Predicting Chronic Kidney Disease with Machine Learning: A Python Approach

Amuktha Ramya Sri Vemireddy

Shettygari Susrutha

Jahnavi reddy gujju

Grand Valley State University

CIS 635 01 - TR - Knowledge Discovery and Data Mining

Professor: Kamrul Hassan

INTRODUCTION

Chronic kidney disease is a progressive loss of kidney function over a period of months or years. Our kidneys work to keep us healthy by cleaning wastes from our blood with millions of tiny filters, called nephrons. If these nephrons are damaged, they begin to shut down. Eventually, there are not enough left to filter our blood well enough to keep us healthy and we begin to feel the symptoms of CKD.

About 1.3 million people die from kidney disease each year, with an additional 1.4 million deaths from cardiovascular disease that are attributed to impaired kidney function.

CKD is increasing in prevalence – and at an alarming rate. CKD deaths increased by 41.5% from 1990 to 2020, rising from the 17th leading cause of death to the 10th. Now, it is expected that CKD will climb to the fifth leading cause of death globally by the year 2040.

The early detection of failing kidney function can be lifesaving, because it allows CKD to be treated through medications, diet, and lifestyle changes rather than dialysis or a kidney transplant, which are economically inaccessible for most people around the world. These treatments are known as renal replacement therapies (RRT) because they attempt to "replace" the normal functioning of the kidneys.

In this project, we explored the utility of various machine learning models in aiding the early diagnosis of chronic kidney disease (CKD). By leveraging machine learning algorithms such as Random Forest Classifier, Gradient Boosting Classifier, Decision Tree Classifier, XG Boost, and K-Nearest Neighbours (KNN), we aimed to develop predictive models that could assist healthcare providers in identifying patterns and making accurate predictions related to CKD. These models hold promise in revolutionizing the early diagnosis and management of CKD, ensuring timely interventions, and improving outcomes for individuals at risk of kidney dysfunction

BUSINESS PURPOSE OF OUR PROJECT

By addressing the following key objectives, our project aims to make a meaningful impact on CKD prevention, management, and research, ultimately improving patient

outcomes and contributing to the overall well-being of individuals affected by this chronic condition.

1. To enhance early detection and intervention for individuals at risk of developing chronic kidney disease (CKD). Early identification allows for timely medical interventions, leading to more effective treatments and potentially reducing healthcare costs associated with advanced-stage CKD management.
2. To improve resource allocation in healthcare settings by efficiently identifying and prioritizing patients at higher risk of CKD progression. This optimization of resources can lead to enhanced patient care, reduced wait times, and improved overall healthcare system performance.
3. Providing healthcare professionals with a reliable decision support tool specifically tailored for CKD management. The predictive model assists clinicians in developing personalized care plans, monitoring disease progression, and making informed treatment decisions.
4. Contributing to public health initiatives by identifying trends and patterns in CKD risk factors. This information can inform preventive strategies, public health campaigns, and policy interventions aimed at reducing the incidence and burden of CKD on a broader scale.
5. Fostering collaboration and advancements in CKD research by providing valuable insights and data-driven analysis. Collaboration with data scientists, researchers, and healthcare experts can lead to innovations in predictive analytics, disease management strategies, and patient outcomes.

DATA PREPERATION

We obtained a comprehensive dataset from the Kaggle website that encompasses several pertinent features associated with chronic kidney disease. The dataset consists of 400 rows and 25 features, providing ample data to develop a decent predictive model. The dataset features are explained below:

1. Age: The age of the patient in years.
2. Blood Pressure: Systolic and diastolic blood pressure measurements in mmHg.
3. Specific Gravity: The specific gravity of urine, indicating urine concentration and kidney function.

4. Albumin: The presence of albumin in urine, which can indicate kidney damage or disease.
5. Sugar: The presence of sugar in urine, potentially indicating diabetes or other metabolic conditions.
6. Red Blood Cells: The presence or count of red blood cells in urine, related to kidney function and health.
7. Pus Cell: The presence of pus cells in urine, which may suggest infection or inflammation in the urinary tract.
8. Pus Cell Clumps: Clumps of pus cells in urine, also indicative of urinary tract issues.
9. Bacteria: The presence of bacteria in urine, often associated with urinary tract infections.
10. Blood Glucose Random: Random blood glucose levels, relevant for diabetes diagnosis and management.
11. Blood Urea: Blood urea nitrogen (BUN) levels, indicating kidney function and hydration status.
12. Serum Creatinine: Serum creatinine levels, another marker of kidney function.
13. Sodium: Sodium levels in the blood, essential for fluid balance and kidney health.
14. Potassium: Potassium levels in the blood, crucial for heart and muscle function.
15. Haemoglobin: Haemoglobin levels in the blood, related to oxygen transport and anaemia.
16. Packed Cell Volume: The volume percentage of red blood cells in the blood, related to blood oxygenation and hydration.
17. White Blood Cell Count: The count of white blood cells in the blood, relevant for infection and inflammation.
18. Red Blood Cell Count: The count of red blood cells in the blood, related to oxygen transport and anaemia.
19. Hypertension: Binary variable indicating the presence of high blood pressure.
20. Diabetes Mellitus: Binary variable indicating the presence of diabetes.
21. Coronary Artery Disease: Binary variable indicating the presence of coronary artery disease.
22. Appetite: Subjective assessment of appetite as good, poor, or very poor.
23. Pedal Oedema: Binary variable indicating the presence of pedal oedema (swelling of the feet and ankles).
    Anaemia: Binary variable indicating the presence of anaemia.
24. Class: The target variable indicating the presence or absence of chronic kidney disease.

EXPLORATORY DATA ANALYSIS

Upon acquiring the dataset, we conducted a comprehensive exploratory data analysis (EDA) to gain a deep understanding of the data and uncover any nuances it may contain. Our goal was to establish a solid groundwork for the subsequent analytical steps, enabling us to make informed decisions about the data and ultimately construct a precise and impactful predictive model.

Handling Missing Values:

Checked for missing values and found these null values in categorical and numerical columns.

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

```
age                       9
blood_pressure           12
specific_gravity         47
albumin                  46
sugar                    49
blood_glucose_random     44
blood_urea               19
serum_creatinine         17
sodium                   87
potassium                88
haemoglobin              52
packed_cell_volume       71
white_blood_cell_count  106
red_blood_cell_count    131
dtype: int64
```

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

```
red_blood_cells          152
pus_cell                  65
pus_cell_clumps            4
bacteria                   4
hypertension               2
diabetes_mellitus          2
coronary_artery_disease    2
appetite                   1
peda_edema                 1
aanemia                    1
class                      0
dtype: int64
```

Filling Null values:

We will address null values using two distinct methods. For instances where null values are prevalent, we will employ random sampling. Conversely, for instances where null values are less common, we will utilize mean/mode sampling.

```
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)
```

```
[ ] # filling num_cols null values using random sampling method

    for col in num_cols:
        random_value_imputation(col)
```

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

```
age                     0
blood_pressure          0
specific_gravity        0
albumin                 0
sugar                   0
blood_glucose_random    0
blood_urea              0
serum_creatinine        0
sodium                  0
potassium               0
haemoglobin             0
packed_cell_volume      0
white_blood_cell_count  0
red_blood_cell_count    0
dtype: int64
```

Filling "red_blood_cells" and "pus_cell" using random sampling method and rest of cat_cols using mode imputation

```
random_value_imputation('red_blood_cells')
random_value_imputation('pus_cell')

for col in cat_cols:
    impute_mode(col)
```

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

```
red_blood_cells            0
pus_cell                   0
pus_cell_clumps            0
bacteria                   0
hypertension               0
diabetes_mellitus          0
coronary_artery_disease    0
appetite                   0
peda_edema                 0
aanemia                    0
class                      0
dtype: int64
```

Data Type Transformation:

This process is crucial to ensure all variables are in correct formats suitable for analysis and modelling. This is necessary in optimizing compatibility of the data to machine learning algorithms employed for CKD prediction. The features 'packed_cell_volume', 'white_blood_cell_count' and 'red_blood_cell_count' are in object type. We need to convert them to numerical data type.

```
# converting necessary columns to numerical type

df['packed_cell_volume'] = pd.to_numeric(df['packed_cell_volume'], errors='coerce')
df['white_blood_cell_count'] = pd.to_numeric(df['white_blood_cell_count'], errors='coerce')
df['red_blood_cell_count'] = pd.to_numeric(df['red_blood_cell_count'], errors='coerce')
```

Data Inconsistencies:

In this phase, incorrect values were rectified across specific columns. Entries like '\tno', '\tyes', and 'yes' in 'diabetes_mellitus' were standardized to 'no' and 'yes'. Similarly, '\tno' in 'coronary_artery_disease' became 'no' for consistency. In 'class', 'ckd\t' was adjusted to 'ckd', and 'notckd' was corrected to 'not ckd'. These changes ensure uniformity and accuracy in the dataset. Additionally, 'class' values were encoded numerically (0 for 'ckd' and 1 for 'not ckd') and converted to numeric data types, enhancing data quality for subsequent analyses.

```
[21] # replace incorrect values

     df['diabetes_mellitus'].replace(to_replace = {'\tno':'no','\tyes':'yes',' yes':'yes'},inplace=True)

     df['coronary_artery_disease'] = df['coronary_artery_disease'].replace(to_replace = '\tno', value='no')

     df['class'] = df['class'].replace(to_replace = {'ckd\t': 'ckd', 'notckd': 'not ckd'})
```

```
[22] df['class'] = df['class'].map({'ckd': 0, 'not ckd': 1})
     df['class'] = pd.to_numeric(df['class'], errors='coerce')
```

```
[23] cols = ['diabetes_mellitus', 'coronary_artery_disease', 'class']

     for col in cols:
         print(f"{col} has {df[col].unique()} values\n")
```

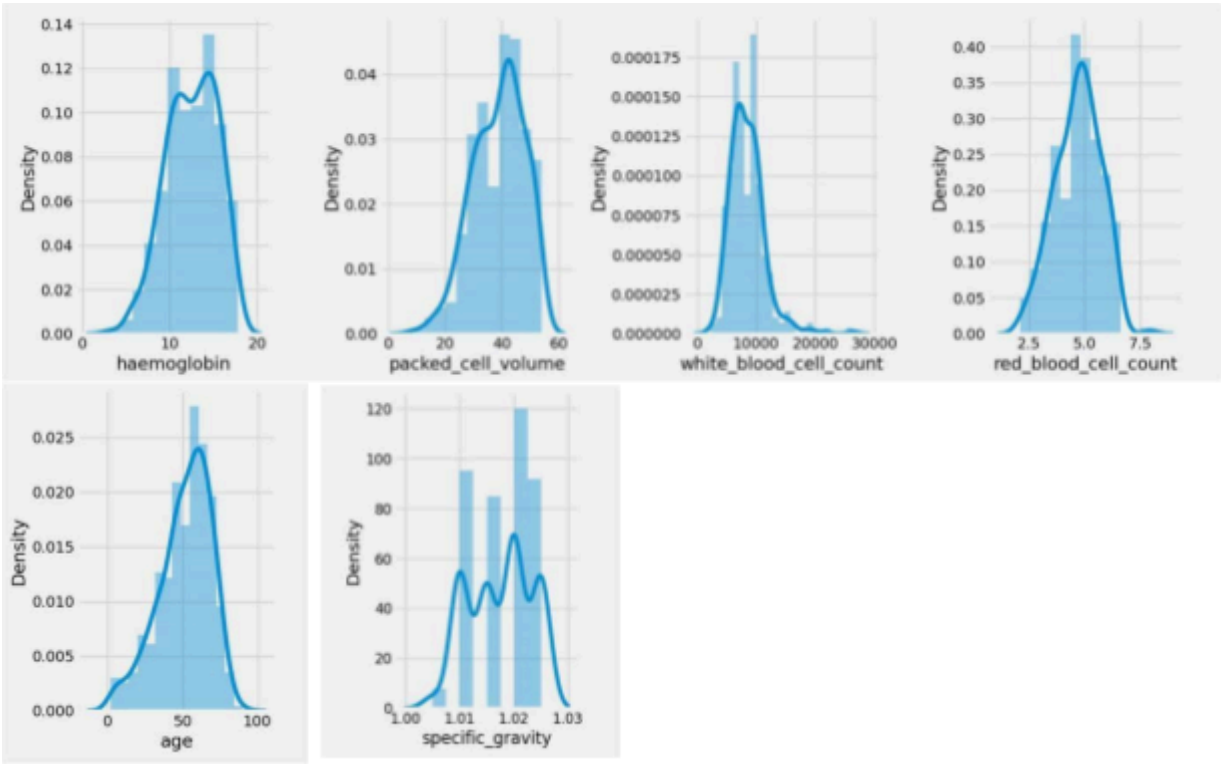
```
diabetes_mellitus has ['yes' 'no' nan] values

coronary_artery_disease has ['no' 'yes' nan] values

class has [0 1] values
```

DATA VISUALIZATION

Data visualization is a vital tool for transforming data into actionable insights, supporting decision-making, enhancing understanding, and driving innovation across various fields and industries.

The distribution of numerical features in the dataset provides valuable insights into the characteristics and patterns of physiological measurements.

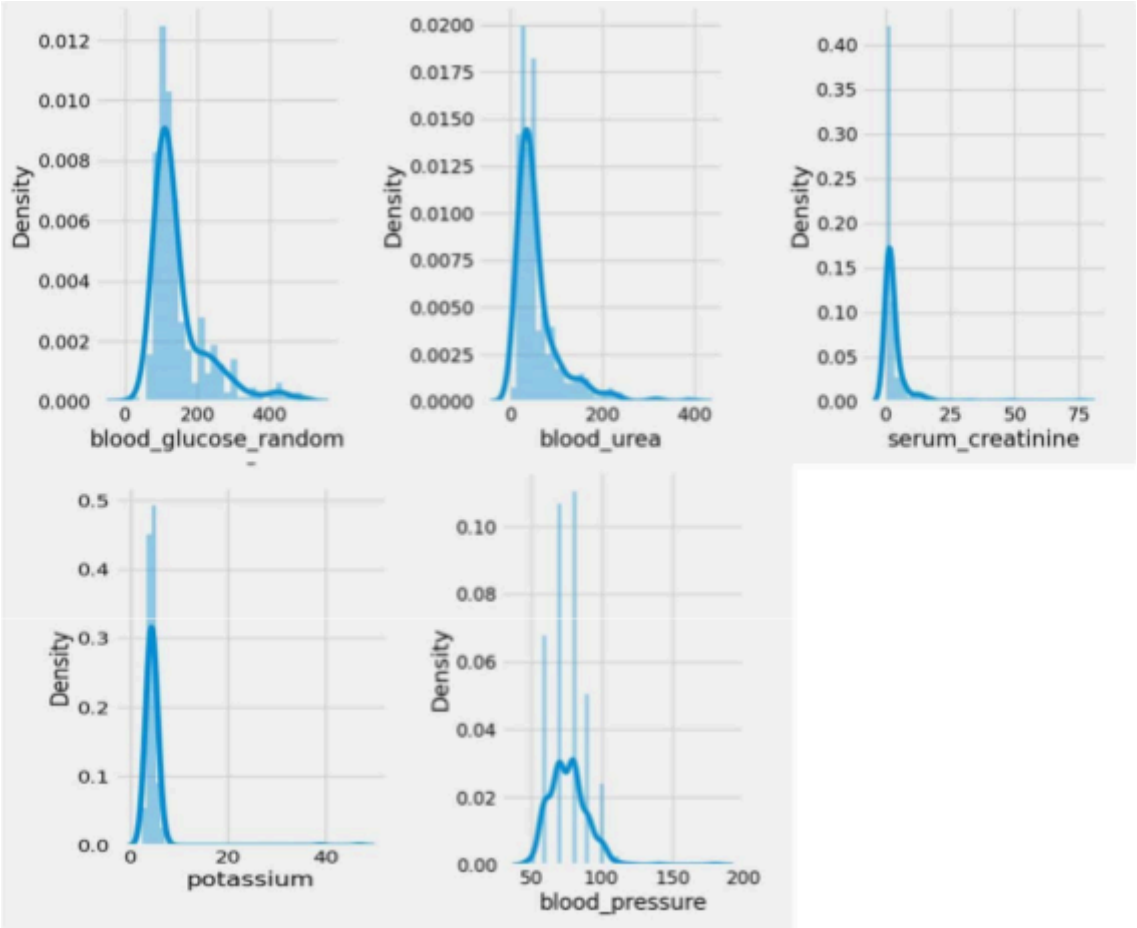
The following features in dataset are characterized by balance, uniformity, normality (if applicable), and suitability for analysis, enabling thorough exploration, meaningful insights, and reliable conclusions.



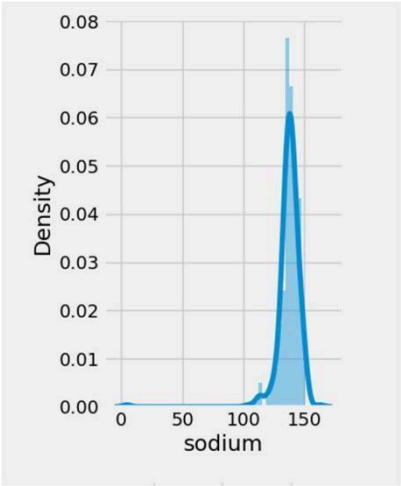
The above features Haemoglobin, Packed cell volume, White blood cell count, red blood cell count, Age and Specific gravity are well distributed features. However, age seems to be little left skewed and white blood cell is little right skewed they are negligible.

The sample has most ages ranging between 50-70, most RBC count at 5.0, most WBC count around 9000, most packed cell volume around 45 and most haemoglobin values ranging from 12-15.

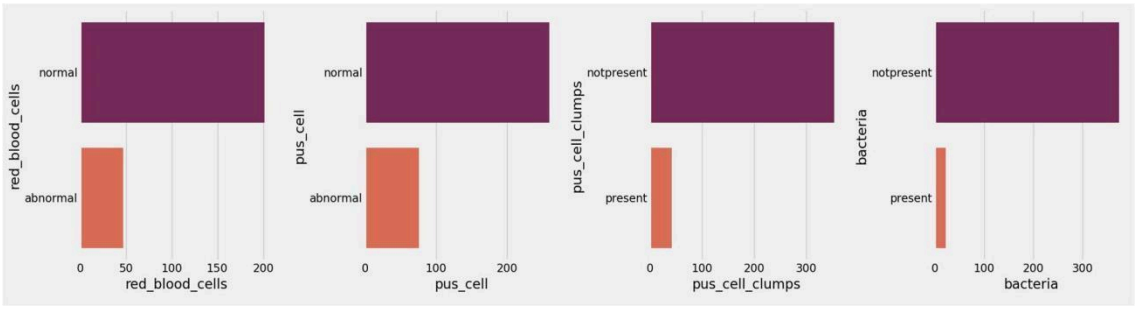
The following feature distributions are left skewed where the maximum population has ‘blood_glucose_random’ at 150-200 and ‘blood_urea’ at 30-70 and ‘serum_creatinine’ and ‘potassium’ value at 5 and ‘blood_pressure ’ ranging from 70-80.



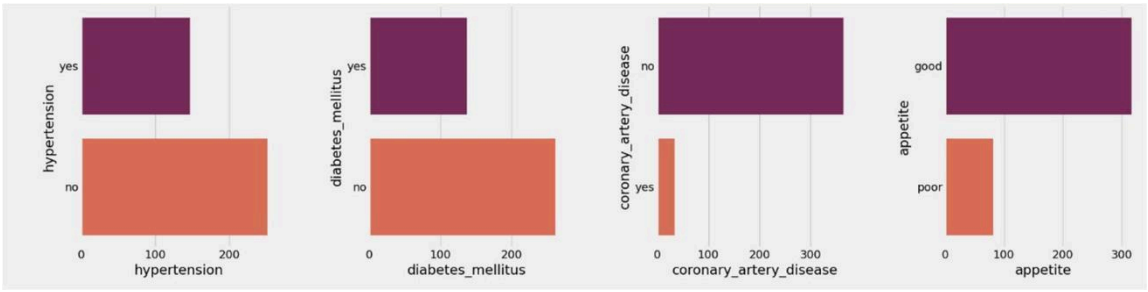
The following feature distribution is right skewed where the sodium levels are ranging from 100 to 150 with maximum at 130.



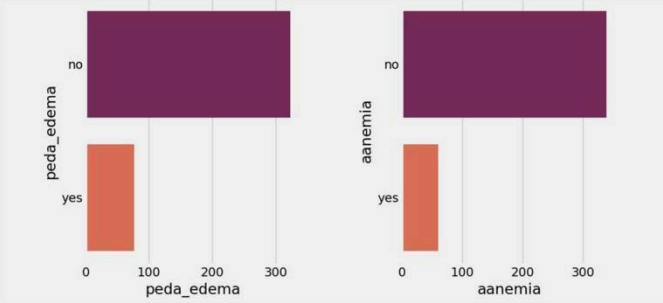
Observing the categorical columns:



The dataset reveals a reassuring trend where most individuals exhibit normal levels of red blood cell count and pus cell count, indicating a generally healthy urinary system. Moreover, the absence of pus cell clumps and bacteria in most cases suggests a low prevalence of urinary tract infections or other related disorders. These observations are significant as they reflect a population with good urinary system health, potentially indicating a dataset that comprises predominantly healthy individuals or individuals without severe urinary system abnormalities. However, further analysis and context about the dataset's population demographics and health conditions are necessary to validate these findings and draw conclusive insights about urinary system health trends.



The dataset's high proportion of individuals without hypertension, diabetes mellitus, or coronary artery diseases, alongside a prevalent good appetite, hints at a population with overall good health. While these observations align with common health trends, caution is necessary in generalizing. Real-world populations can be diverse, encompassing individuals with varying health statuses. Further analysis is crucial to validate these findings and ensure the dataset accurately represents real-world health patterns.



observation that most samples in the dataset do not exhibit edema (oedema) and anemia (anaemia) is noteworthy. Edema refers to the abnormal accumulation of fluid in tissues, often a sign of underlying health conditions such as heart failure or kidney disease. Anemia, on the other hand, indicates a lower-than-normal level of red blood cells or hemoglobin, which can result from various factors like nutritional deficiencies or chronic diseases

The absence of edema and anemia in the majority of samples suggests a dataset that predominantly comprises individuals without these specific health issues.

FEATURE ENGINEERING

Feature engineering is a critical process in machine learning that involves transforming raw data into meaningful features for model training. It includes tasks like handling missing values, encoding categorical variables, scaling numerical features, and creating new features from existing ones. The goal is to improve model performance by providing relevant, informative, and properly formatted input features.

All the categorical columns have two categories:

```
for col in cat_cols:
    print(f"{col} has {df[col].nunique()} categories\n")
```

```
red_blood_cells has 2 categories

pus_cell has 2 categories

pus_cell_clumps has 2 categories

bacteria has 2 categories

hypertension has 2 categories

diabetes_mellitus has 2 categories

coronary_artery_disease has 2 categories

appetite has 2 categories

peda_edema has 2 categories

aanemia has 2 categories

class has 2 categories
```

Each category represents a binary choice or classification within these columns, suggesting that these features are binary or Boolean in nature, where each observation falls into one of two distinct categories where the presence of abnormality is labelled as 1 and normal value is labelled as 0.

Label encoding is a technique used in data preprocessing to convert categorical variables into numerical format. In label encoding, each unique category or label within a categorical variable is assigned a numerical value.

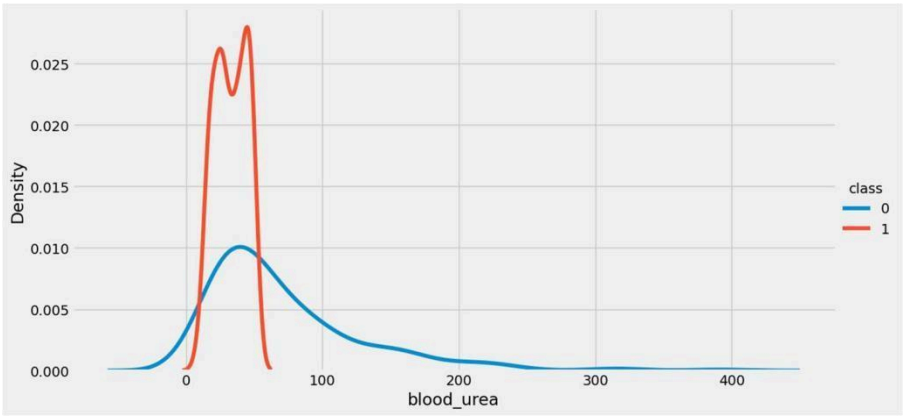
```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

for col in cat_cols:
    df[col] = le.fit_transform(df[col])
```

```
[66] df.head()
```

	age	blood_pressure	specific_gravity	albumin	sugar	red_blood_cells	pus_cell	pus_cell_clumps	bacteria
0	48.0	80.0	1.020	1.0	0.0	1	1	0	0
1	7.0	50.0	1.020	4.0	0.0	1	1	0	0
2	62.0	80.0	1.010	2.0	3.0	1	1	0	0
3	48.0	70.0	1.005	4.0	0.0	1	0	1	0
4	51.0	80.0	1.010	2.0	0.0	1	1	0	0

The distributions of the newly engineered variables are as follows.

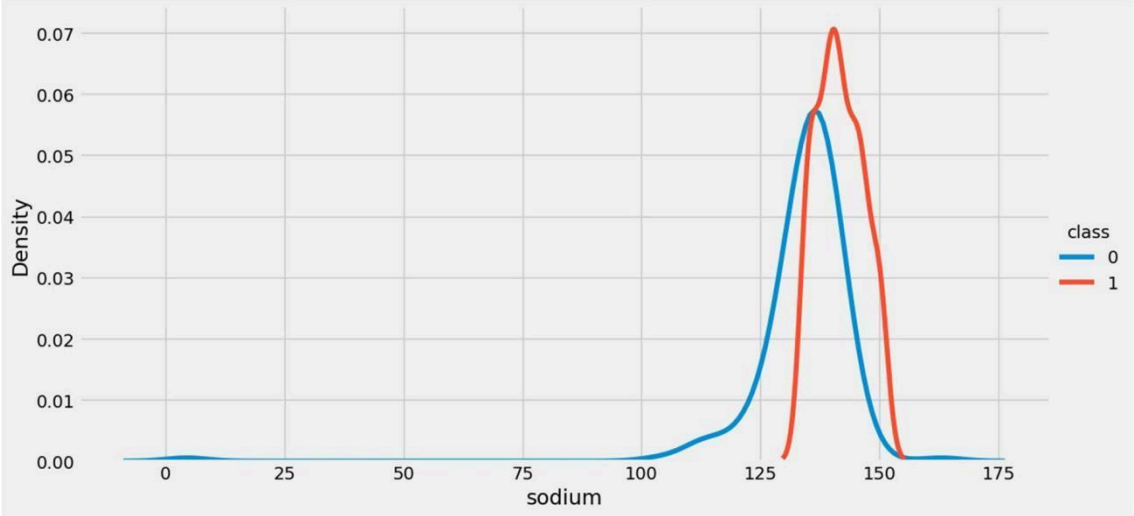
Blood urea level:



The above visualization depicts the relationship between blood urea levels and the density of observations for two classes related to chronic kidney disease (CKD). The y-axis represents density, ranging from 0.000 to 0.025, indicating the frequency of data points within specific blood urea level ranges. The x-axis shows blood urea levels, ranging from 0 to 400. Class 1 exhibits higher concentration around 0 to 70 suggesting a higher concentration of observations and possibly a stronger correlation with CKD presence or severity. In contrast, Class 0 has a peak reaching only 0.010, indicating a

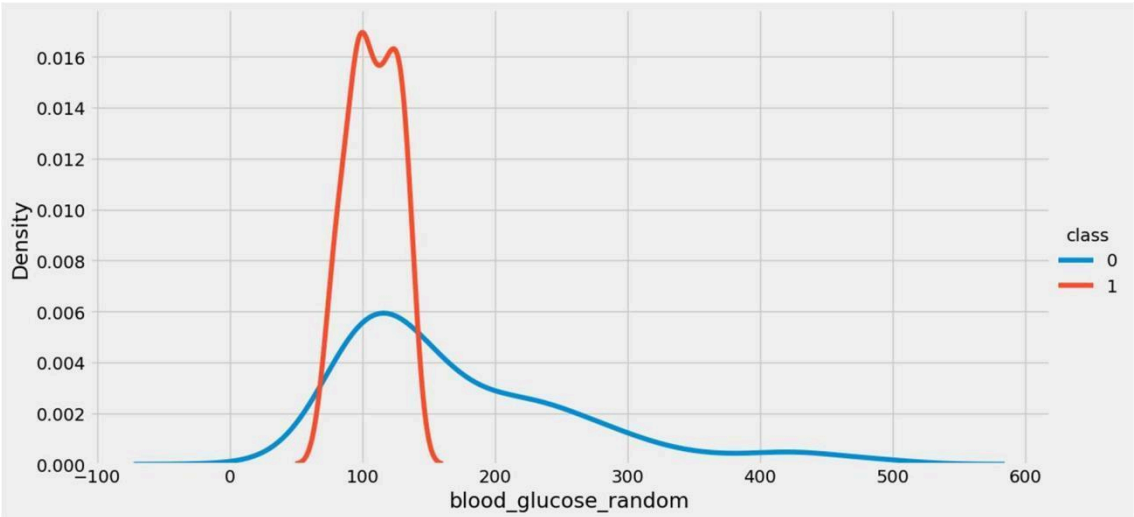
lower concentration of observations and potentially a lower likelihood or severity of CKD.

Sodium levels:



Class 1 exhibits a peak in density exceeding 0.07, suggesting a higher concentration of observations and possibly a stronger correlation with CKD presence or severity. In contrast, Class 0 has a peak less than 0.06, indicating a lower concentration of observations and potentially a lower likelihood or severity of CKD.

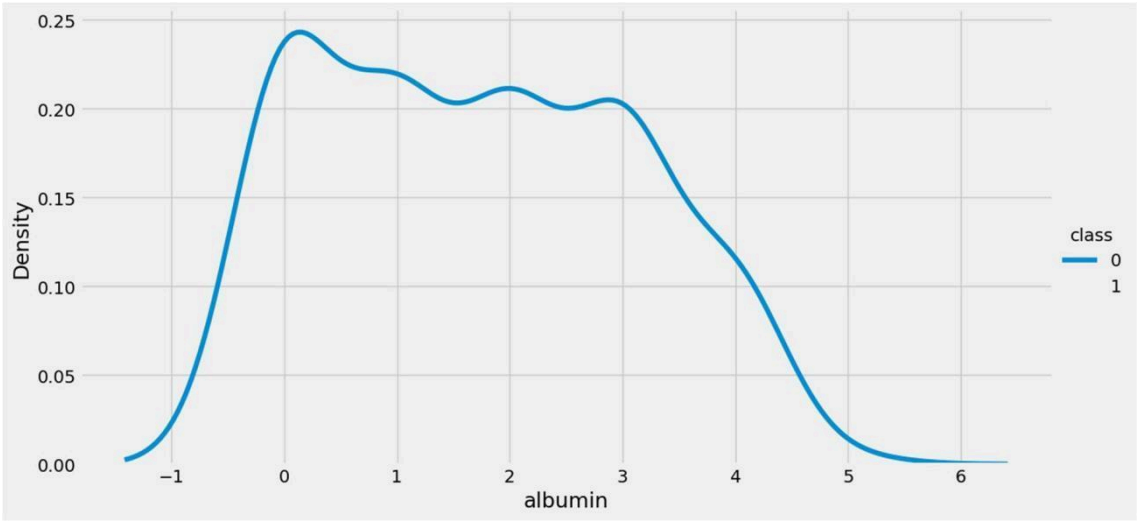
Blood glucose random:



Class 1 exhibits a peak in density exceeding 0.016, suggesting a higher concentration of observations and possibly a stronger correlation with CKD presence or severity. In

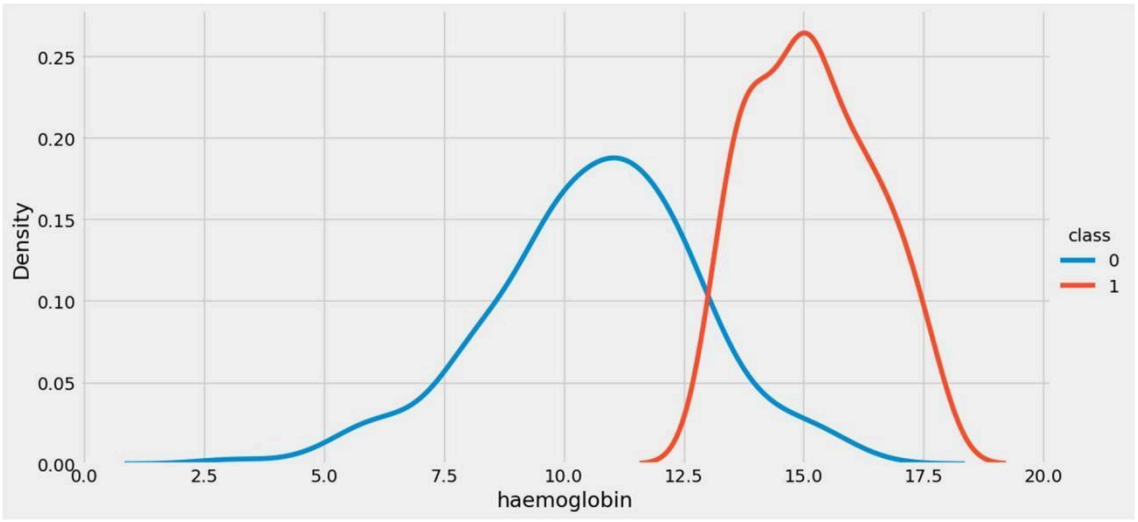
contrast, Class 0 has a peak reaching only 0.06, indicating a lower concentration of observations and potentially a lower likelihood or severity of CKD.

Albumin



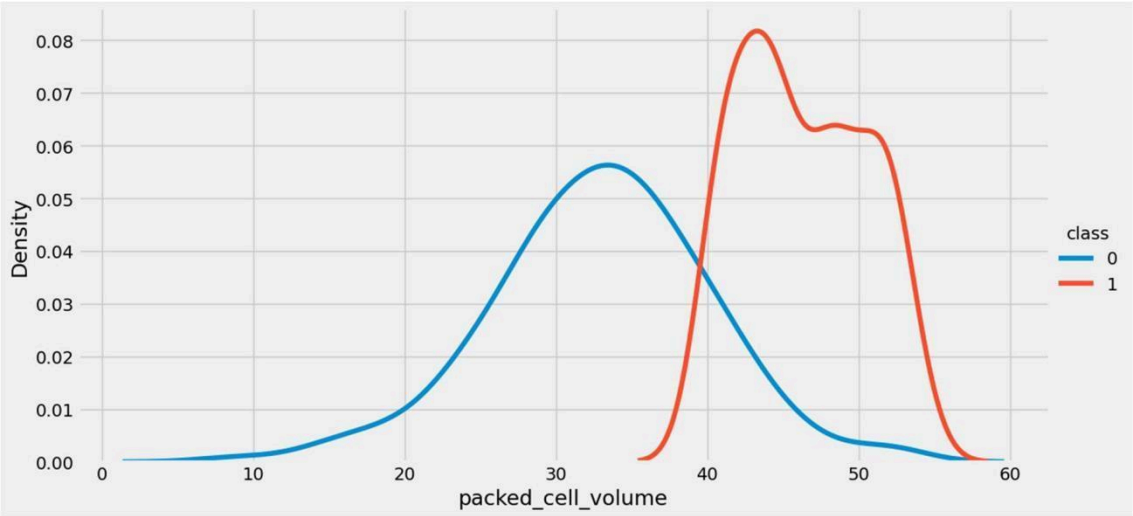
Only Class 0's distribution is visible in the plot, suggesting a specific pattern of albumin levels that may be associated with individuals without CKD or with lower CKD severity. The absence of Class 1's distribution indicates a significant difference in albumin levels between the two classes, highlighting albumin as a potential biomarker for CKD diagnosis or severity assessment.

Haemoglobin:



Class 0’s haemoglobin distribution is well distributed and the peak is less than 0.20 indicating low correlation with CKD where as class 1 has concentration from 12.5 to 18

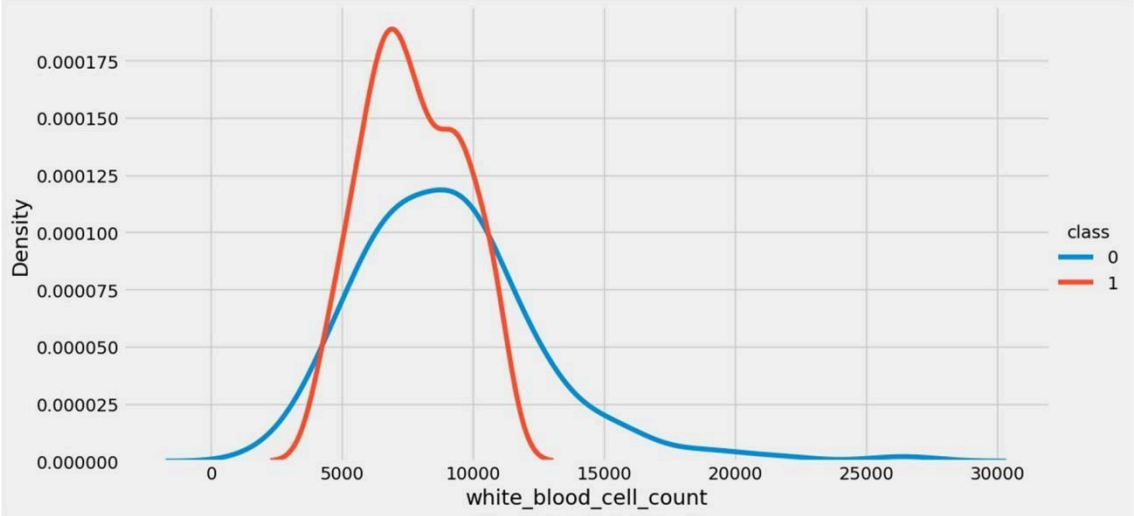
with high concentration at 15 whereas the class 0's distribution is highly concentrated between 10.0 to
12.5.



Packed cell volume:

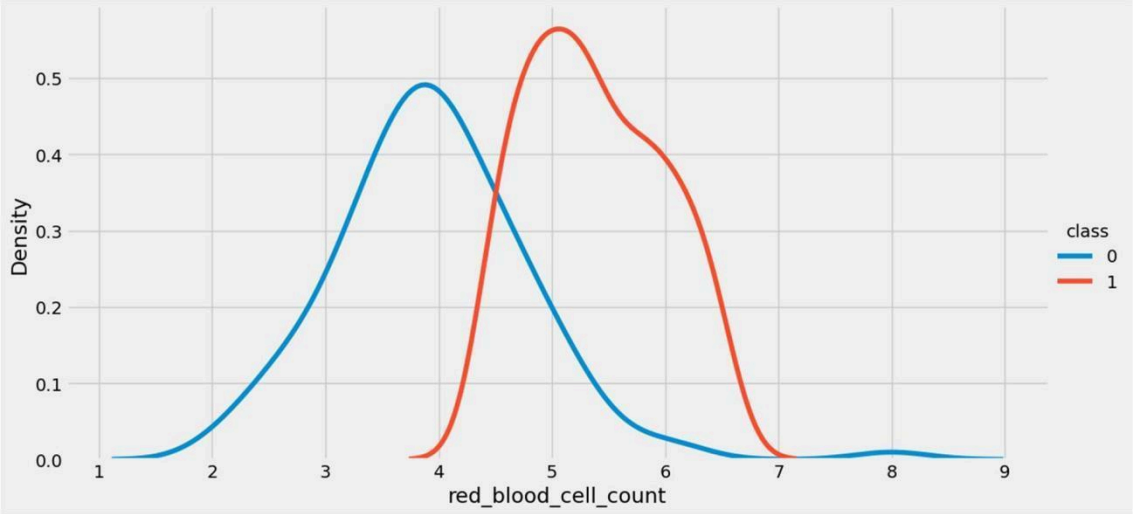
For class 0 packed cell volume has high frequency between 30-40 and for class 1 it is between 40 and 50. Showing the visible difference between their correlation with CKD.

White blood cell count:



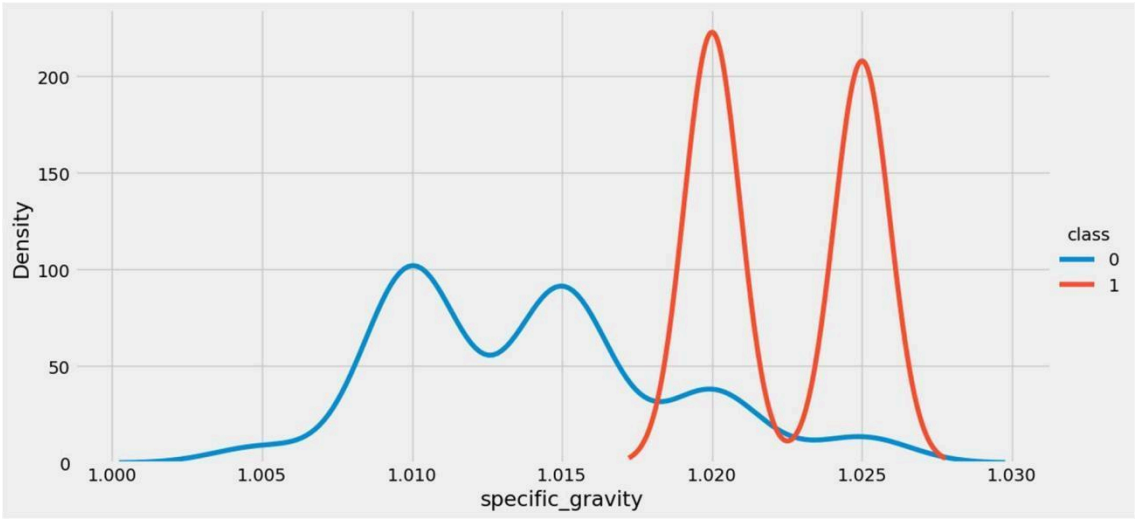
The WBC count for class 0 is concentrated from 5000 to 10000 but it is still less concentrated than class 1. Class 1 is highly concentrated between 5000-10000.

Red blood cell count:



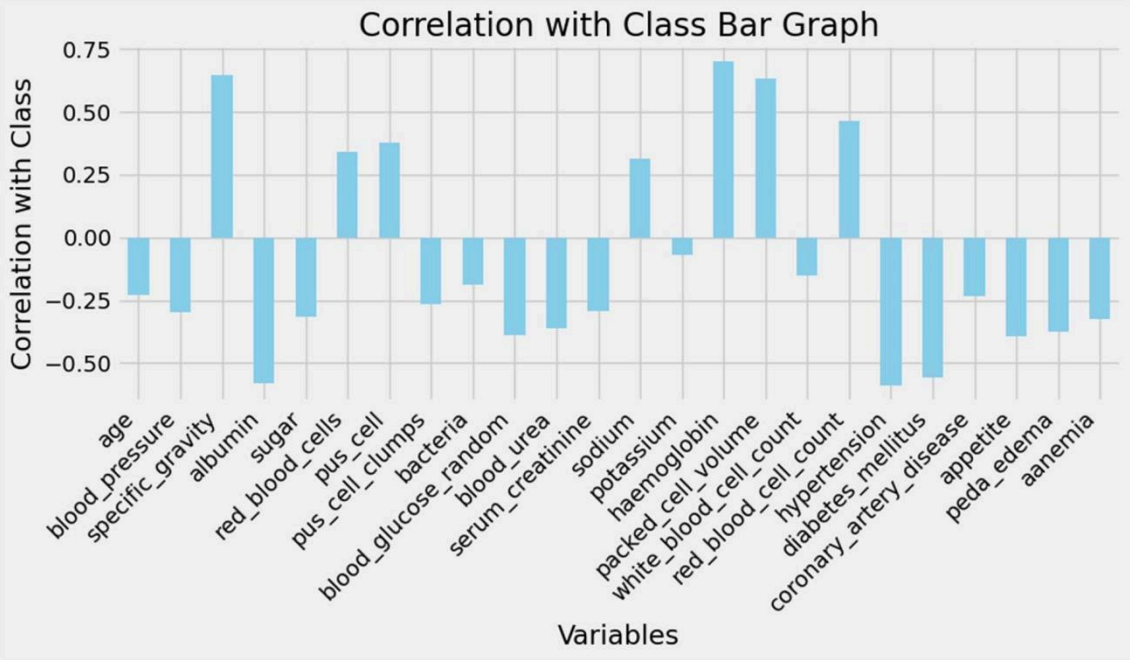
The RBC count for class 0 is highly concentrated at 4 and for class 1 it is at 5 and showing little less concentration at 6. This indicates that correlation between class 1 and CKD is pretty high.

Specific gravity:



For class 0 specific gravity is concentrated at1.010 and 1.015 where as for class 1 it is concentrated at 1.020 and 1.025 showing higher correlation with CKD.

CORRELATION



The bar graph visualizes various features and how they correlate to the diabetes target variable. Several variables are positively correlated with the target variable, with a few variables negatively correlated. Positive correlations indicate an increase in the variable is associated with an increase in the likelihood of having CKD, while negative correlations suggest inverse relationship.

MODEL BUILDING

After thorough preparation of the dataset and in depth understanding of its characteristics, the objective ahead was to build a model that correctly predicts diabetes. This stage involves deploying different machine learning algorithms, and making use of the powerful Python libraries, to reveal patterns and relationships within the data.

Selection of Machine Learning Algorithms:

Given the nature of the health data, applicable models need to be chosen since the results must be interpretable so that they serve the business purposes. we decided to deploy the following models since they are tailored for binary classification tasks and that they are able to reveal actionable insights. The models are: Random Forest Classifier, Gradient Boosting Classifier, Xg Boost, Decision Tree Classifier and KNN.

Model Training and Evaluation:

After tuning the hyperparameters the models were trained on the selected features. Data was split on 70% training to 30% testing ratio, the testing set is kept separate during modelling process. Performance of a model on unseen data is the true test to certify that the algorithm generalization capabilities, ensuring the model can make accurate predictions on new realworld data. The efficiency of each model was evaluated on accuracy, precision, recall and F1 score metrics. This allowed the assessment and determination of each model's ability to appropriately categorize instances of CKD and non-CKD, while striking a balance of both sensitivity and specificity.

```
[85] # splitting data into training and test set

     from sklearn.model_selection import train_test_split

     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.30, random_state = 0)
```

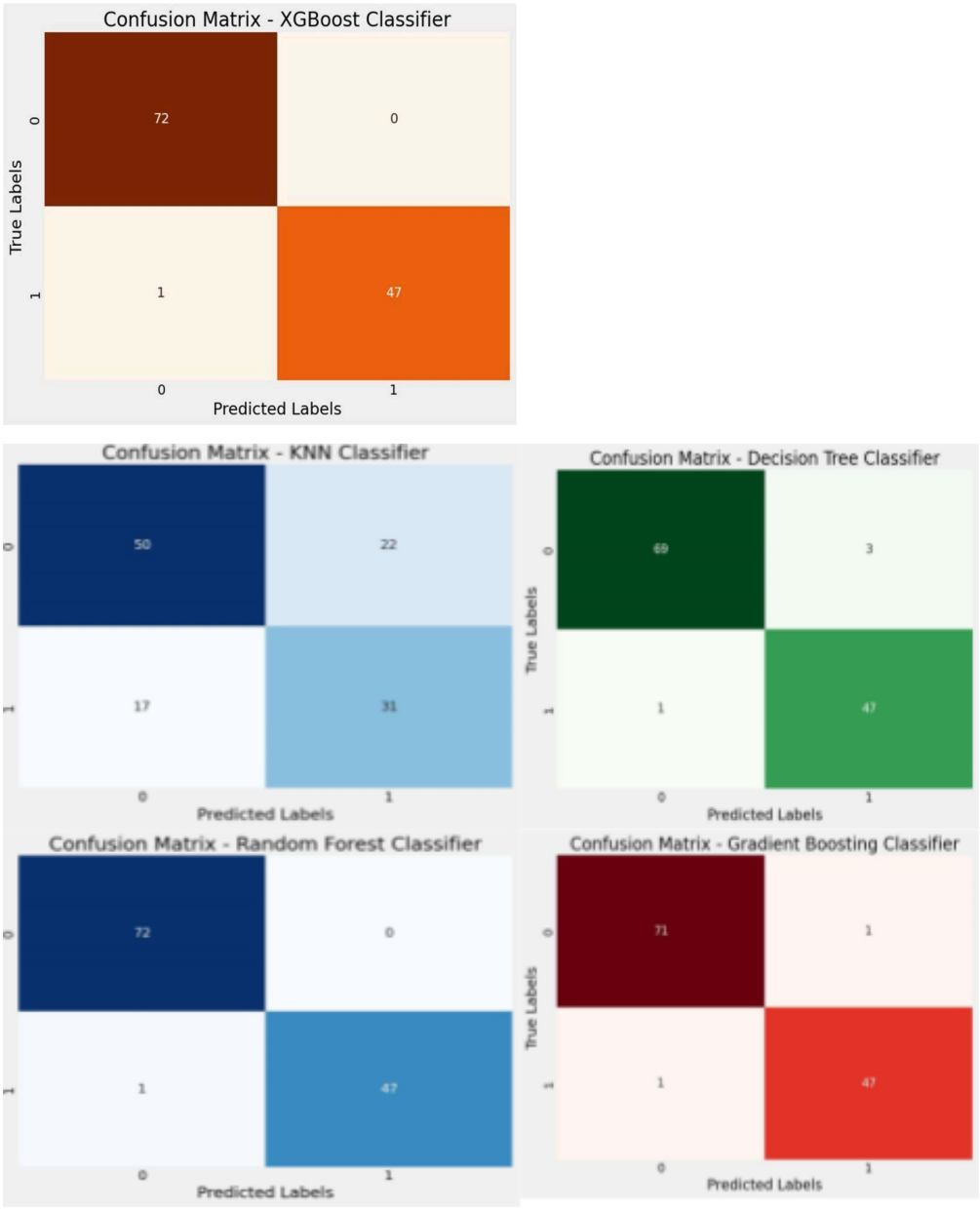
RESULTS

Accuracy serves as an overall measure of model's correctness, it is a ratio of correctly predicted instances and the total number of instances, Precision measures model's ability to correctly pinpoint positive occurrence among the predicted positives. A high precision indicates reduced likelihood of false positives. Sensitivity on the other hand ascertains the model's ability to identify all positive instances among the actual positives. F1 score is a balance of precision and recall, as it is the harmonic mean of both.

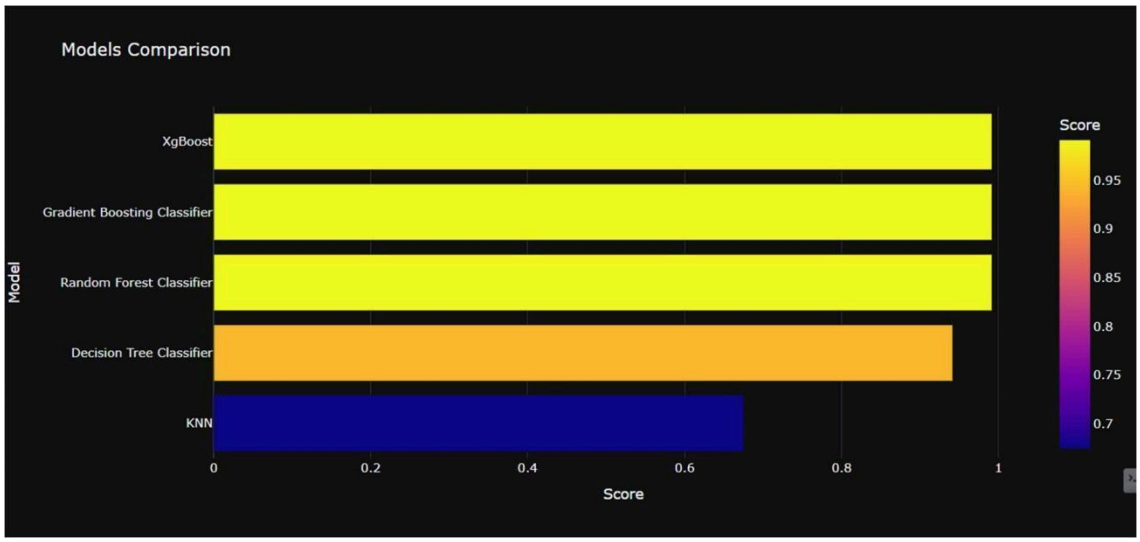
Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-Nearest Neighbours (KNN)	70	75	0.72	70
Decision Tree	97	97	0.97	97
Random Forest	99	99	0.99	99

Gradient Boosting	98	98	0.98	98
XG Boost	99	99	0.99	99

The table indicates that the Decision Tree, Random Forest, Gradient Boosting, and XG Boost classifiers have significantly outperformed the K-Nearest Neighbours (KNN) model across all performance metrics.



MODELS COMPARISION:



XG BOOST and RANDOM FOREST models have shown the highest accuracy compared to the other models i.e. 99%.

CONCLUSION

In conclusion, this project focused on using a variety of features related to chronic kidney disease to train and test machine learning models for prediction purposes. Features such as age, blood pressure, specific gravity, albumin, sugar, blood glucose random, and others were utilized in building models like K-Nearest neighbours (KNN), Decision Tree Classifier, Random Forest Classifier, Gradient Boosting Classifier, and XGBoost Classifier. The objective was to accurately predict the presence of chronic kidney disease based on these features.

Through extensive data preprocessing, feature engineering, and model training, we achieved promising results in terms of model accuracy and performance. The models demonstrated the ability to effectively distinguish between individuals with and without chronic kidney disease, showcasing the potential for early detection and intervention.

Overall, this project underscores the importance of leveraging machine learning techniques in healthcare applications, particularly in diagnosing chronic diseases like kidney disease. The models developed in this project can serve as valuable tools for healthcare professionals in making informed decisions and improving patient outcomes through timely interventions. Further research and refinement of these models could

lead to enhanced predictive capabilities and ultimately contribute to better healthcare management and patient care.

ENHANCEMENTS AND FUTURE DIRECTIONS

Feature Refinements: Review and expand the feature set to include potential additional indicators that could enhance the predictive capabilities of the models. This may involve collaborating with domain experts to identify relevant clinical, lifestyle, and genetic factors.

Hyperparameter Optimization: Explore hyperparameter tuning techniques to identify the optimal set of parameters that improve the accuracy and robustness of the predictive models. Consider experimenting with different algorithms and ensemble methods to further refine model performance.

Model Validation: Extend model validation efforts by testing the models on new and diverse datasets. This step helps assess the generalizability of the models across different populations and healthcare settings, ensuring their reliability and effectiveness in real-world scenarios.

Domain Expert Collaboration: Collaborate closely with healthcare professionals and domain experts specializing in nephrology to gain insights into the clinical relevance of features. This collaboration can lead to the development of more actionable and clinically meaningful predictive models.

Continuous Model Improvement: Implement a strategy for continuous model improvement and adaptation by incorporating new data periodically. This ensures that the models remain relevant and aligned with evolving healthcare trends, while also monitoring their performance over time for any necessary adjustments.

By implementing these next steps, we can further refine and optimize the predictive framework for chronic kidney disease prediction, ultimately contributing to improved patient outcomes and healthcare decision-making.

REFERENCES

1. Levey, A. S., Eckardt, K.-U., Dorman, N. M., Christiansen, S. L., Cheung, M., Jadoul, M., & Winkelmayer, W. C. (2020). Nomenclature for kidney function and disease: Report of a Kidney Disease: Improving Global Outcomes (KDIGO) Consensus Conference. Kidney International, 97(6), 1117–1129. https://doi.org/10.1016/j.kint.2020.02.010
2. Jha, V., Garcia-Garcia, G., Iseki, K., Li, Z., Naicker, S., Plattner, B., ... Yang, C.-W. (2013). Chronic kidney disease: Global dimension and perspectives. Lancet, 382(9888), 260–272. https://doi.org/10.1016/S0140-6736(13)60687-X
3. Boehringer Ingelheim Pharmaceuticals, Inc. (n.d.). Chronic kidney disease - JARDIANCE® (empagliflozin) tablets. Retrieved April 18, 2024, from https://patient.boehringer-ingelheim.com/us/products/jardiance/chronic-kidney-disease
4. Inker, L. A., Astor, B. C., Fox, C. H., Isakova, T., Lash, J. P., Peralta, C. A., ... & Feldman, H. I. (2014). KDOQI US commentary on the 2012 KDIGO clinical practice guideline for the evaluation and management of CKD. *American Journal of Kidney Diseases, 63*(5), 713-735. https://doi.org/10.1053/j.ajkd.2014.01.416
5. Tsai, W.-C., Wu, H.-Y., Peng, Y.-S., Ko, M.-J., Wu, M.-S., Hung, K.-Y., Wu, K.-D., Chu, T.-S., & Chien, K.-L. (2016). Risk factors for development and progression of chronic kidney disease: A systematic review and exploratory meta-analysis. *Medicine, 95*(11), e3013. https://doi.org/10.1097/MD.0000000000003013
6. Qi, L., Fan, Q.-L., Han, Q.-X., Geng, W.-J., Zhao, H.-H., Ding, X.-N., ... & Zhu, H.-Y. (2020). Machine learning in nephrology: scratching the surface. *Chinese Medical Journal, 133*(6), 687-698. https://doi.org/10.1097/CM9.0000000000000694
7. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785
8. Kaggle. (n.d.). Find Open Datasets and Machine Learning Projects. Retrieved April 20, 2024, from https://www.kaggle.com/datasets.