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then It collects the votes from different decision trees to decide the final prediction.

```
Training Accuracy of Random Forest Classifier is 1.0
Test Accuracy of Random Forest Classifier is 0.9666666666666667

Confusion Matrix :-
[[72  0]
 [ 4 44]]

Classification Report :-
               precision    recall  f1-score   support

           0       0.95      1.00      0.97        72
           1       1.00      0.92      0.96        48

    accuracy                           0.97       120
   macro avg       0.97      0.96      0.96       120
weighted avg       0.97      0.97      0.97       120
```

VIII. Decision Tree Classifier:

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

```
Training Accuracy of Decision Tree Classifier is 1.0
Test Accuracy of Decision Tree Classifier is 0.95

Confusion Matrix :-
[[72  0]
 [ 6 42]]

Classification Report :-
              precision    recall  f1-score   support

           0       0.92      1.00      0.96        72
           1       1.00      0.88      0.93        48

    accuracy                           0.95       120
   macro avg       0.96      0.94      0.95       120
weighted avg       0.95      0.95      0.95       120
```

IX. KNN:

K-Nearest Neighbours is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data).

We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute.

```
Training Accuracy of KNN is 0.8
Test Accuracy of KNN is 0.675

Confusion Matrix :-
[[51 21]
 [18 30]]

Classification Report :-
               precision    recall  f1-score   support

           0       0.74      0.71      0.72        72
           1       0.59      0.62      0.61        48

    accuracy                           0.68       120
   macro avg       0.66      0.67      0.66       120
weighted avg       0.68      0.68      0.68       120
```

IV. RESULTS AND DISCUSSION

This paper is implemented in Kaggle distributions. It aims at simplifying package management and deployment. The comparative analysis of classification algorithms is done based on the performance factors of classification accuracy, precision and f1 score.

The top five features contribute to 90% accuracy and top 8 attributes contribute equally to 26 attributes contribution in the dataset. In this experiment, the classification algorithm is applied to all the 26 attributes and the results are based on the following terms:
1. True positives (TP): These are the cases in which CKD is predicted (they have the disease).

2. True négatives (TN): If predicted no-ckd, and they don't have the disease.

3. False positives (FP): If predicted CKD, but they don't have the disease

4. False négatives (FN): If predicted no-ckd, but they do have the disease

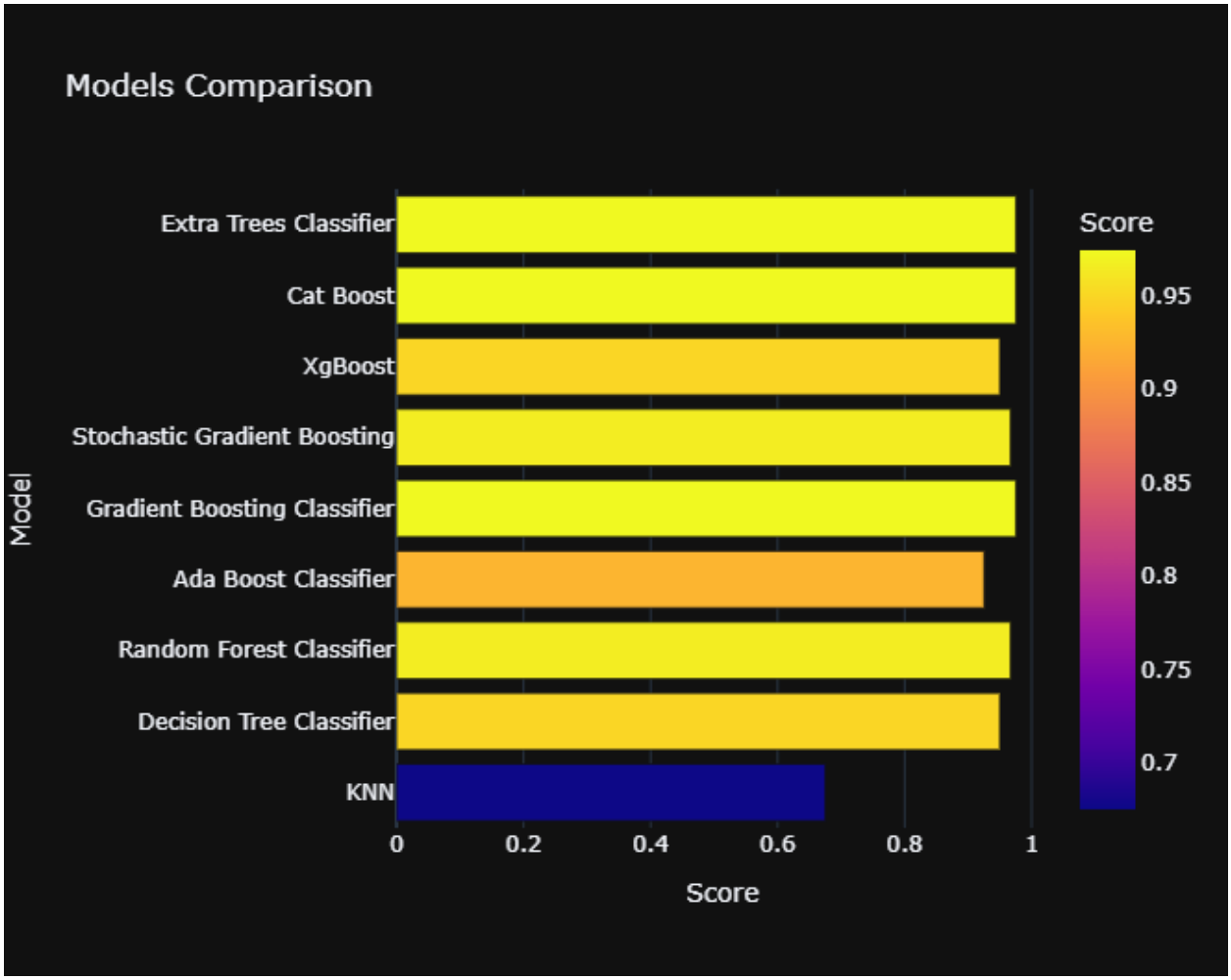
I. **Confusion Matrix: is a summary of prediction results on a classification problem. The following table-1 shows the confusion matrix obtained for each classification model.**

	Model	Score
4	Gradient Boosting Classifier	0.975000
7	Cat Boost	0.975000
8	Extra Trees Classifier	0.975000
2	Random Forest Classifier	0.966667
5	Stochastic Gradient Boosting	0.966667
1	Decision Tree Classifier	0.950000
6	XgBoost	0.950000
3	Ada Boost Classifier	0.925000
0	KNN	0.675000

2. Accuracy: Accuracy is decomposed as the fraction of unerringly classified records and the total number of records present in the dataset.

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

The above graph shows our acquired results for the classification algorithms applied. It is observed that the highest accuracy was obtained for Gradient Boosting Classifier(97.50%), Cat Boost (97.55%), Extra Tree Classifier(97.50%) compared to Ada Boost Classifier(92.5%) and KNN(67.50%).



V. CONCLUSION

The objective of this analysis is to observe classification algorithms to analyze and predict CKD. We have compared the performance of five classifiers in the prognosis of CKD. The experimental results of our proposed method have demonstrated that RF and XGB have produced superior prediction performance in terms of classification accuracy for our considered dataset. For the future, we are working on enhancing the performance of prediction system accuracy by ensemble diffèrent classifier algorithms.