CHAPTER-1

INTRODUCTION

Diabetes: A Growing Challenge with Vast Data Potential

Diabetes is a chronic, metabolic disease characterized by elevated levels of blood glucose (or blood sugar), which leads over time to serious damage to the heart, blood vessels, eyes, kidneys and nerves. Early diagnosis and intervention are crucial for managing the condition and improving patient outcomes. Fortunately, the vast amount of data available on diabetes presents an opportunity to develop more accurate and effective diagnostic tools.

This project tackles the challenge of diabetes prediction using machine learning. We leverage a secondary dataset obtained from the Kaggle website. This dataset, originating from the Iraqi population, offers valuable insights into diabetes risk factors specific to that region. The data were collected from patients at Medical City Hospital and the Specializes Center for Endocrinology and Diabetes-Al-Kindy Teaching Hospital. Patient records were meticulously reviewed, and relevant medical information and laboratory analysis results were extracted to form the diabetes dataset.

The dataset encompasses a rich set of features, including demographic information (age, gender) and crucial laboratory tests (blood sugar level, creatinine ratio, body mass index, cholesterol, fasting lipid profile, HbA1c). The objective of this project is to develop and compare logistic regression models built using Python to predict the likelihood of diabetes in patients based on these combined clinical and biochemical parameters.

By analyzing this data, we aim to:

Develop Logistic Regression Models: We will construct two logistic regression models to predict diabetes risk. One model will use the data without outlier removal, while the other will remove outliers before analysis. This approach allows us to assess the impact of outliers on model performance.

Evaluate Model Performance: We will utilize various evaluation metrics (accuracy, precision, recall, F1-score) to compare the performance of both models. This comparison will reveal the impact of outlier removal on the model's ability to accurately classify diabetic and non-diabetic patients.

Gain Insights into Diabetes Prediction: The findings from this project will contribute to our understanding of how well a combination of demographic information and basic health measurements can predict the presence of diabetes. This knowledge can pave the way for further research into more sophisticated prediction models incorporating additional clinical data.

Analyze Outlier Impact: By comparing the performance of models with and without outlier removal, we can gain valuable insights into the importance of data quality for building robust logistic regression models. This knowledge can be applied to future research endeavors using similar datasets.

Through this project, we hope to leverage machine learning techniques and readily available data to contribute to the fight against diabetes**.**

Diabetes Overview

Diabetes mellitus encompasses a spectrum of metabolic disorders characterized by hyperglycemia resulting from defects in insulin secretion, insulin action, or both. The condition leads to long-term complications affecting various organs and systems, including the heart, blood vessels, eyes, kidneys, and nerves.

Classification

Diabetes is classified into three main categories: diabetic, non-diabetic, and pre-diabetic.

1. Diabetic (Class 1):

Meaning: Individuals classified as diabetic have been diagnosed with diabetes mellitus. This chronic metabolic disorder is characterized by high blood sugar levels resulting from either inadequate insulin production, insulin resistance, or both.

Impacts:

Health Complications: Diabetic individuals are at increased risk of developing various health complications, including cardiovascular disease, neuropathy (nerve damage), nephropathy (kidney disease), retinopathy (eye damage), and peripheral vascular disease.

Quality of Life: Managing diabetes requires strict adherence to medication, lifestyle modifications (such as dietary changes and regular exercise), and frequent monitoring of blood glucose levels. The condition can significantly impact an individual's quality of life due to the need for constant vigilance and potential complications.

Long-Term Consequences: Poorly managed diabetes can lead to severe complications, such as heart attacks, strokes, kidney failure, blindness, and lower limb amputations, significantly reducing life expectancy.

Symptoms: Common symptoms of diabetes include frequent urination, excessive thirst, unexplained weight loss, fatigue, blurred vision, slow wound healing, and recurrent infections.

2. Non-Diabetic (Class 2):

Meaning: Individuals classified as non-diabetic have blood glucose levels within the normal range and do not exhibit signs or symptoms of diabetes mellitus.

Impacts:

Lower Risk of Complications: Non-diabetic individuals have a significantly lower risk of developing the complications associated with diabetes, such as cardiovascular disease, neuropathy, nephropathy, retinopathy, and peripheral vascular disease.

Reduced Healthcare Burden: Since non-diabetic individuals do not require regular monitoring or treatment for diabetes, they typically experience fewer healthcare-related expenses and a lower burden on healthcare systems.

Symptoms: Non-diabetic individuals do not experience the symptoms typically associated with diabetes, such as frequent urination, excessive thirst, unexplained weight loss, and fatigue.

3. Pre-Diabetic (Class 3): Individuals classified as pre-diabetic have blood glucose levels that are higher than normal but not yet high enough to be diagnosed as diabetic. Pre-diabetes is considered a warning sign and an intermediate stage between normal glucose metabolism and diabetes.

Impacts:

Increased Risk: Pre-diabetic individuals are at a higher risk of developing type 2 diabetes compared to those with normal glucose levels. Without intervention, many individuals with pre-diabetes may progress to diabetes within a few years.

Opportunity for Prevention: Pre-diabetes offers an opportunity for early intervention through lifestyle modifications such as weight loss, healthy diet changes, increased physical activity, and medication (e.g., metformin). These interventions can delay or even prevent the onset of diabetes and its associated complications.

Symptoms: Pre-diabetic individuals may not experience any symptoms initially, but some may exhibit mild signs of insulin resistance, such as fatigue, increased hunger, and difficulty losing weight.

Understanding the distinctions between these three classes is crucial for identifying individuals at risk of diabetes, implementing preventive measures, and providing appropriate management strategies to improve health outcomes and quality of life.

1. Urea

In individuals with diabetes, urea levels can serve as an important indicator of kidney function. Diabetes is a leading cause of kidney disease, known as diabetic nephropathy, which can result in impaired kidney function and decreased ability to excrete urea efficiently. Elevated urea levels may suggest the presence of renal dysfunction, highlighting the importance of regular monitoring in diabetic patients to assess kidney health and prevent complications.

Urea: Adult males (18-60 years): 6.0-20.0 mg/dL; Adult females (18-60 years): 2.5-7.0 mg/dL (These ranges can vary depending on age and other factors. Consult a medical professional for specific interpretation**)**

2. Cr (Creatinine)

Creatinine, like urea, is a marker commonly used to evaluate kidney function. In diabetes, sustained high blood sugar levels can damage the small blood vessels in the kidneys, leading to diabetic nephropathy. As kidney function declines, creatinine clearance decreases, resulting in elevated blood creatinine levels. Monitoring creatinine levels is crucial in diabetic individuals to detect early signs of kidney damage and initiate interventions to slow disease progression.

Creatinine: Adult males: 0.6-1.2 mg/dL; Adult females: 0.5-1.1 mg/dL (Similar to urea, these ranges can vary based on factors like age and muscle mass. Consult a medical professional for specific interpretation)

3.HbA1c (Glycated Hemoglobin) : HbA1c is a key biomarker for assessing long-term glycemic control in individuals with diabetes. Elevated blood glucose levels lead to the non-enzymatic glycation of hemoglobin molecules, forming HbA1c. The measurement of HbA1c reflects average blood glucose levels over the preceding 2-3 months, providing valuable information about overall diabetes management and the risk of complications such as cardiovascular disease, neuropathy, and retinopathy. Tight control of HbA1c levels is essential to reduce the risk of diabetic complications and improve long-term outcomes in diabetes patients.

Normal HbA1c Range: Typically below 5.7%

Prediabetes HbA1c Range: Between 5.7% and 6.4%

Diabetes HbA1c Range: Generally 6.5% or higher

4. Chol (Total Cholesterol): Dyslipidemia, characterized by abnormal cholesterol levels, is a common comorbidity in individuals with diabetes. High blood sugar levels and insulin resistance disrupt lipid metabolism, leading to elevated total cholesterol levels. Dyslipidemia contributes to the increased risk of cardiovascular disease, which is the leading cause of morbidity and mortality in diabetes patients. Monitoring total cholesterol levels and implementing lifestyle modifications and lipid-lowering medications are essential strategies in the comprehensive management of diabetes and its associated complications.

Cholesterol:

Total Cholesterol: Desirable: Less than 200 mg/dL; Borderline high: 200-239 mg/dL; High: 240 mg/dL or higher

LDL Cholesterol: Optimal: Less than 100 mg/dL; Near optimal: 100-129 mg/dL; Borderline high: 130-159 mg/dL; High: 160-189 mg/dL; Very high: 190 mg/dL or higher

HDL Cholesterol: Higher levels are generally better: Less than 50 mg/dL (low); 50-60 mg/dL (average); 60 mg/dL or higher (high**)**

5. TG (Triglycerides):

Elevated triglyceride levels are frequently observed in individuals with diabetes and are closely associated with insulin resistance, obesity, and metabolic syndrome. Insulin resistance leads to increased lipolysis in adipose tissue and elevated hepatic production of triglyceride-rich lipoproteins, contributing to hypertriglyceridemia. High triglyceride levels are an independent risk factor for cardiovascular disease and are commonly found in diabetic patients with atherogenic dyslipidemia. Lifestyle modifications, including dietary changes and increased physical activity, are essential for managing triglyceride levels and reducing cardiovascular risk in diabetes patients.

Triglycerides: Normal: Less than 150 mg/dL; Borderline high: 150-199 mg/dL; High: 200-499 mg/dL; Very high: 500 mg/dL or higher

Obesity: 30.0 kg/m^2 or higher (Class 1: 30.0-34.9 kg/m^2; Class 2: 35.0-39.9 kg/m^2; Class 3: 40.0 kg/m^2 or higher)

6. HDL (High-Density Lipoprotein):

HDL cholesterol, often referred to as "good cholesterol," plays a crucial role in reverse cholesterol transport and protects against atherosclerosis. However, individuals with diabetes frequently exhibit dysregulated HDL metabolism characterized by low HDL levels and impaired HDL functionality. Insulin resistance and hyperglycemia contribute to the downregulation of key enzymes involved in HDL metabolism, leading to dysfunctional HDL particles with reduced antiatherogenic properties. Low HDL levels are a significant risk factor for cardiovascular disease in diabetes patients and warrant targeted interventions to improve lipid profiles and reduce cardiovascular risk.

7. LDL (Low-Density Lipoprotein):

LDL cholesterol, commonly known as "bad cholesterol," is a primary contributor to the development of atherosclerosis and cardiovascular disease. Diabetes is associated with an atherogenic lipid profile characterized by elevated LDL cholesterol levels and increased susceptibility to LDL oxidation. Hyperglycemia promotes the glycation of LDL particles, rendering them more atherogenic and proinflammatory. Elevated LDL cholesterol levels are a major modifiable risk factor for cardiovascular disease in individuals with diabetes, highlighting the importance of aggressive lipid-lowering therapy to reduce cardiovascular morbidity and mortality.

8. VLDL (Very-Low-Density Lipoprotein):

VLDL particles are triglyceride-rich lipoproteins synthesized by the liver and play a critical role in lipid metabolism. In individuals with diabetes, insulin resistance and hyperinsulinemia promote the overproduction of VLDL by the liver, leading to elevated circulating levels of triglycerides. Excess VLDL contributes to the development of atherogenic dyslipidemia, characterized by elevated triglycerides and reduced HDL cholesterol levels, which are associated with an increased risk of cardiovascular disease. Targeting VLDL metabolism through lifestyle interventions and pharmacotherapy is essential for managing dyslipidemia and reducing cardiovascular risk in diabetes patients.

9. BMI (Body Mass Index):

Obesity and excess body weight are significant risk factors for the development of type 2 diabetes. Adipose tissue dysfunction and chronic low-grade inflammation associated with obesity contribute to insulin resistance and the pathogenesis of type 2 diabetes. BMI, a measure of body weight relative to height, serves as a simple and practical tool for assessing adiposity and identifying individuals at increased risk of diabetes. Lifestyle modifications focused on weight management, including dietary changes, regular physical activity, and behavior therapy, are fundamental components of diabetes prevention and management strategies.

BMI:

Normal weight: 18.5-24.9 kg/m²

Overweight: 25.0-29.9 kg/m²

By examining these variables in the context of diabetes, we gain valuable insights into the complex interplay between metabolic, biochemical, and physiological factors underlying the pathogenesis and progression of the disease. Understanding the relationships between these variables can inform risk stratification, personalized treatment approaches, and interventions aimed at improving outcomes and reducing the burden of diabetes and its complications on individuals and healthcare systems

CHAPTER -2

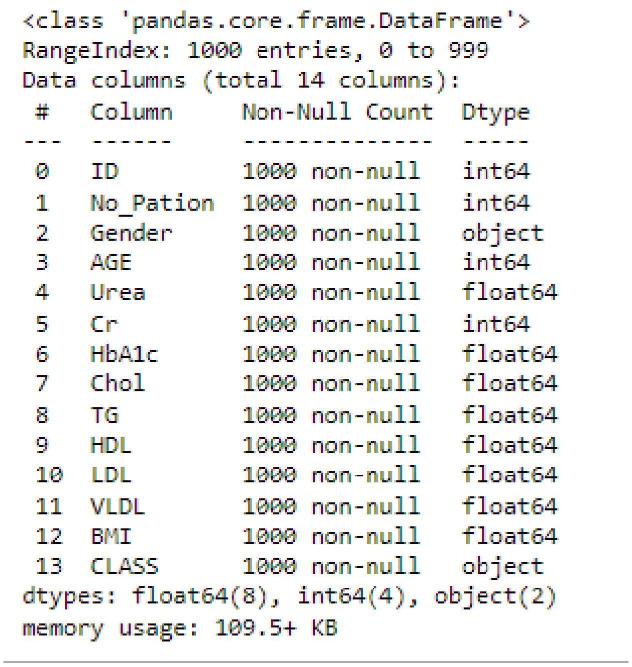
DATA DESCRIPTION

About Dataset:

The Data used in this project is a Secondary data taken from the Kaggle website.

Diabetes is an opportune disease which has large wealth of data available and has with it huge complications. There is a need for a better and a more accurate approach in the diagnosis of the disease

The data were collected from the Iraqi society, as they data were acquired from the laboratory of Medical City Hospital and (the Specializes Center for Endocrinology and Diabetes-Al-Kindy Teaching Hospital). Patients' files were taken and data extracted from them and entered in to the database to construct the diabetes dataset. The data consist of medical information, laboratory analysis.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 14 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   ID         1000 non-null   int64  
 1   No_Pation  1000 non-null   int64  
 2   Gender     1000 non-null   object 
 3   AGE        1000 non-null   int64  
 4   Urea       1000 non-null   float64
 5   Cr         1000 non-null   int64  
 6   HbA1c      1000 non-null   float64
 7   Chol       1000 non-null   float64
 8   TG         1000 non-null   float64
 9   HDL        1000 non-null   float64
 10  LDL        1000 non-null   float64
 11  VLDL       1000 non-null   float64
 12  BMI        1000 non-null   float64
 13  CLASS      1000 non-null   object 
dtypes: float64(8), int64(4), object(2)
memory usage: 109.5+ KB
```

VARIABLE DESCRIPTION

Variable	Description	Data Type
ID	Unique identifier for each record	int64
No_Pation	Patient number or identifier	int64
Gender	Gender of the patient	object
AGE	Age of the patient	int64
Urea	Level of urea in the patient's blood	float64
Cr	Blood metric, possibly creatinine levels	int64
HbA1c	Glycated hemoglobin levels	float64
Chol	Cholesterol levels	float64
TG	Triglyceride levels	float64
HDL	High-density lipoprotein (HDL) cholesterol levels	float64
LDL	Low-density lipoprotein (LDL) cholesterol levels	float64
VLDL	Very low-density lipoprotein (VLDL) cholesterol levels	float64
BMI	Body mass index (BMI)	float64
CLASS	Class or category of patients	object

OBJECTIVE:

The objective of this project is to develop and compare logistic regression models using Python to predict likelihood of diabetes in patients based on their clinical and biochemical parameters. By analyzing a dataset containing patient demographics and relevant medical measurements, on diabetes risk

We will first create a logistic regression model without outlier removal to assess the predictive power of the features (Gender, AGE, Urea, Cr, HbA1c, Chol, TG, HDL, LDL, VLDL, BMI) in identifying diabetes. This initial model will serve as a baseline to understand the potential of these features for classification. We will then develop a second logistic regression model after eliminating outliers from the data. This step aims to evaluate the impact of outliers on the model's performance in classifying diabetic and non-diabetic patients based on the aforementioned features.

By comparing the performance metrics (accuracy, precision, recall, F1-score) of both models, we will analyze the effect of outlier removal on the model's ability to predict diabetes. This comparison will help determine whether outlier removal is necessary for achieving the best performance in this specific scenario, focusing on the combined influence of demographic and basic health measurement data. This project seeks to contribute to the understanding of how well a combination of demographic information and basic health measurements can predict the presence of diabetes. The findings can inform further research into diabetes prediction models incorporating additional clinical or biochemical data. Additionally, the analysis of outlier impact can provide insights into data quality considerations for building robust logistic regression models.

CHAPTER-3

RESEARCH METHODOLOGY

MACHINE LEARNING :

Machine learning (ML) is a branch of artificial intelligence (AI) focused on developing algorithms that learn from data and make predictions or decisions without being explicitly programmed. It utilizes statistical techniques to enable computers to improve their performance on a task through experience. ML techniques, including artificial neural networks, have shown significant advancements, particularly in fields like natural language processing, computer vision, and medicine. While not all ML methods rely on statistics, computational statistics plays a crucial role in shaping the field. ML finds applications across various domains, from business analytics to healthcare, and is often used interchangeably with predictive analytics. The theoretical underpinnings of ML are supported by mathematical optimization techniques, with data mining serving as a complementary field for exploratory data analysis. The probably approximately correct (PAC) learning framework offers a theoretical perspective on ML algorithms' capabilities and limitations.

Approaches :

Machine learning approaches are traditionally divided into three broad categories, which correspond to learning paradigms, depending on the nature of the "signal" or "feedback" available to the learning system:

Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.[6]

LOGISTIC REGRESSION:

Logistic regression is a supervised machine learning algorithm widely used for binary classification tasks, such as identifying whether an email is spam or not and diagnosing diseases by assessing the presence or absence of specific conditions based on patient test results. This approach utilizes the logistic (or sigmoid) function to transform a linear combination of input features into a probability value ranging between 0 and 1. This probability indicates the likelihood that a given input corresponds to one of two predefined categories. The essential mechanism of logistic regression is grounded in the logistic function's ability to model the probability of binary outcomes accurately. With its distinctive S-shaped curve, the logistic function effectively maps any real-valued number to a value within the 0 to 1 interval. This feature renders it particularly suitable for binary classification tasks, such as sorting emails into "spam" or "not spam". By calculating the probability that the dependent variable will be categorized into a specific group, logistic regression provides a probabilistic framework that supports informed decision-making.

Definition of the logistic function

An explanation of logistic regression can begin with an explanation of the standard logistic function. The logistic function is a sigmoid function, which takes any real input, and outputs a value between zero and one.[2] For the logit, this is interpreted as taking input log-odds and having output probability. The standard logistic function

$$\boldsymbol{\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}}$$

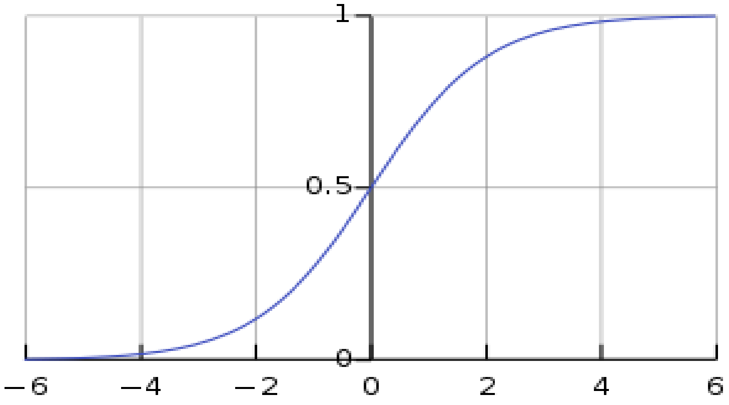


Figure 1. The standard logistic function $\boldsymbol{\sigma(t);\ \sigma(t) \in (0, 1)\ f}$or all

Assumptions of Logistic Regression:

Binary Outcome: Assumes the dependent variable has two outcomes.

Independence of Observations: Assumes observations are independent.

Linearity of Independent Variables and Log Odds: Assumes a linear relationship between independent variables and the log odds of the outcome.

No Multicollinearity: Assumes no high correlation among independent variables.

Large Sample Size: Performs better with larger sample sizes.

No Outliers: Sensitive to outliers, which can distort coefficients.

Linearity of Log Odds and Independent Variables: Assumes linearity between independent variables and log odds.

Correct Model Specification: Assumes all relevant variables are included.

Advantages:

Simple and Interpretable: Easy to understand and interpret.

Efficient with Small Datasets: Performs well with limited data.

Robust to Noise: Handles noise and irrelevant features.

Probabilistic Predictions: Provides clear likelihood indications.

Handles Numeric and Categorical Data: Versatile for various data types.

Disadvantages:

Assumes Linearity: Assumes linear relationship between variables.

Limited to Binary or Ordinal Outcomes: Primarily for two or three categories.

Sensitive to Outliers: Outliers can impact coefficients and performance.

Limited Capture of Complex Relationships: May not handle nonlinear relationships well.

Requires Independent Observations: Assumes independence among data points.

Logistic regression types can be categorized based on the number of categories in the dependent variable. Here are the main types:

1. Binary Logistic Regression:

This type is used when the dependent variable has two categories, typically coded as 0 and 1. Examples include yes/no, success/failure, or presence/absence.

2. Multinomial Logistic Regression:

Multinomial logistic regression is employed when the dependent variable has three or more unordered categories. Each category is compared to a reference category. Examples include predicting the type of disease (e.g., diabetes, hypertension, or obesity) or the preferred mode of transportation (e.g., car, bus, or bike).

3. Ordinal Logistic Regression:

Ordinal logistic regression is utilized when the dependent variable has three or more ordered categories. The categories have a logical order, but the intervals between them are not necessarily equal. Examples include Likert scale responses (e.g., strongly agree, agree, neutral, disagree, strongly disagree) or education level (e.g., high school, bachelor's degree, master's degree, Ph.D.).

4. Count Data Logistic Regression:

This type is used when the dependent variable represents counts of events or occurrences. Examples include the number of doctor visits, the number of accidents, or the number of purchases. Count data logistic regression models are often applied when the counts are over-dispersed or have excess zeros.

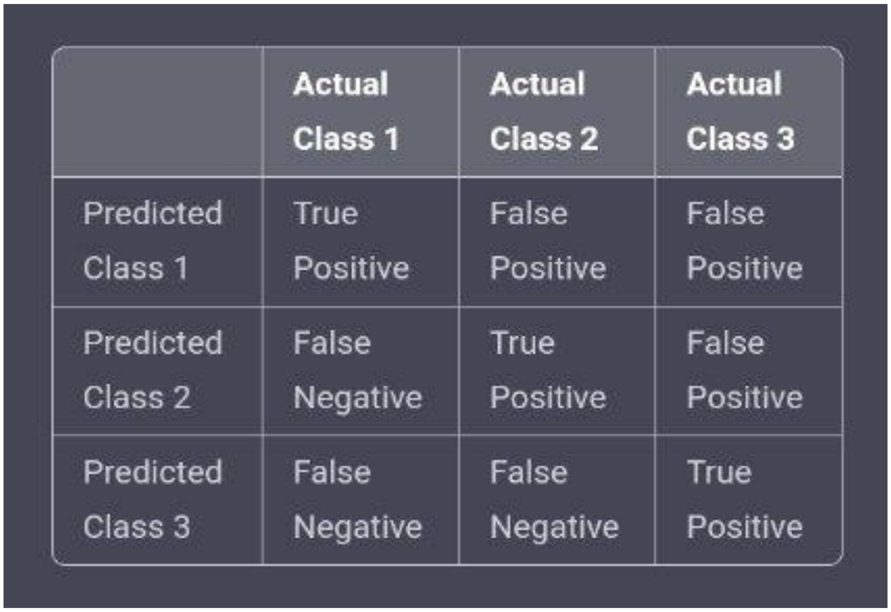
These are the logistic regression types based on the number of categories in the dependent variable. Each type has its own assumptions, considerations, and interpretation methods, tailored to the specific nature of the data and research questions.

Multinomial logistic regression aligns with the research objectives of accurately classifying individuals into distinct categories, such as diabetic, non-diabetic, or pre-diabetic, based on clinical and biochemical parameters. By leveraging multinomial logistic regression, this study aims to provide comprehensive insights into the classification of patients and contribute to the understanding of diabetes risk assessment and management strategies."

Evaluation Metrics for Multinomial Regression: Multinomial regression, unlike regular regression, deals with categorical outcomes with more than two classes. Evaluating these models requires metrics suited for classification tasks.

1. Confusion Matrix:

- It is used for the optimization of machine learning models.
- The confusion matrix is a N x N matrix, where N is the number of classes or outputs.
- For 2 classes, we get a 2 x 2 confusion matrix.
- For 3 classes, we get a 3 X 3 confusion matrix.



	Actual Class 1	Actual Class 2	Actual Class 3
Predicted Class 1	True Positive	False Positive	False Positive
Predicted Class 2	False Negative	True Positive	False Positive
Predicted Class 3	False Negative	False Negative	True Positive

- True Positives (TP): Correctly predicted instances for a class.
- True Negatives (TN): Correctly predicted instances that don't belong to the class.
- False Positives (FP): Instances incorrectly predicted as belonging to the class (Type I error).
- False Negatives (FN): Instances belonging to the class but predicted as not (Type II error).

The confusion matrix helps identify areas for improvement, like high FN for a specific class indicating under-predictio**n.**

2. ROC Curve (Receiver Operating Characteristic Curve):

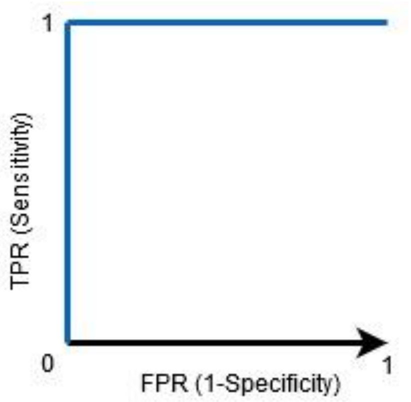
An ROC curve, or receiver operating characteristic curve, is like a graph that shows how well a classification model performs. It helps us see how the model makes decisions at different levels of certainty. The curve has two lines: one for how often the model correctly identifies positive cases (true positives) and another for how often it mistakenly identifies negative cases as positive (false positives). By looking at this graph, we can understand how good the model is and choose the threshold that gives us the right balance between correct and incorrect predictions.

The **Receiver Operator Characteristic (ROC)** curve is an evaluation metric for binary classification problems. It is a probability curve that plots the **TPR** against **FPR** at various threshold values and essentially **separates the 'signal' from the 'noise.'** In other words, it shows the performance of a classification model at all classification thresholds.

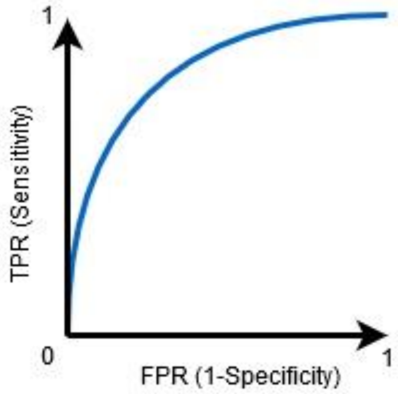
3. Area Under the ROC Curve (AUC-ROC):

The **Area Under the Curve (AUC)** is the measure of the ability of a binary classifier to distinguish between classes and is used as a summary of the ROC curve.

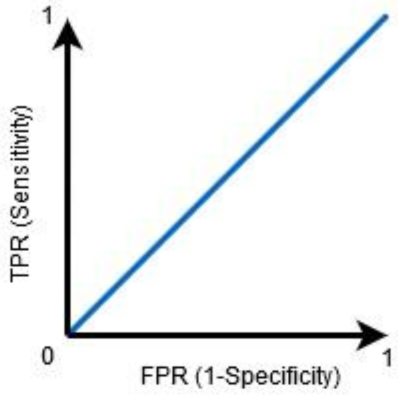
The higher the AUC, the better the model's performance at distinguishing between the positive and negative classes.



When AUC = 1, the classifier can correctly distinguish between all the Positive and the Negative class points. If, however, the AUC had been 0, then the classifier would predict all Negatives as Positives and all Positives as Negatives.



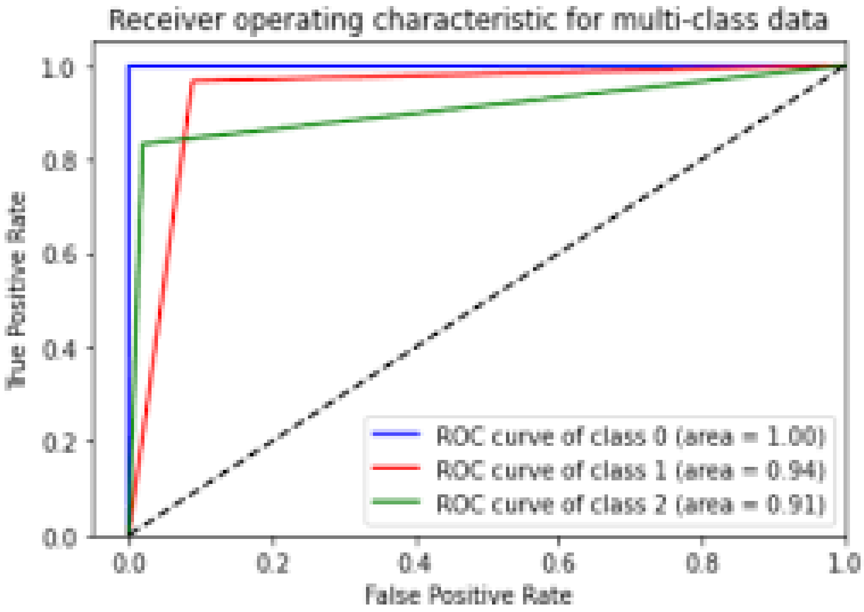
When 0.5<AUC<1, there is a high chance that the classifier will be able to distinguish the positive class values from the negative ones. This is so because the classifier is able to detect more numbers of True positives and True negatives than False negatives and False positives.



When AUC=0.5, then the classifier is not able to distinguish between Positive and Negative class points. Meaning that the classifier either predicts a random class or a constant class for all the data points.

the AUC-ROC curve is only for binary classification problems. But we can extend it to multiclass classification problems using the One vs. All technique.

So, if we have three classes, 0, 1, and 2, the ROC for class 0 will be generated as classifying 0 against not 0, i.e., 1 and 2. The ROC for class 1 will be generated as classifying 1 against not 1, and so on.



4. Precision, Recall, and F1-Score:

1. Sensitivity (True Positive Rate, Recall)

- Sensitivity measures the proportion of actual positive cases that are correctly identified by the model.
- It's calculated as the ratio of true positives (correctly predicted positive instances) to the sum of true positives and false negatives (positive instances incorrectly predicted as negative).

$$Sensitivity = \frac{TP}{TP + FN}$$

2. Specificity (True Negative Rate):

Specificity measures the proportion of actual negative cases that are correctly identified by the model. It’s calculated as the ratio of true negatives (correctly predicted negative instances) to the sum of true negatives and false positives (negative instances incorrectly predicted as positive).

$$Specificity = \frac{TN}{TN + FP}$$

3.Accuracy:

- Accuracy measures the overall correctness of the model across all classes.
- It's calculated as the ratio of correctly classified instances (both true positives and true negatives) to the total number of instances.

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

4. F1 Score:

- F1 Score is the harmonic mean of precision and recall. It provides a balance between precision and recall.
- It's calculated as 2 times the product of precision and recall divided by the sum of precision and recall

$$\frac{2 * (Precision * Recall)}{(Precision + Recall)}$$

For a multinomial logistic regression model, these metrics would be calculated considering multiple classes. Sensitivity, specificity, accuracy, and F1 score can be calculated for each class individually (using one-vs-all approach) or can be calculated as a macro or micro average across all classes.

Macro Average:

Computes the metric independently for each class and then takes the average, treating all classes equally .Useful when each class is considered equally important and you want to assess the model's generalization across all classes.

Micro Average:

First aggregates the contributions of all classes and then computes the metric, effectively treating each instance equally. Particularly beneficial when classes are imbalanced, as it puts more weight on the performance of dominant classes, providing a more balanced view of the model's effectiveness.

Z-Scores and Standardization:

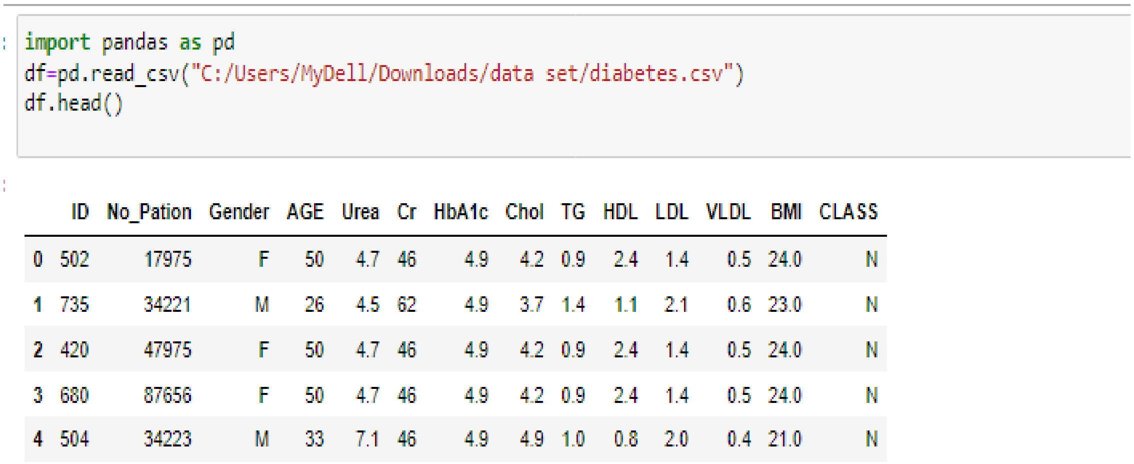
A z-score, also known as a standard score, represents how many standard deviations a specific data point is away from the mean of the data set. It essentially standardizes the data by converting the original values into units of standard deviations. This allows for comparison across different features with varying scales.

CHAPTER-4

DATA ANALYSIS

Data import in Python

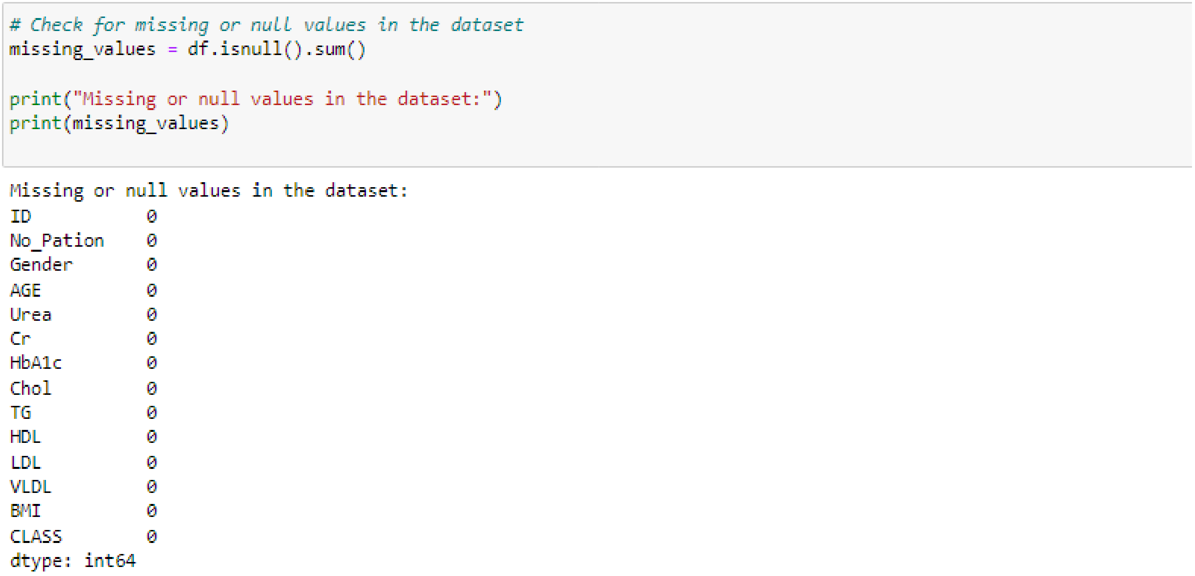
Data Analysis was carried out jupyter notebook using python. The following steps were implemented in order to process to build the logistic regression



```
import pandas as pd
df=pd.read_csv("C:/Users/MyDell/Downloads/data set/diabetes.csv")
df.head()
```

	ID	No_Pation	Gender	AGE	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	BMI	CLASS
0	502	17975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
1	735	34221	M	26	4.5	62	4.9	3.7	1.4	1.1	2.1	0.6	23.0	N
2	420	47975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
3	680	87656	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
4	504	34223	M	33	7.1	46	4.9	4.9	1.0	0.8	2.0	0.4	21.0	N

MISSING VALUES IDENTIFICATION:



```
# Check for missing or null values in the dataset
missing_values = df.isnull().sum()

print("Missing or null values in the dataset:")
print(missing_values)
```

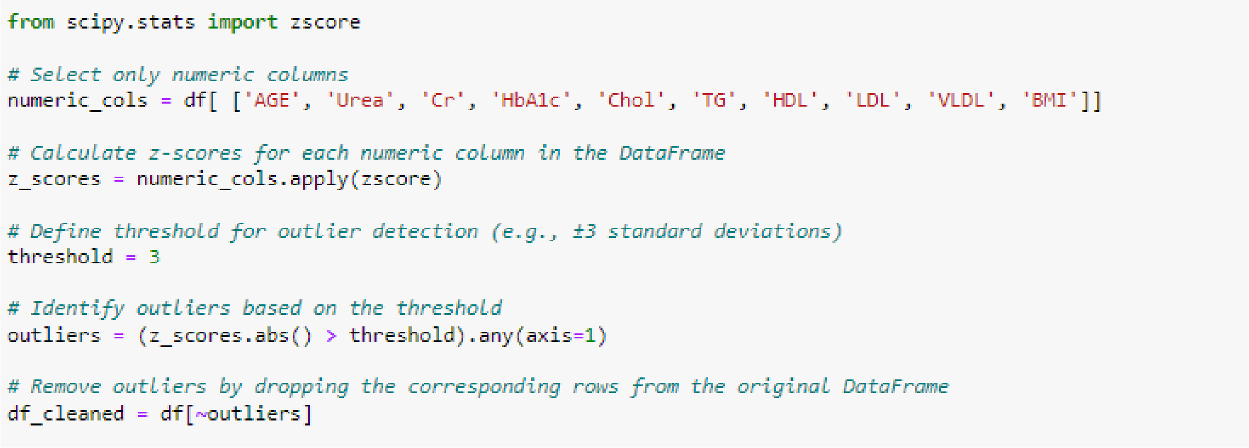
```
Missing or null values in the dataset:
ID           0
No_Pation    0
Gender       0
AGE          0
Urea         0
Cr           0
HbA1c        0
Chol         0
TG           0
HDL          0
LDL          0
VLDL         0
BMI          0
CLASS        0
dtype: int64
```

INTERPRETATION:

It indicates that the there is no missing value in the dataset containing 14 variables, the missing values is 0,so the dataset containing zero missing values

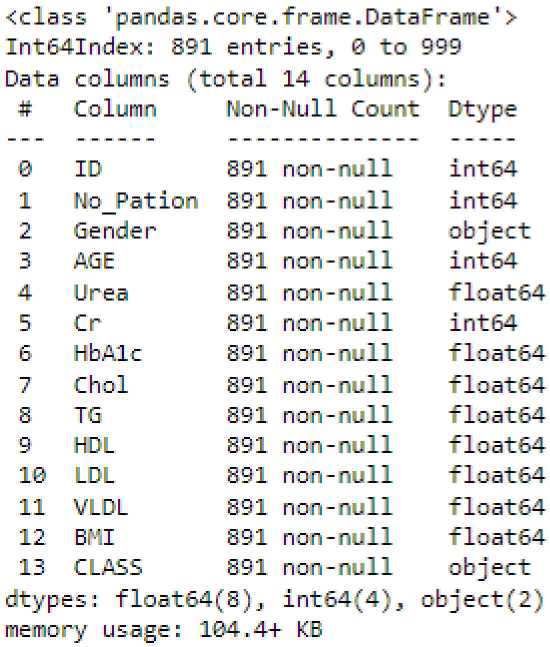
OUTLIERS TREATMENT :

Outliers from dataset (1000,14) were founded Using the Z-score the code for detect the outliers is given below:



```
from scipy.stats import zscore

# Select only numeric columns
numeric_cols = df[ ['AGE', 'Urea', 'Cr', 'HbA1c', 'Chol', 'TG', 'HDL', 'LDL', 'VLDL', 'BMI']]

# Calculate z-scores for each numeric column in the DataFrame
z_scores = numeric_cols.apply(zscore)

# Define threshold for outlier detection (e.g., ±3 standard deviations)
threshold = 3

# Identify outliers based on the threshold
outliers = (z_scores.abs() > threshold).any(axis=1)

# Remove outliers by dropping the corresponding rows from the original DataFrame
df_cleaned = df[~outliers]
```

The dataset samples after removal of the outliers is shown below



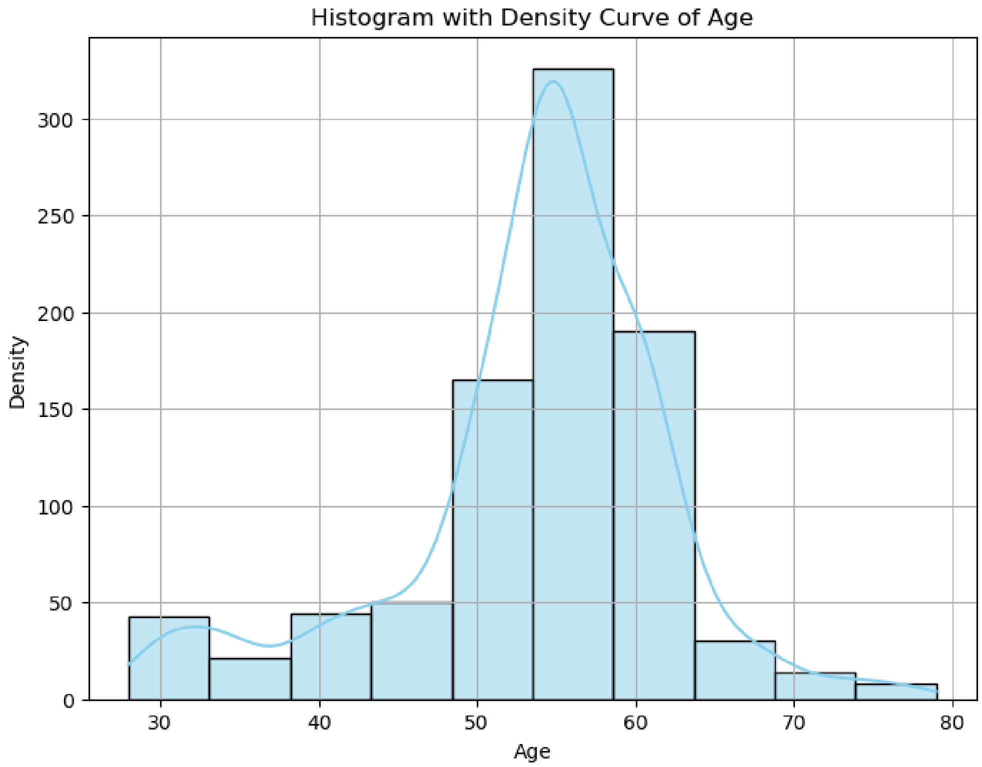
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 0 to 999
Data columns (total 14 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   ID         891 non-null    int64  
 1   No_Pation  891 non-null    int64  
 2   Gender     891 non-null    object 
 3   AGE        891 non-null    int64  
 4   Urea       891 non-null    float64
 5   Cr         891 non-null    int64  
 6   HbA1c      891 non-null    float64
 7   Chol       891 non-null    float64
 8   TG         891 non-null    float64
 9   HDL        891 non-null    float64
 10  LDL        891 non-null    float64
 11  VLDL       891 non-null    float64
 12  BMI        891 non-null    float64
 13  CLASS      891 non-null    object 
dtypes: float64(8), int64(4), object(2)
memory usage: 104.4+ KB
```

INTERPRETATION:

- Z-scores provide an intuitive understanding of how far a data point deviates from the mean in terms of standard deviations.
- Establishing a threshold (e.g., ±3) allows for systematic outlier detection by focusing on z-scores exceeding that threshold.

UNIVARIATE ANALYSIS :

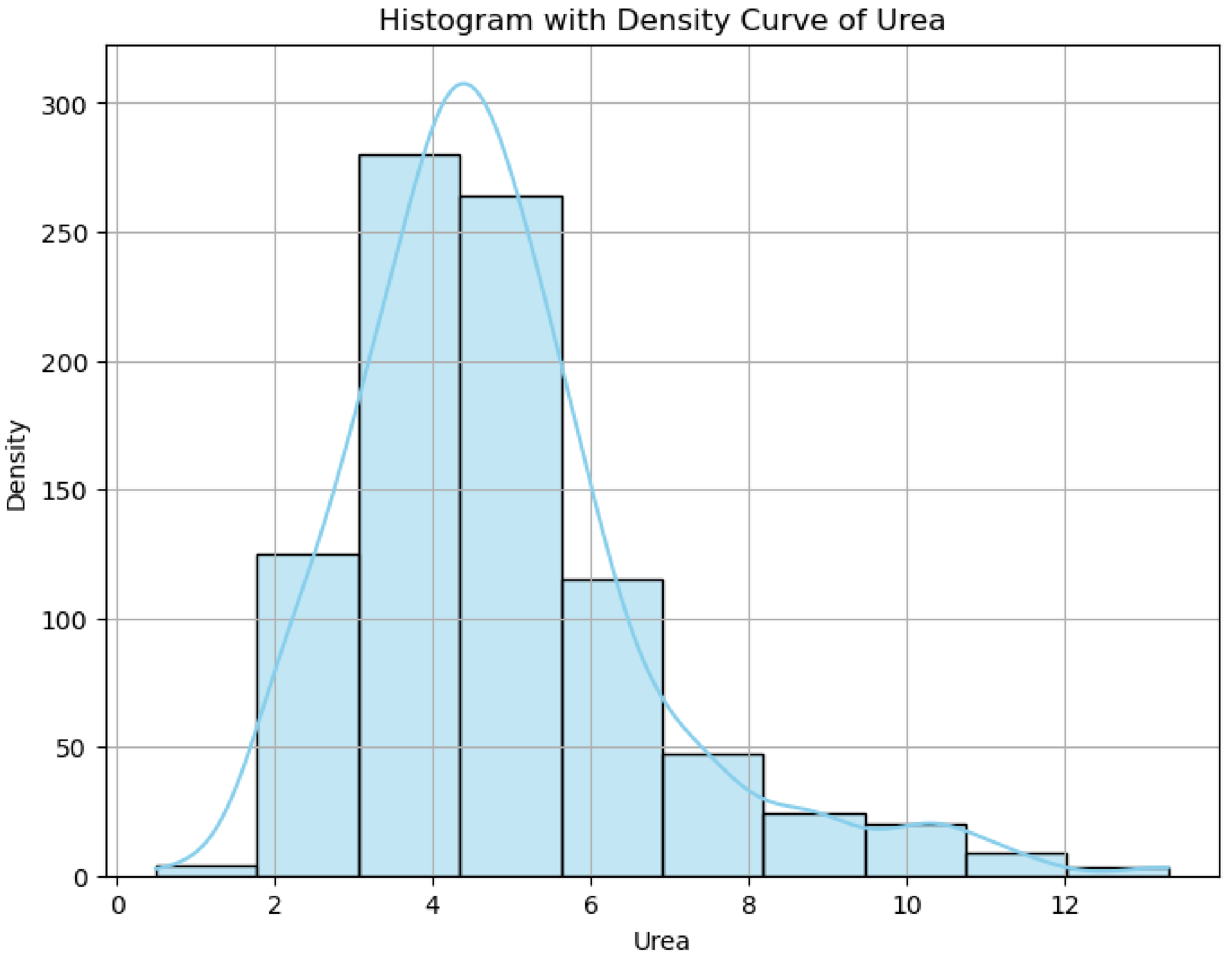
AGE VARIABLE



INTERPRETATION:

The age distribution after removing outliers appears close to a normal curve. This suggests a more focused analysis on the typical age range for diabetes prediction in our data. The central tendency (peak of the curve) represents the most frequent age group, which might be more relevant for understanding the relationship between age and diabetes risk in this dataset.

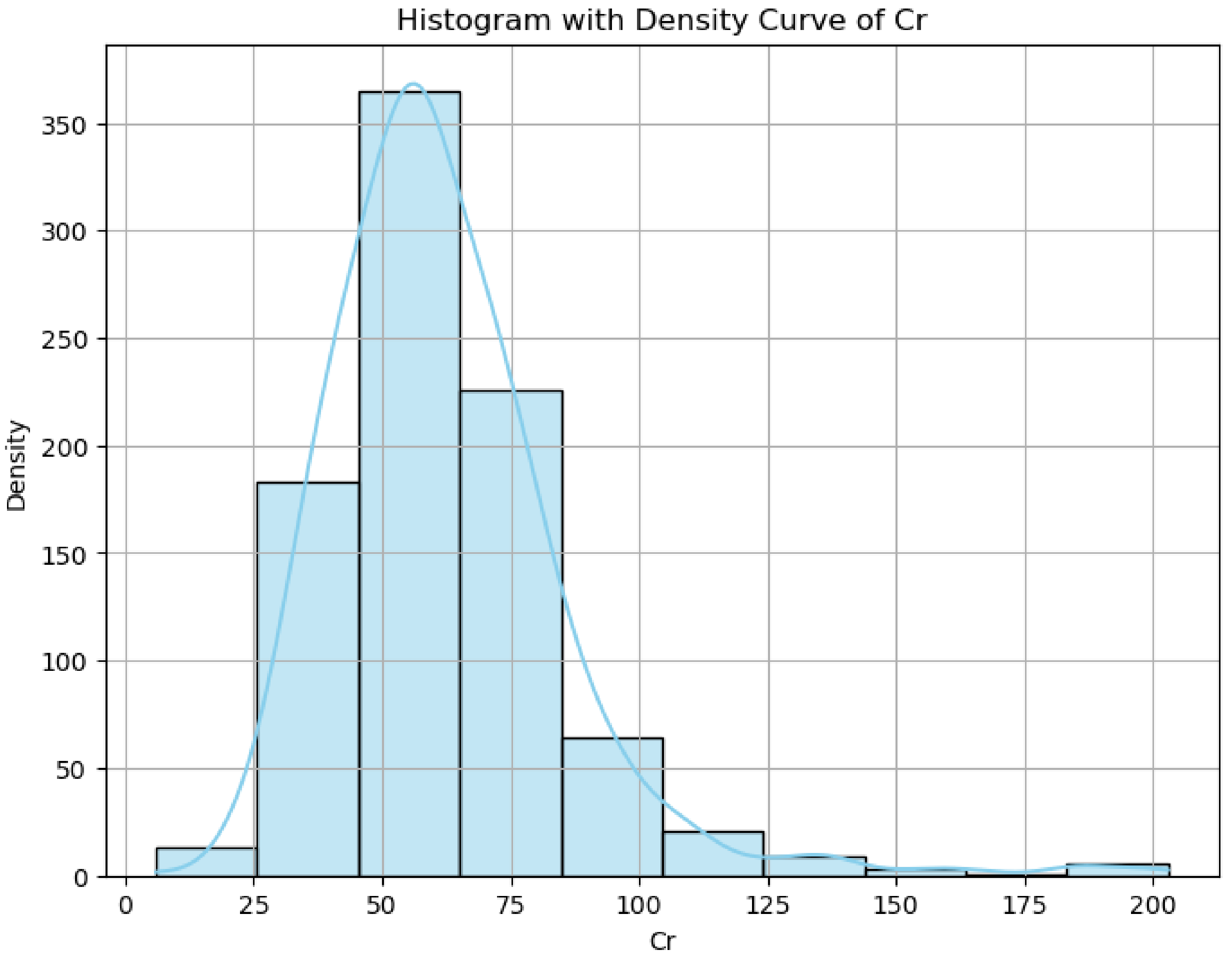
UREA VARIABLE:



INTERPRETATION:

The distribution of Urea after outlier removal appears close to normal. The central tendency (peak) represents the most frequent Urea level in the data. This might be useful for understanding the typical Urea range in our diabetes prediction analysis.

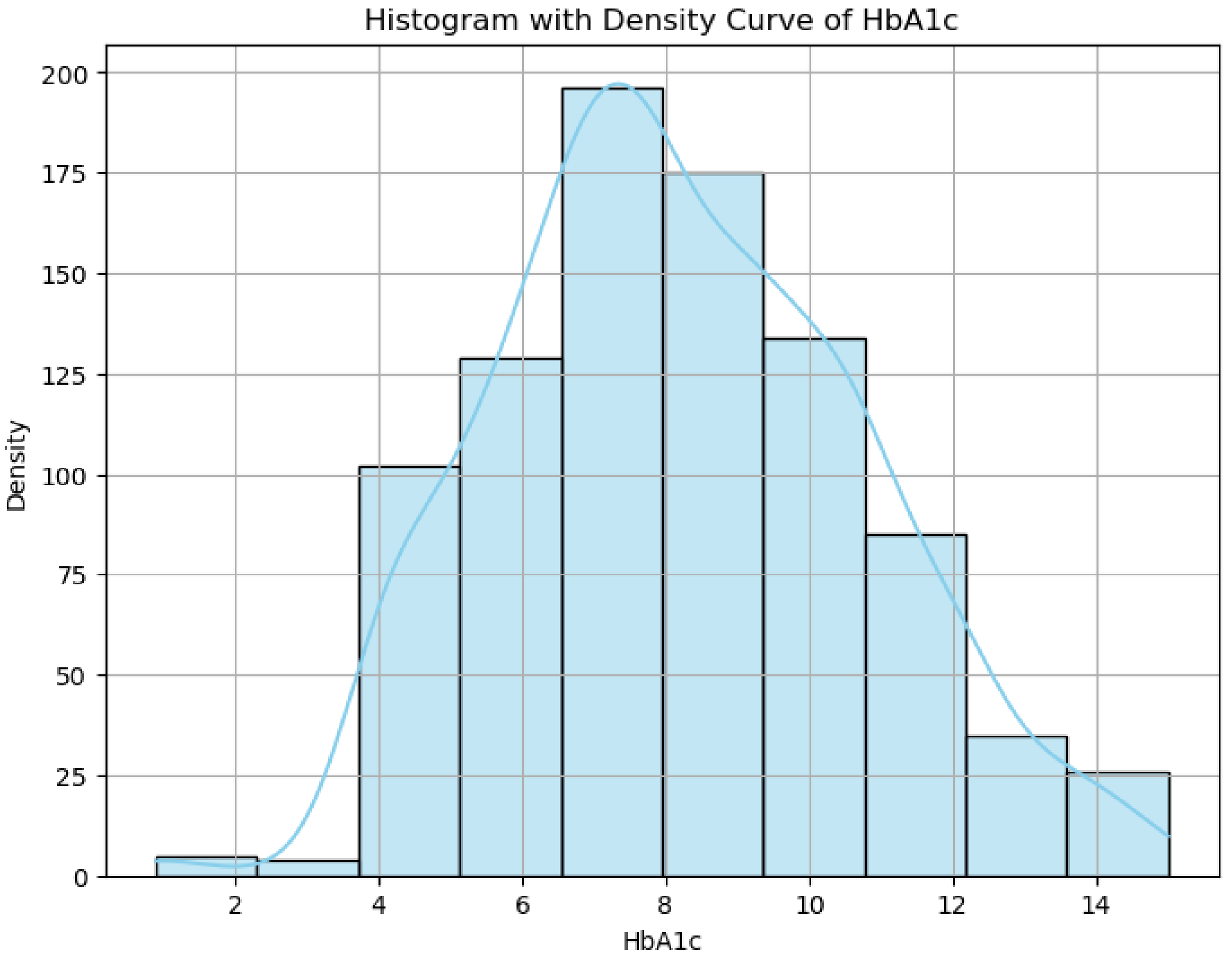
CRETININE VARIABLE:



INTERPRETATION:

This histogram depicts the distribution of Creatinine levels after removing outliers using z-scores. The overall shape of the distribution appears close to normal. This suggests a more focused analysis on the typical Creatinine range in the data after eliminating extreme values.

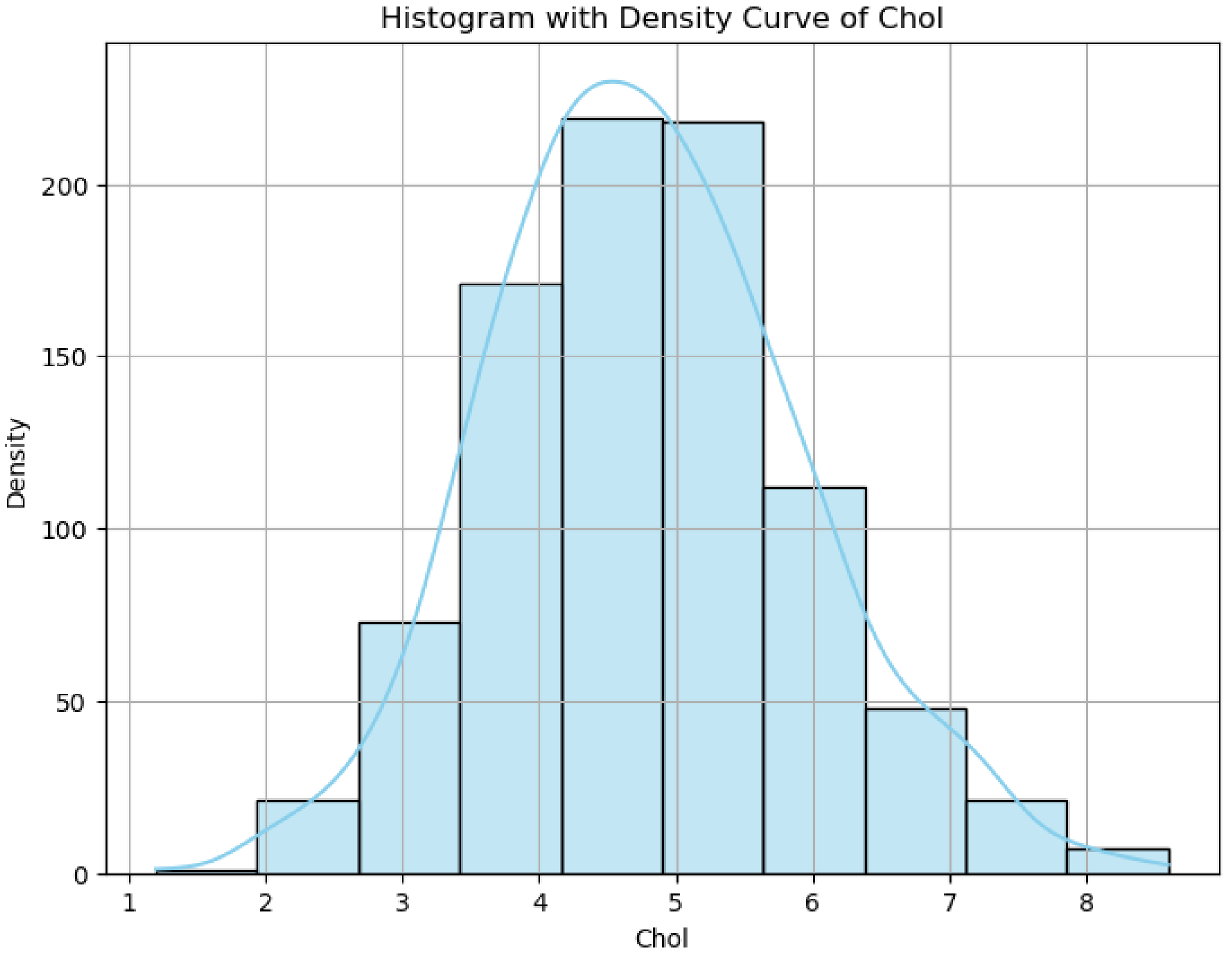
GLYCATED HEMOGLOBIN VARIABLE :



INTERPRETATION:

The distribution of HbA1c leans slightly to the right, indicating a possible right skew. This suggests there might be more data points concentrated towards lower HbA1c values (better glycemic control) compared to higher values. The peak of the density curve is around 5.8%, which represents the most frequent HbA1c value in the data after removing outliers. Lower HbA1c values generally indicate better blood sugar control.

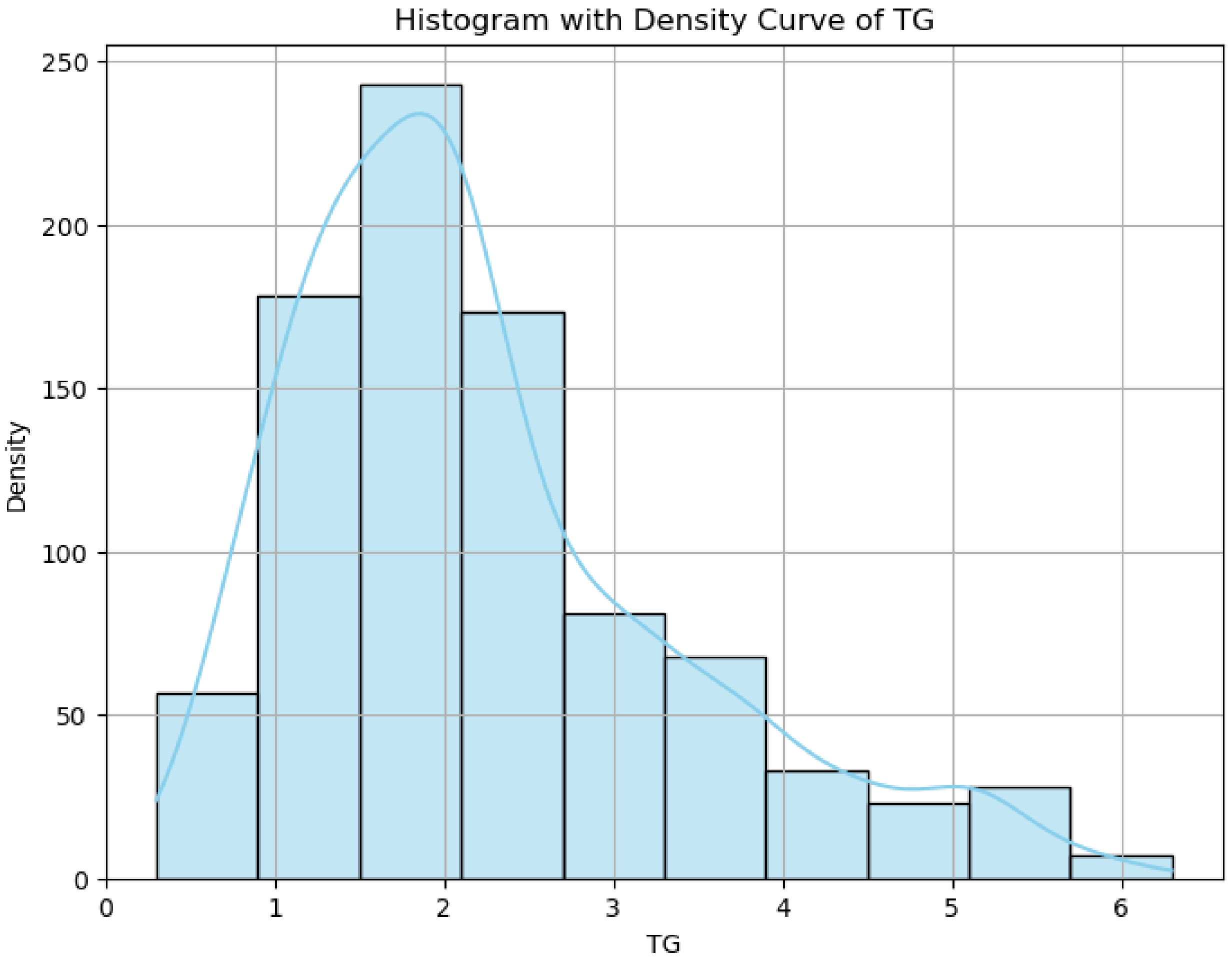
CHOLESTEROL VARIABLE :



INTERPRETATION:

The overall shape of the cholesterol distribution leans slightly to the right, indicating a possible right skew. This suggests that there might be more data points concentrated towards lower cholesterol levels compared to higher values after removing outliers.

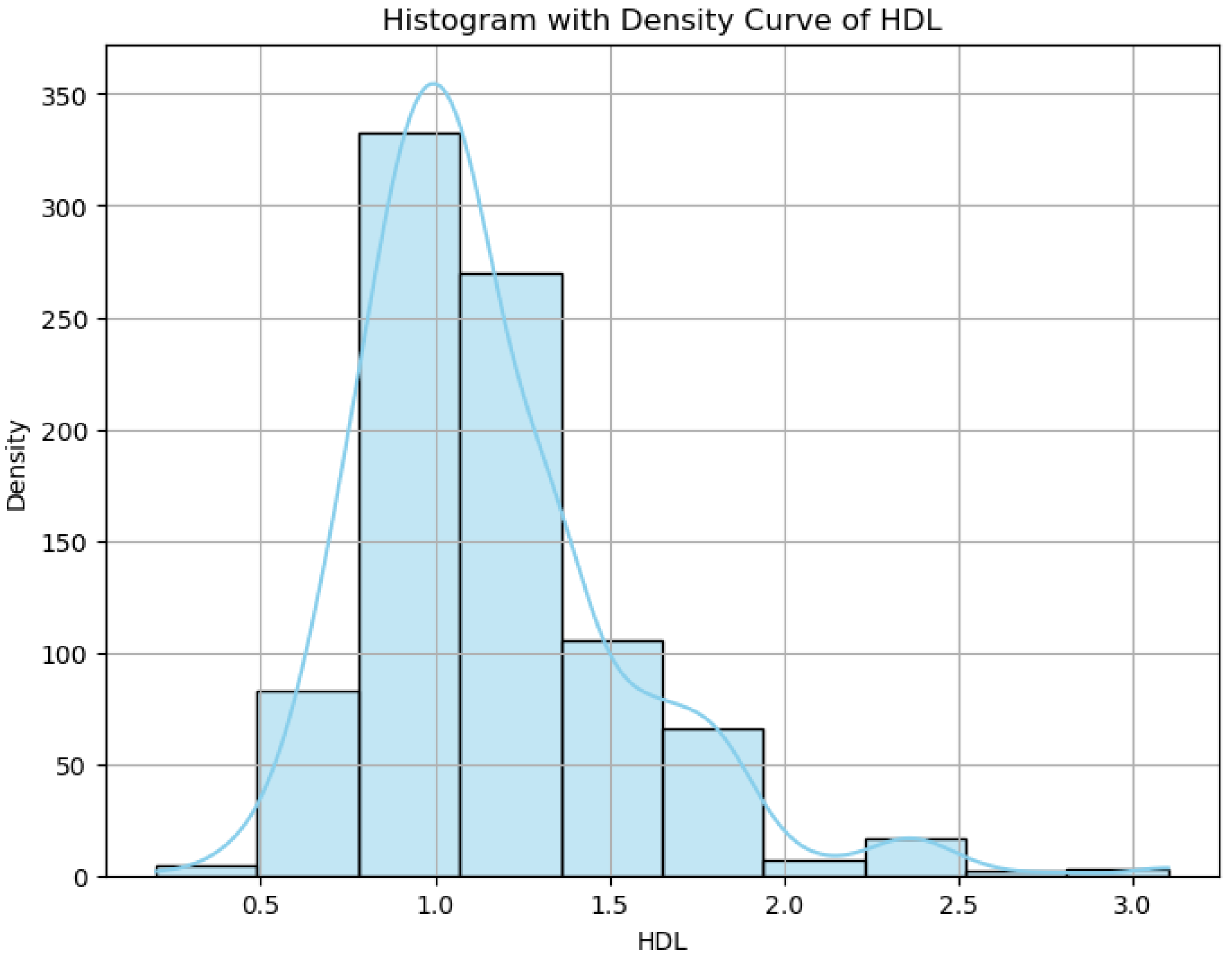
TRIGLYCERIDES VARIABLE :



INTERPRETATION:

The distribution appears right-skewed, with a longer tail extending towards higher triglyceride (TG) levels. This suggests that there might be more data points concentrated towards lower TG values compared to higher values after outlier removal.

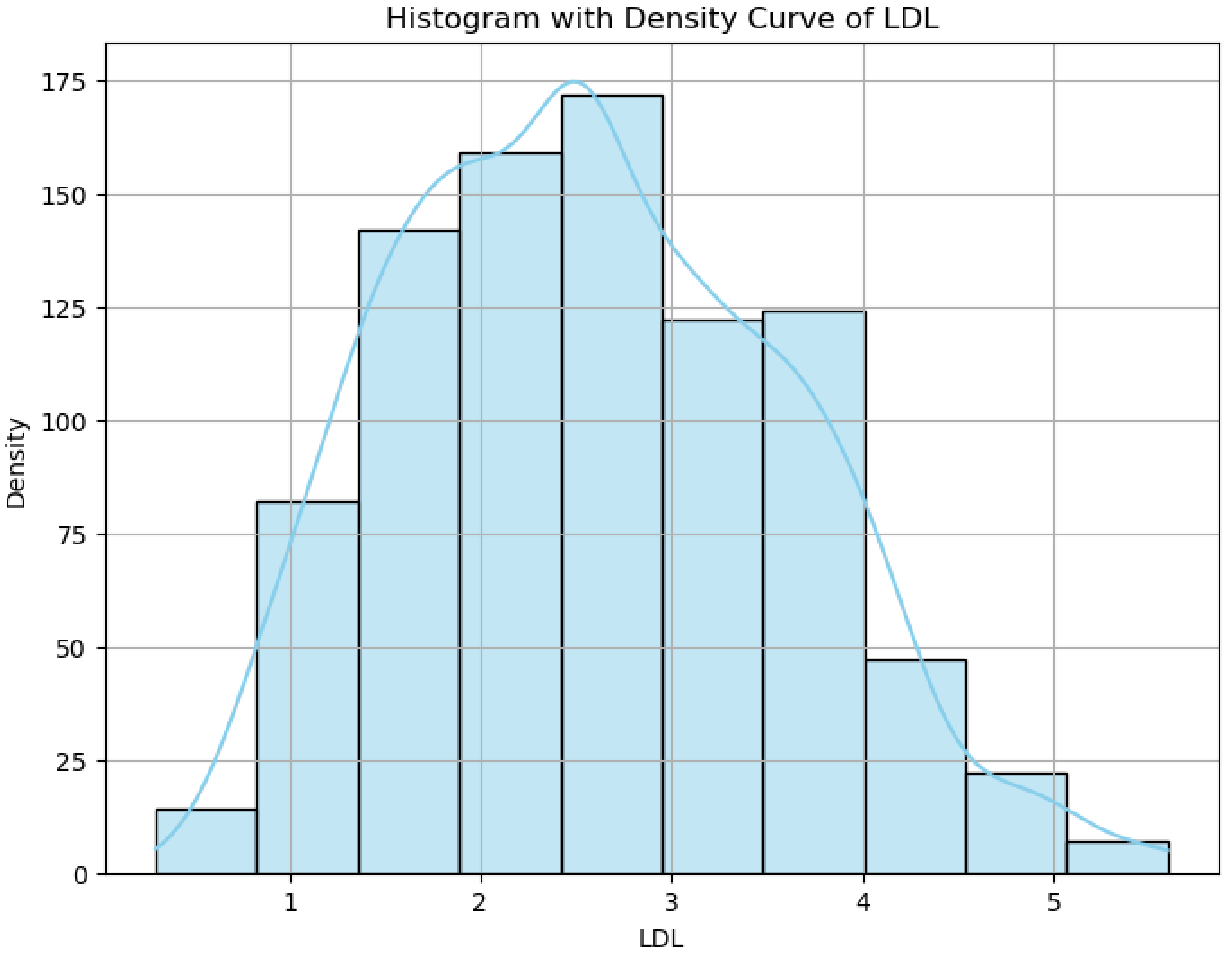
HIGH-DENSITY LIPOPROTEIN (HDL) :



INTERPRETATION:

The histogram and the density curve is plotted for the High density lipoprotein .The distribution of HDL cholesterol levels after outlier removal appears close to normal

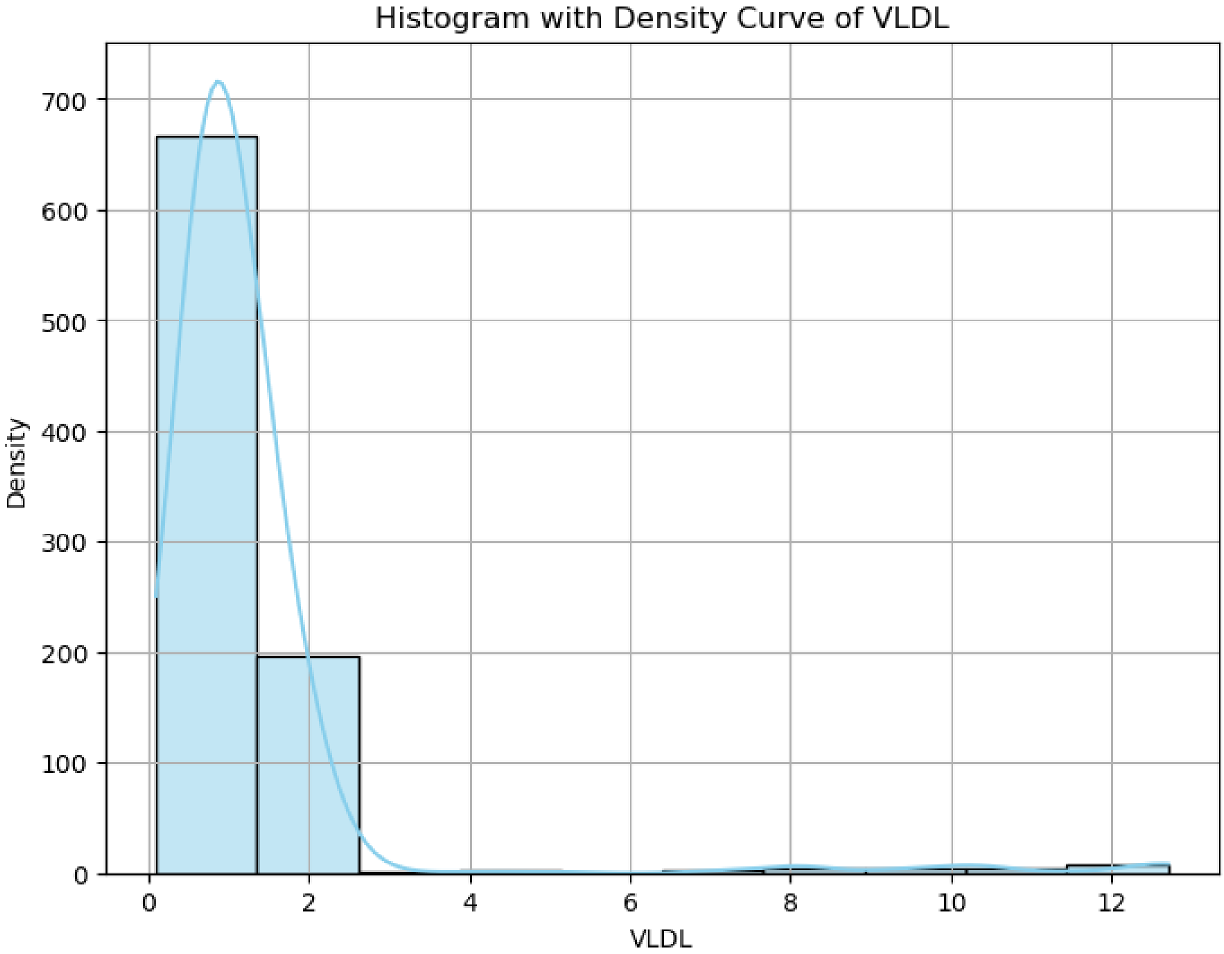
LOW DENSITY LIPOPROTEIN (LDL)   VARIABLE:



INTERPRETATION:

This analysis shows a right-skewed distribution of LDL cholesterol levels after outlier removal.

VERY LOW DENSITY LIPOPROTEIN (HDL)

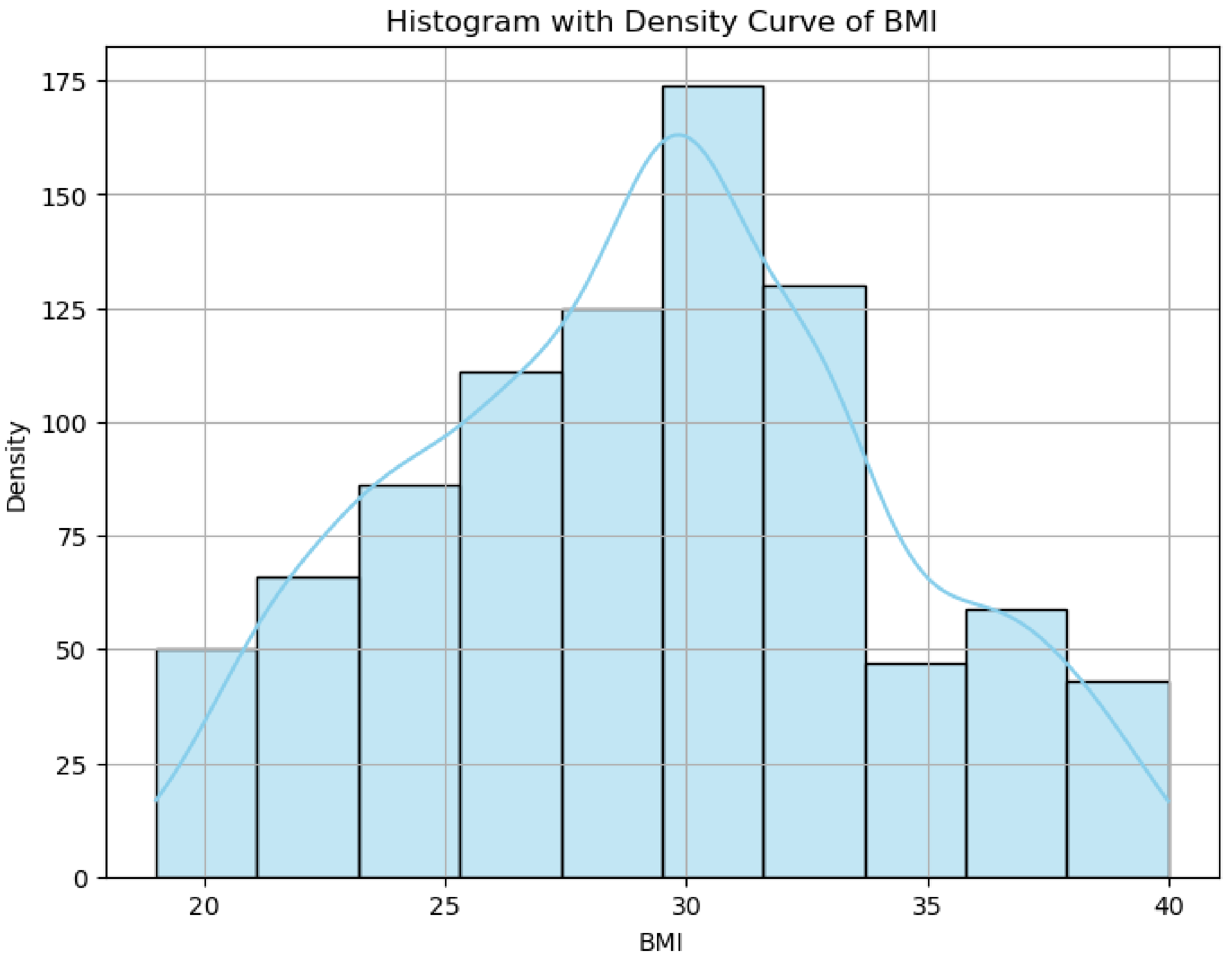


INTERPRETATION:

This analysis shows a right-skewed distribution of VLDL cholesterol levels after outlier removal.

The most frequent VLDL level is around 20 mg/dL.

BODY MASS INDEX(BMI) VARIABLE :

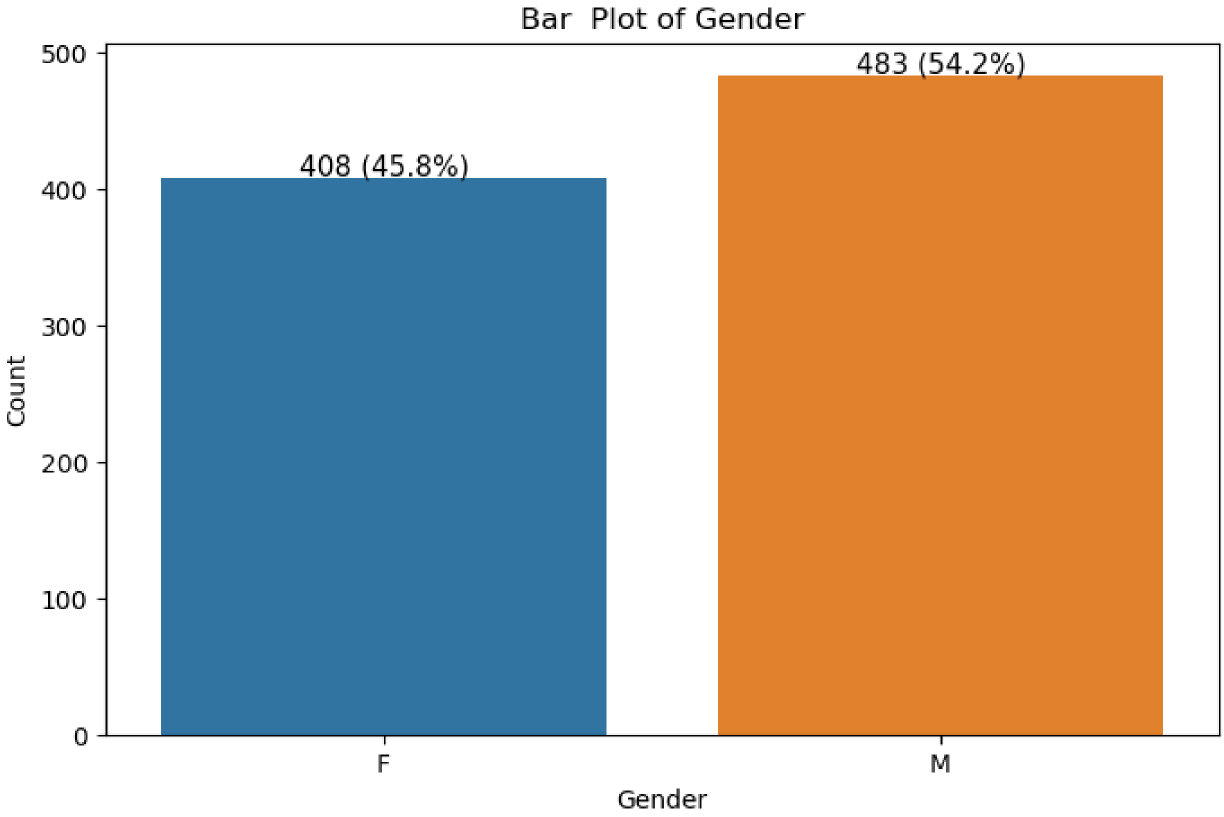


INTERPRETATION:

- The distribution leans slightly to the right, indicating a possible right skew. This suggests there might be more data points concentrated towards lower BMI values (possibly normal weight range) compared to higher values after outlier removal.

UNIVARIATE ANALYSIS FOR CATEGORICAL VARIABLES :

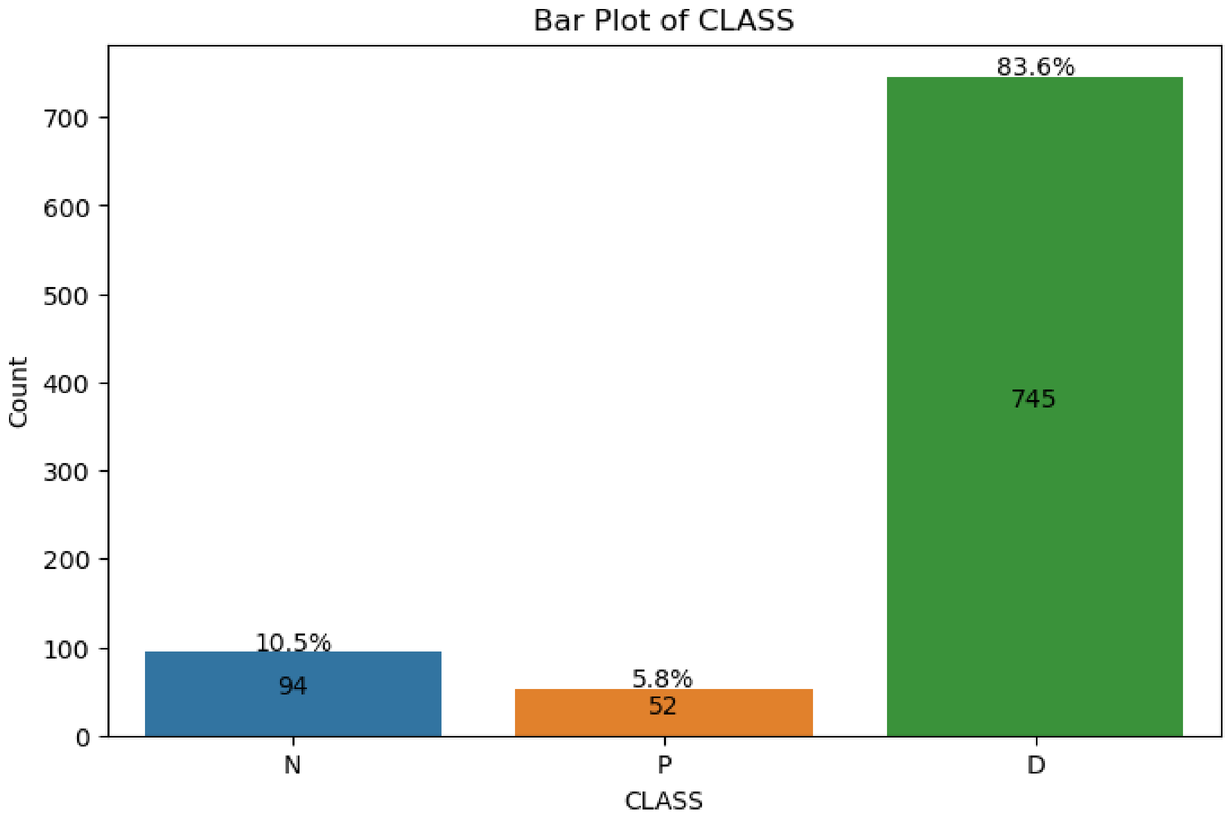
GENDER VARIABLE



INTERPRETATION:

From the above graph it denotes that the dataset containing three types of gender male, female, and other. There are 483 males in the dataset and 408 females in the dataset. There is only one other in the dataset.

CLASS VARIABLE:



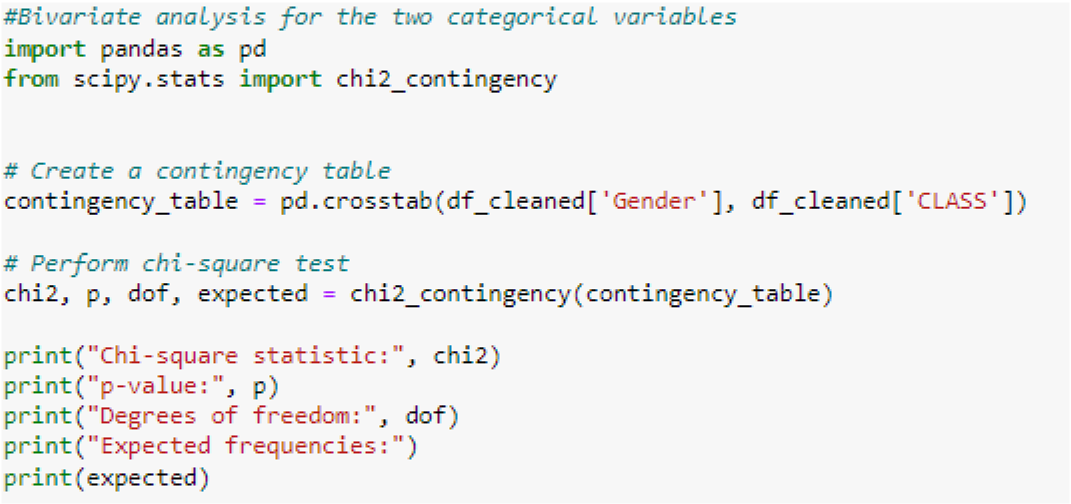
INTERPRETATION:

The above first bar plot shows that count of stroke patients in the dataset. There are 94 Non Diabetic patients and 52 prediabetic patient and the 745 Diabetic patients that is approximately 83.6 % patients are not affected by the diabetes ,5.8% are pre diabetic patients and 10.5 % were Non Diabetic

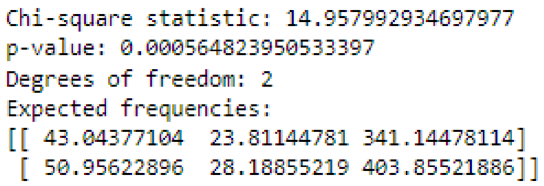
BIVARIATE ANALYSIS FOR TWO CATEGORICAL VARIABLES(GENDER AND THE CLASS VARIABLES):

H0: There is no significant Association between the Gender and the class variable

H1: There is a significant Association between the Gender and the Class variable



```
#Bivariate analysis for the two categorical variables
import pandas as pd
from scipy.stats import chi2_contingency


# Create a contingency table
contingency_table = pd.crosstab(df_cleaned['Gender'], df_cleaned['CLASS'])

# Perform chi-square test
chi2, p, dof, expected = chi2_contingency(contingency_table)

print("Chi-square statistic:", chi2)
print("p-value:", p)
print("Degrees of freedom:", dof)
print("Expected frequencies:")
print(expected)
```



```
Chi-square statistic: 14.957992934697977
p-value: 0.000564823950533397
Degrees of freedom: 2
Expected frequencies:
[[ 43.04377104  23.81144781 341.14478114]
 [ 50.95622896  28.18855219 403.85521886]]
```

INTERPRETATION:

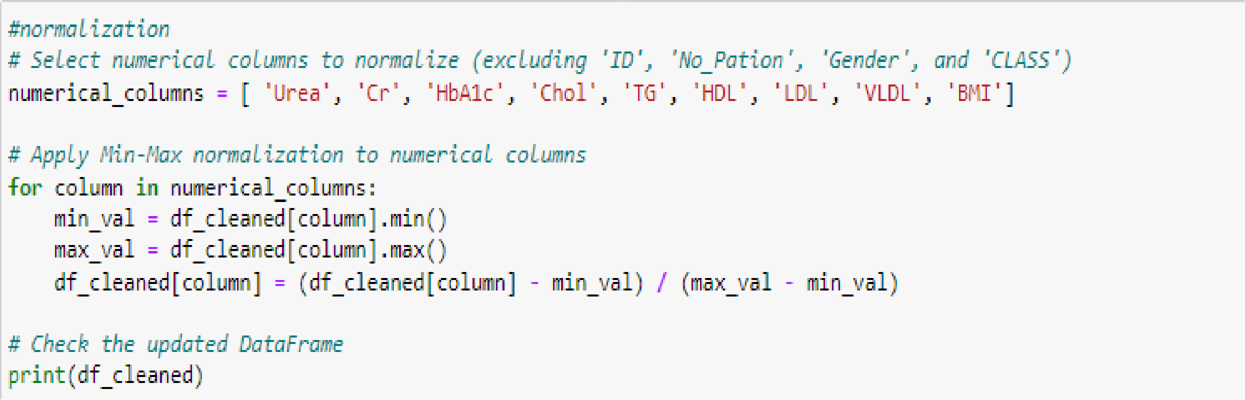
Based on the chi-square test results, we can conclude that there's a statistically significant association between gender and the "CLASS" variable in your data at a significance level of 0.05. This means that the distribution of individuals across the "CLASS" categories is not independent of their gender.

NORMALIZATION:

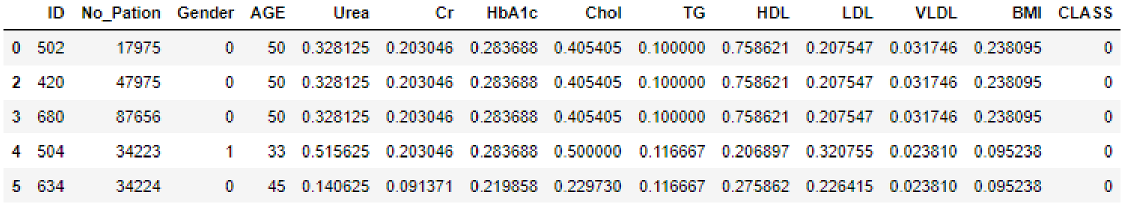
Normalization, in the context of data analysis, refers to the process of rescaling numerical features within a dataset to a common range. This is often done between 0 and 1 (Min-Max normalization) or to have a standard deviation of 1 (Z-score normalization).

Formula

$$N(x_{ij}) = \frac{x_{ij} - x_{jmin}}{x_{jmax} - x_{jmin}}$$



```python
#normalization
# Select numerical columns to normalize (excluding 'ID', 'No_Pation', 'Gender', and 'CLASS')
numerical_columns = [ 'Urea', 'Cr', 'HbA1c', 'Chol', 'TG', 'HDL', 'LDL', 'VLDL', 'BMI']

# Apply Min-Max normalization to numerical columns
for column in numerical_columns:
    min_val = df_cleaned[column].min()
    max_val = df_cleaned[column].max()
    df_cleaned[column] = (df_cleaned[column] - min_val) / (max_val - min_val)

# Check the updated DataFrame
print(df_cleaned)
```



	ID	No_Pation	Gender	AGE	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	BMI	CLASS
0	502	17975	0	50	0.328125	0.203046	0.283688	0.405405	0.100000	0.758621	0.207547	0.031746	0.238095	0
2	420	47975	0	50	0.328125	0.203046	0.283688	0.405405	0.100000	0.758621	0.207547	0.031746	0.238095	0
3	680	87656	0	50	0.328125	0.203046	0.283688	0.405405	0.100000	0.758621	0.207547	0.031746	0.238095	0
4	504	34223	1	33	0.515625	0.203046	0.283688	0.500000	0.116667	0.206897	0.320755	0.023810	0.095238	0
5	634	34224	0	45	0.140625	0.091371	0.219858	0.229730	0.116667	0.275862	0.226415	0.023810	0.095238	0

INTERPRETATION:

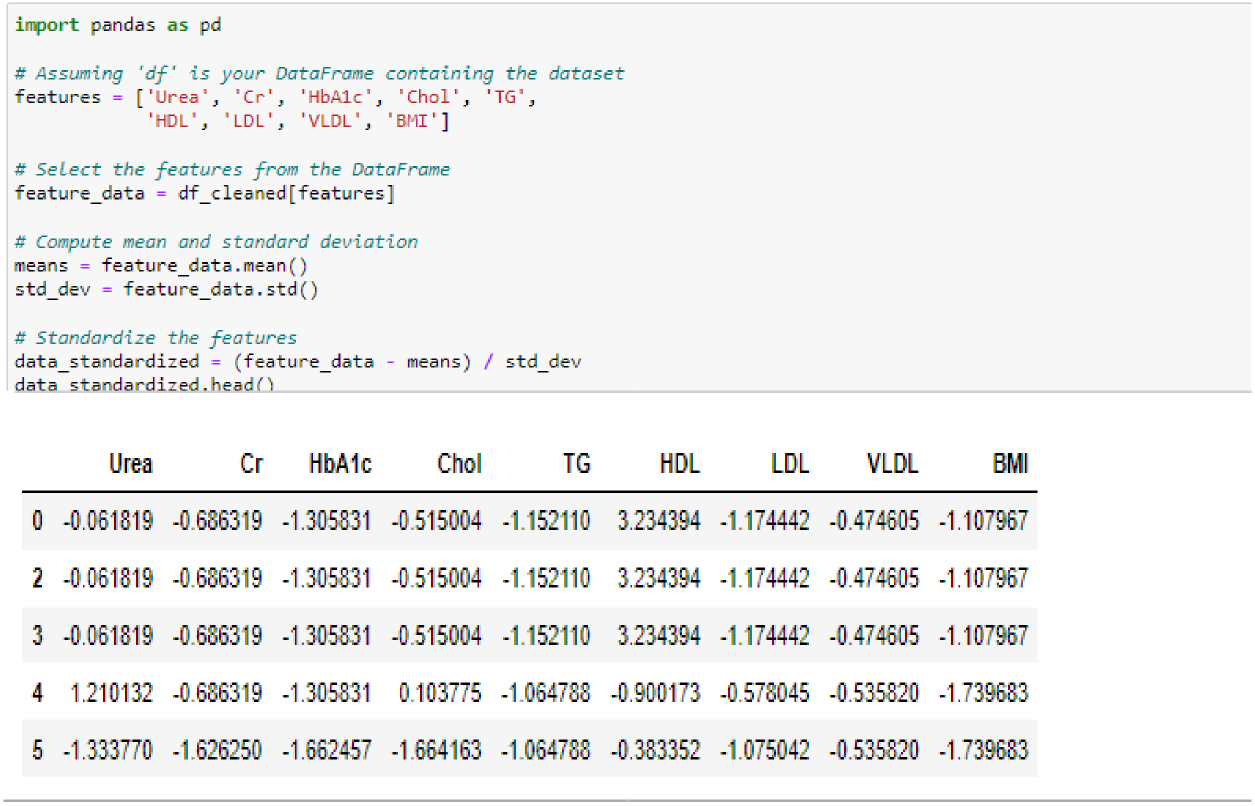
The original data points are transformed into a range of 0 to 1, making them more comparable for analysis, especially when dealing with features measured on different scales.

STANDARDIZATION:

Standardization, in the context of data analysis, refers to a technique for transforming your numerical data to have a standard normal distribution.

Formula

$$Z(x_{ij}) = \frac{x_{ij} - \overline{x}_j}{\sigma_j}$$



```
import pandas as pd

# Assuming 'df' is your DataFrame containing the dataset
features = ['Urea', 'Cr', 'HbA1c', 'Chol', 'TG',
            'HDL', 'LDL', 'VLDL', 'BMI']

# Select the features from the DataFrame
feature_data = df_cleaned[features]

# Compute mean and standard deviation
means = feature_data.mean()
std_dev = feature_data.std()

# Standardize the features
data_standardized = (feature_data - means) / std_dev
data_standardized.head()
```

	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	BMI
0	-0.061819	-0.686319	-1.305831	-0.515004	-1.152110	3.234394	-1.174442	-0.474605	-1.107967
2	-0.061819	-0.686319	-1.305831	-0.515004	-1.152110	3.234394	-1.174442	-0.474605	-1.107967
3	-0.061819	-0.686319	-1.305831	-0.515004	-1.152110	3.234394	-1.174442	-0.474605	-1.107967
4	1.210132	-0.686319	-1.305831	0.103775	-1.064788	-0.900173	-0.578045	-0.535820	-1.739683
5	-1.333770	-1.626250	-1.662457	-1.664163	-1.064788	-0.383352	-1.075042	-0.535820	-1.739683

INTERPRETATION:

Standardization transforms your data such that each feature has a mean of 0 and a standard deviation of 1. This makes the features more comparable by putting them on a "common ground" regardless of their original units or scales.

FEATURE SELECTION:

Reduce the dimensionality of the data by selecting a subset of the most informative features. This improves model training speed and reduces the risk of overfitting, where the model memorizes irrelevant details in the data and performs poorly on unseen data.

Benefits of using SelectKBest:

- **Faster Training Time:** By reducing the number of features, the model requires less computation time to train.
- **Improved Model Performance:** Focusing on relevant features can lead to better accuracy on unseen data.
- **Reduced Overfitting**: Fewer features help prevent the model from overfitting to the training data.

Selected features mention below:

Selected Feature Names:

AGE

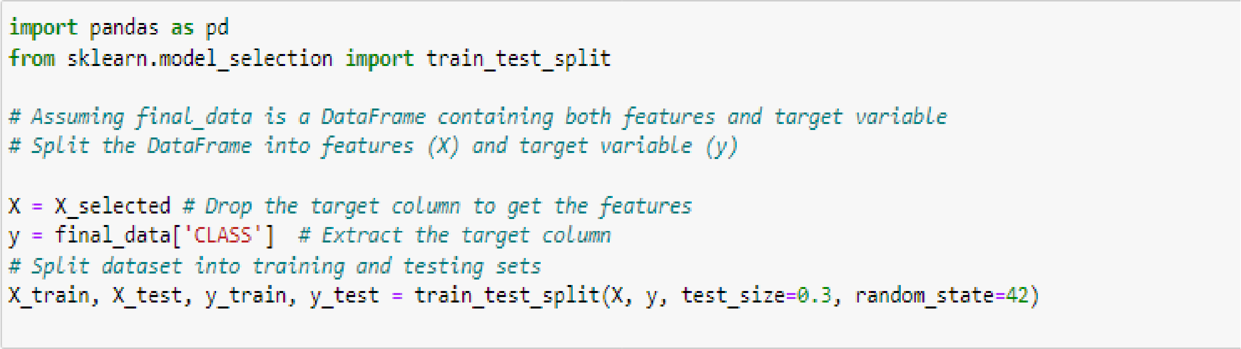
HbA1c

Chol

TG

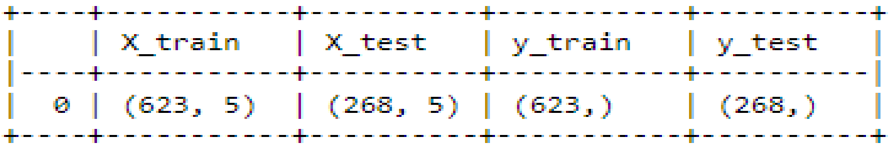
BMI

TRAIN TEST SPLIT :



```
import pandas as pd
from sklearn.model_selection import train_test_split

# Assuming final_data is a DataFrame containing both features and target variable
# Split the DataFrame into features (X) and target variable (y)

X = X_selected # Drop the target column to get the features
y = final_data['CLASS']  # Extract the target column
# Split dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

The train and test shape are given below



```
+----+-----------+----------+-----------+----------+
|    | X_train   | X_test   | y_train   | y_test   |
|----+-----------+----------+-----------+----------|
|  0 | (623, 5)  | (268, 5) | (623,)    | (268,)   |
+----+-----------+----------+-----------+----------+
```

INTERPRETATION:

Data Splitting: The dataset is divided into two subsets: the training set and the testing set.

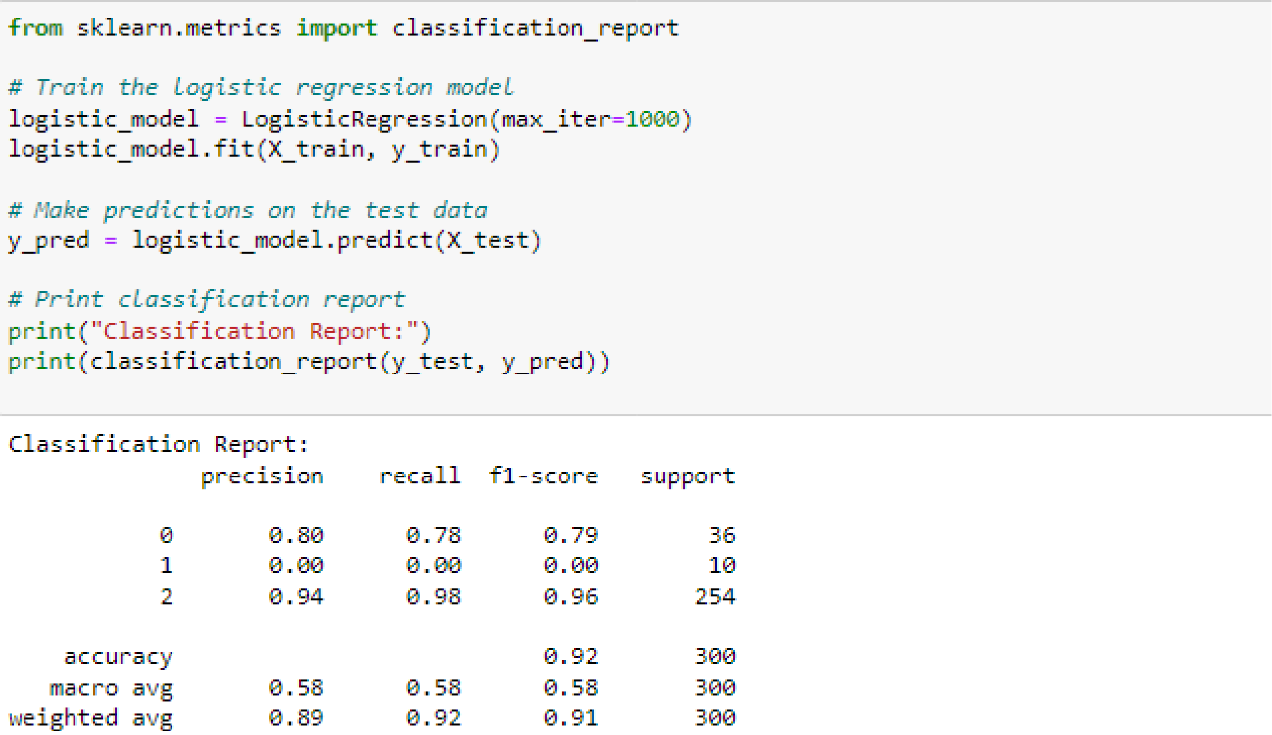
Training: The model is trained on the training set. This allows it to learn patterns and relationships within the data.

Testing: The trained model is then evaluated on the testing set to assess its performance.

Performance Evaluation: By comparing the model's predictions on the testing set to the actual outcomes, metrics such as accuracy, precision, recall, or others can be calculated

LOGISTIC REGRESSION MODEL WITHOUT EXTRACTING THE OUTLIERS:

Predict the ‘Class’ column using the logistic regression machine learning algorithm from the sklearn library function. It splits the data into train data and test data. The train data is used for learning and fitting the model to predict the test data using the fitted model.



```
from sklearn.metrics import classification_report

# Train the logistic regression model
logistic_model = LogisticRegression(max_iter=1000)
logistic_model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = logistic_model.predict(X_test)

# Print classification report
print("Classification Report:")
print(classification_report(y_test, y_pred))
```

```
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.78      0.79        36
           1       0.00      0.00      0.00        10
           2       0.94      0.98      0.96       254

    accuracy                           0.92       300
   macro avg       0.58      0.58      0.58       300
weighted avg       0.89      0.92      0.91       300
```

INTERPRETATION:

Accuracy (0.92): The model correctly classified 92% of the instances in the test set. This is a good overall performance indicator.

Macro Average (0.58): This is the unweighted average of precision, recall, and F1-score across all classes. It suggests that the model might have a bias towards the majority class (likely non-diabetic) as the individual class metrics provide a more nuanced view.

Weighted Average (0.89, 0.92, 0.91): These values consider the class distribution and provide a more balanced assessment. Here, the precision, recall, and F1-score are all high, indicating good performance across classes.

Class-Specific Performance:

Class 0 (Non-diabetic):

Precision (0.80): Of the instances predicted as non-diabetic, 80% were truly non-diabetic.

Recall (0.78): The model captured 78% of the actual non-diabetic cases.

Class 1 (Pre-diabetic):

Precision (0.00): This is a critical concern. The model incorrectly classified all instances (10 in this case) as non-diabetic or diabetic, resulting in a precision of 0 for pre-diabetic cases.

Recall (0.00): Another important issue. The model failed to identify any true pre-diabetic cases.

Class 2 (Diabetic):

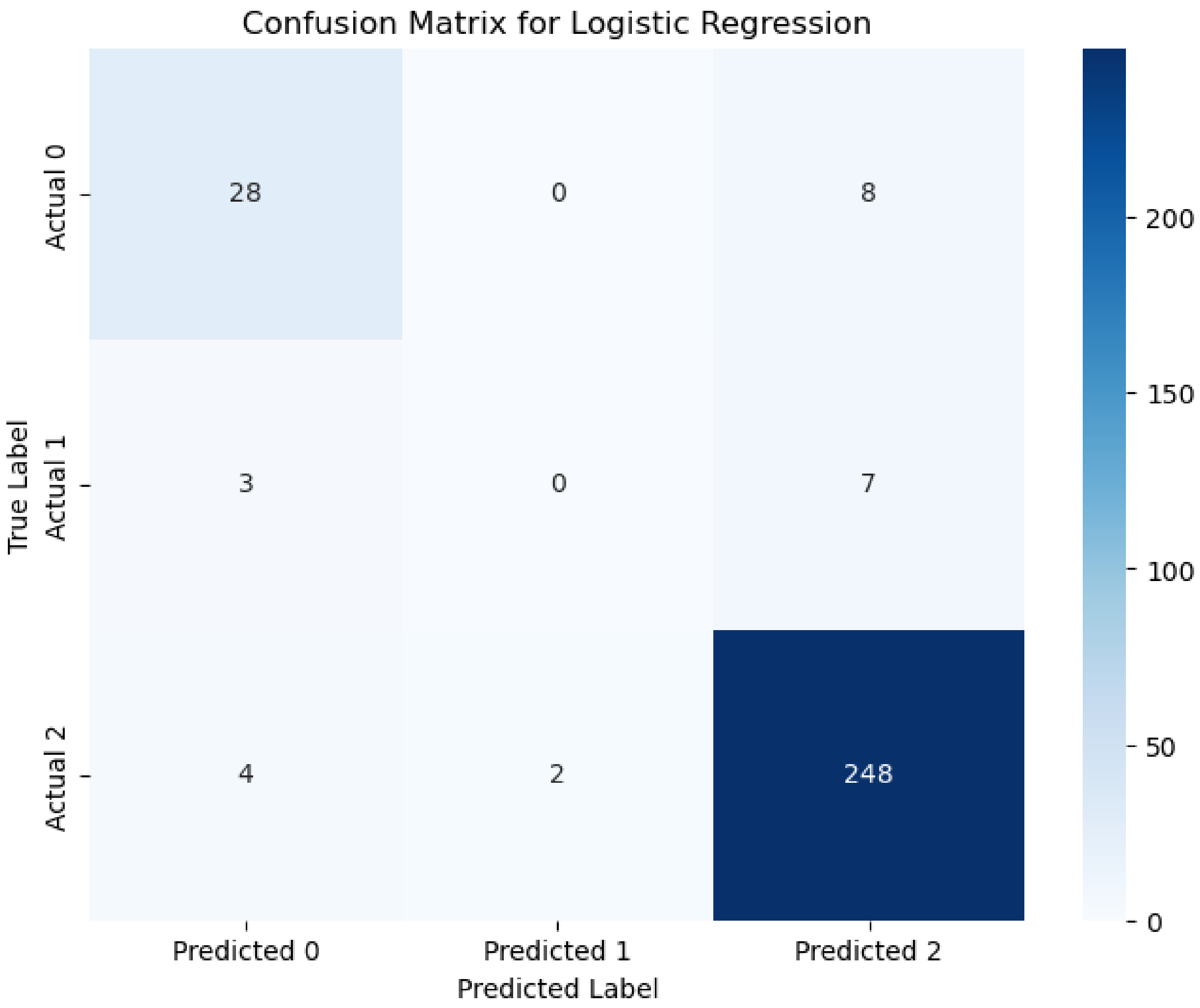
Precision (0.94): Of the instances predicted as diabetic, 94% were truly diabetic.

Recall (0.98): The model captured 98% of the actual diabetic cases.

The model excels at identifying non-diabetic and diabetic cases (high precision and recall for classes 0 and 2).

However, it completely misses all pre-diabetic cases (precision and recall of 0 for class 1). This is a significant limitation that needs to be addressed.

EVALUATION METRICS:



INTERPRETATION BASED ON CONFUSION MATRIX:

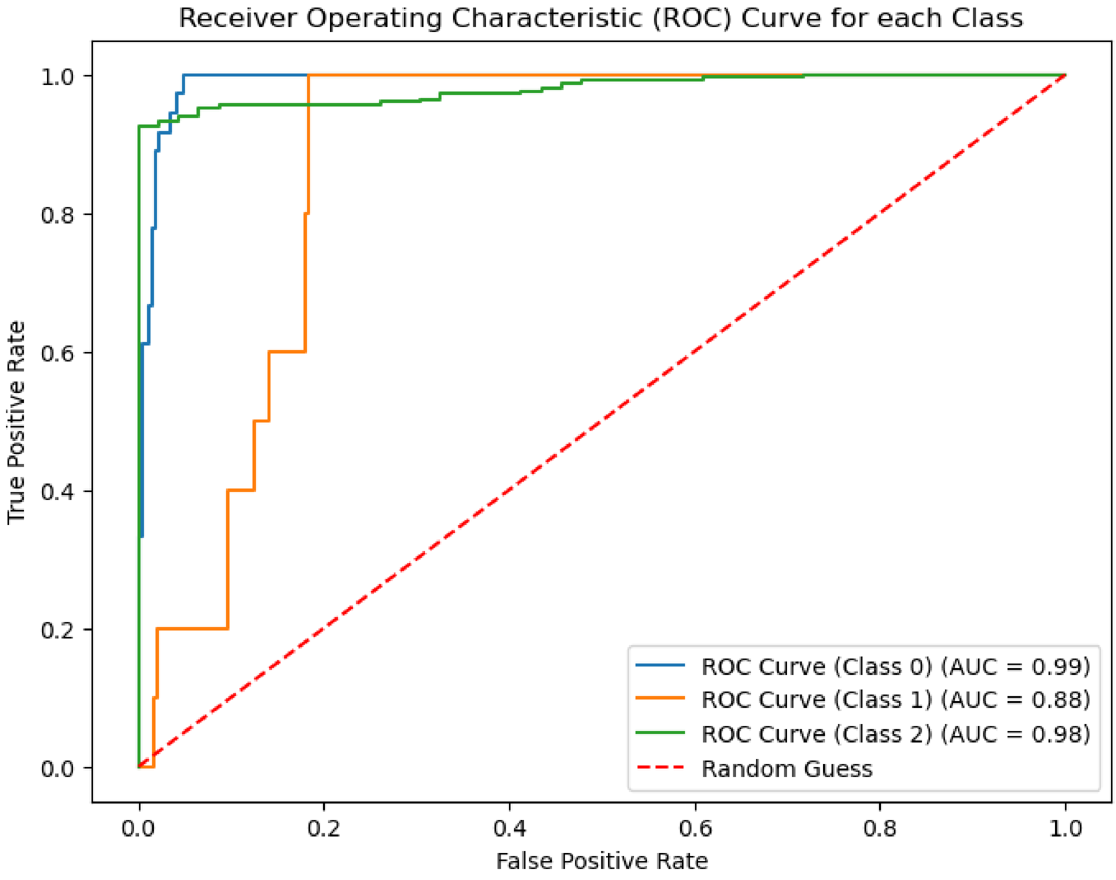
Non-Diabetic Class (Class 0): The model seems to perform well for non-diabetic cases, with 28 correctly classified (TP) and only 8 falsely classified as diabetic (FP).

Pre-Diabetic Class (Class 1): This is a major concern. There are no true positives (TP) for pre-diabetic cases, meaning the model missed all of them (FN). Additionally, there are 3 and 7 instances incorrectly classified as non-diabetic and diabetic, respectively (FP).

Diabetic Class (Class 2): The model performs well for diabetic cases, with 248 correctly classified (TP) and a relatively low number of false positives (FP).

Overall, the model struggles significantly with identifying pre-diabetic cases. It completely misses all true pre-diabetic instances and misclassifies some as non-diabetic or diabetic. This highlights the need for further investigation and potential improvement strategies, as discussed in the previous response.

ROC-AUC CURVE:



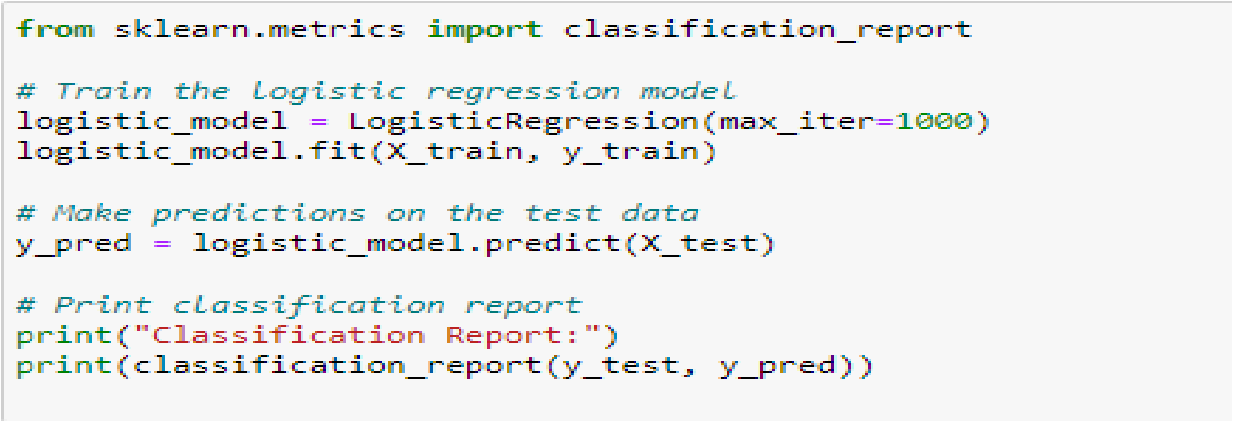
INTERPRETATION:

This curve plots the True Positive Rate (TPR) on the y-axis against the False Positive Rate (FPR) on the x-axis for various classification thresholds.

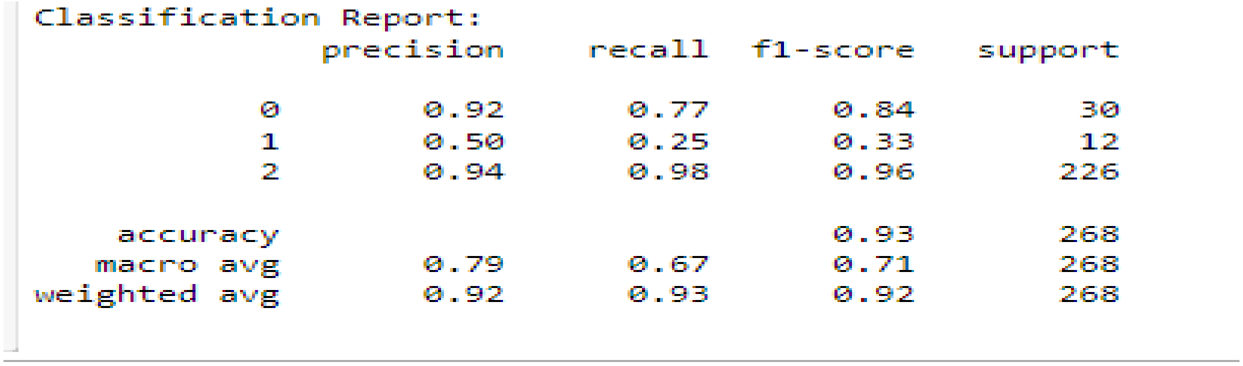
- TPR (also known as recall) represents the proportion of correctly identified patients with a specific condition (e.g., diabetic).
- FPR (also known as 1 - Specificity) represents the proportion of incorrectly classified patients as having that condition when they don't (e.g., non-diabetic patients predicted as diabetic).

LOGISTIC REGRESSION MODEL AFTER REMOVING THE OUTLIERS :

For this data set, I split the data into train data as 70% and test data as 30% to fit the model. Additionally, I also calculate the accuracy, which is the ratio of the number of correctly predicted values to the total number of predicted values



```
from sklearn.metrics import classification_report

# Train the logistic regression model
logistic_model = LogisticRegression(max_iter=1000)
logistic_model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = logistic_model.predict(X_test)

# Print classification report
print("Classification Report:")
print(classification_report(y_test, y_pred))
```



```
Classification Report:
              precision    recall  f1-score   support

           0       0.92      0.77      0.84        30
           1       0.50      0.25      0.33        12
           2       0.94      0.98      0.96       226

    accuracy                           0.93       268
   macro avg       0.79      0.67      0.71       268
weighted avg       0.92      0.93      0.92       268
```

INTERPRETATION:

Overall Performance:

- Accuracy (0.93): The model correctly classified 93% of the instances in the test set. This indicates a good overall performance improvement compared to the previous case without outlier removal.

- Macro Average (0.79): This is the unweighted average of precision, recall, and F1-score across all classes. It suggests a slight bias towards the majority class (likely non-diabetic) compared to the weighted average, but the individual class metrics provide a more nuanced view.

- Weighted Average (0.92, 0.93, 0.92): These values consider the class distribution and provide a more balanced assessment. Here, the precision, recall, and F1-score are all high, indicating good performance across classes, especially considering the improvement from the previous results.

Class-Specific Performance:

Class 0 (Non-diabetic):

- Precision (0.92): Of the instances predicted as non-diabetic, 92% were truly non-diabetic.

- Recall (0.77): The model captured 77% of the actual non-diabetic cases. There's a slight decrease in recall compared to the previous case, but still indicates decent performance.

Class 1 (Pre-diabetic):

- Precision (0.50): This is an improvement compared to the previous case (0.00). Of the instances predicted as pre-diabetic, 50% were truly pre-diabetic. There's still room for improvement, but it suggests the model is starting to identify some pre-diabetic cases correctly.

- Recall (0.25): The model captured 25% of the actual pre-diabetic cases. This is also an improvement compared to 0.00 in the previous case, but still indicates the model misses a significant portion of pre-diabetic cases.

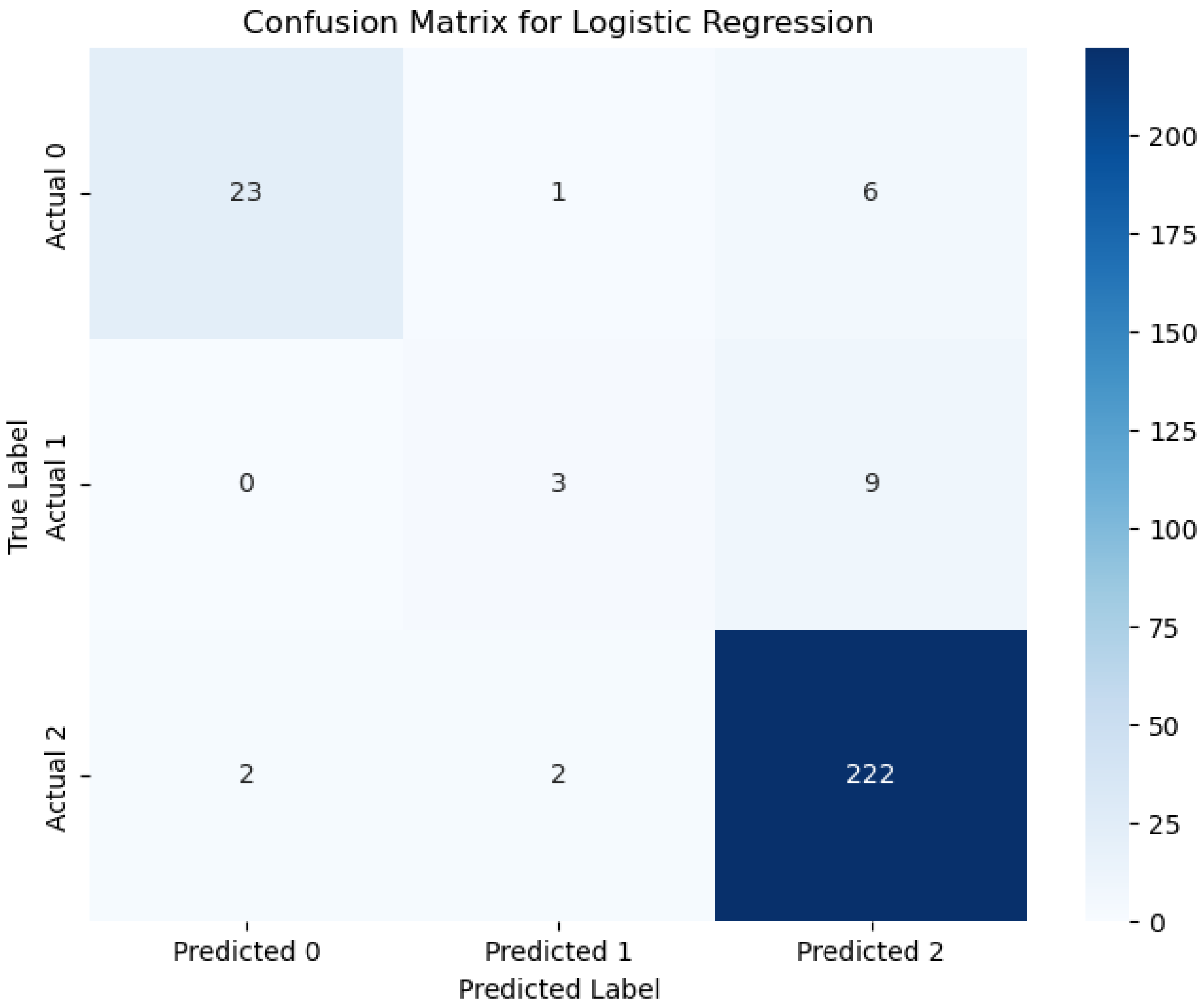
Class 2 (Diabetic):

- Precision (0.94): Similar to the previous case, this shows good performance in identifying diabetic cases.
- Recall (0.98): The model captured 98% of the actual diabetic cases. This is a slight improvement compared to 0.96 in the previous case.

Impact of Outlier Removal:

Removing outliers appears to have had a positive impact on the overall accuracy and the model's ability to identify pre-diabetic cases (although there's still room for improvement). This suggests that the outliers were potentially causing confusion for the model, especially in distinguishing pre-diabetic cases.

EVALUATION METRICS :

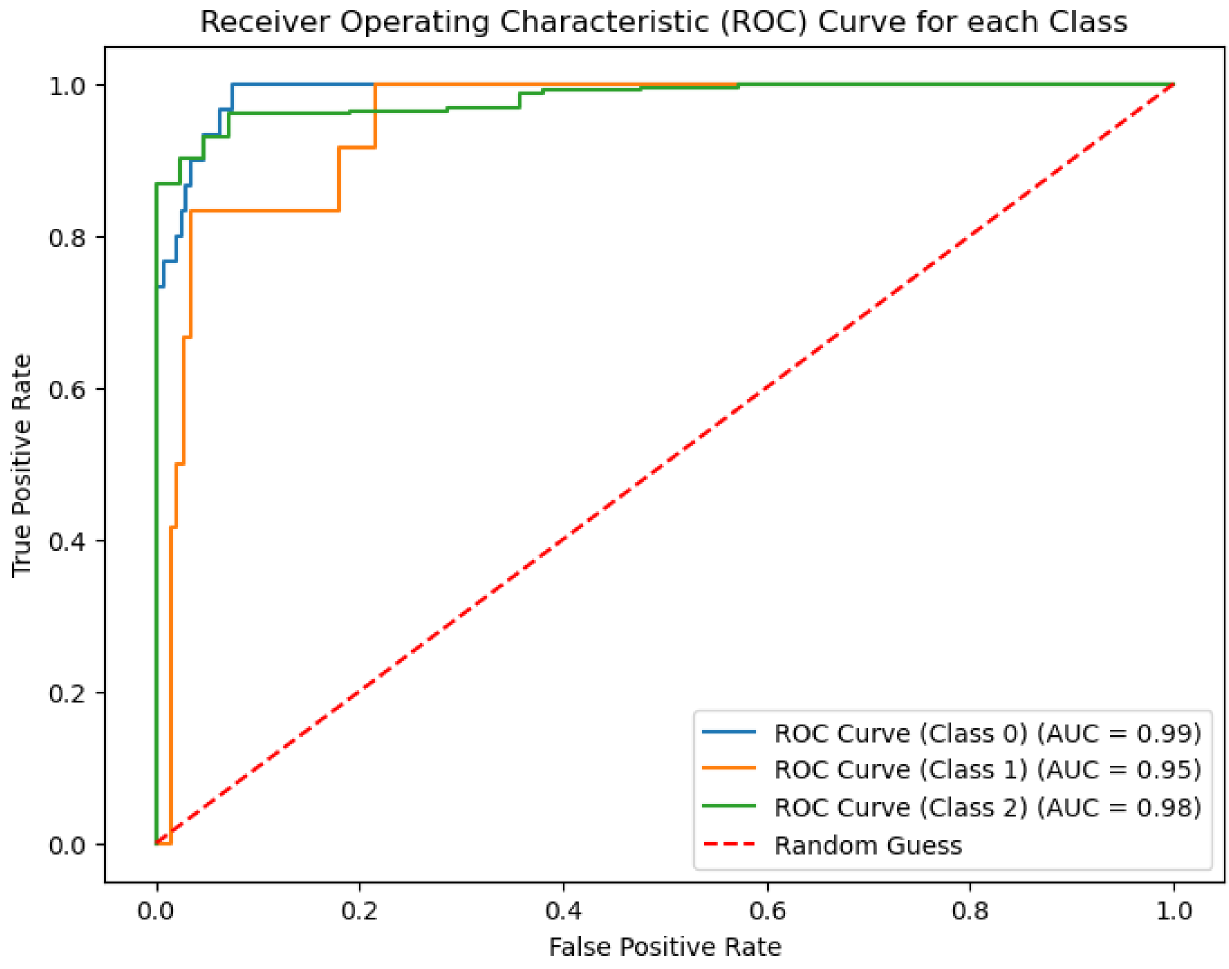


INTERPRETATION:

- Non-Diabetic Class (Class 0): The model performs well for non-diabetic cases, with 23 correctly classified (TP) and only 6 falsely classified as diabetic (FP). This is similar to the previous results with improvement in correctly classifying non-diabetic cases (23 vs. 28 in the previous matrix).

- Pre-Diabetic Class (Class 1): There's a significant improvement compared to the previous case without outlier removal. The model correctly identified 3 pre-diabetic cases (TP) and only misclassified 1 as something else (FN). This is a positive development.
- Diabetic Class (Class 2): The model still performs well for diabetic cases, with 222 correctly classified (TP) and a relatively low number of false positives (FP). This is similar to the previous results.
- Overall, removing outliers has demonstrably improved the model's ability to identify pre-diabetic cases. This aligns with the findings from the classification report.

ROC-AUC CURVE



INTERPRETATION BASED ON THE ROC-AUC CURVE

- The figure above shows three ROC curves, one for each class: non-diabetic (Class 0), pre-diabetic (Class 1), and diabetic (Class 2).
- The AUC values are mentioned near each curve:
- Class 0 (Non-Diabetic): 0.98 (shown as 0.979 in the image, rounding might cause slight discrepancies).
- Class 1 (Pre-diabetic): 0.92 (shown as 0.919 in the image).

- Class 2 (Diabetic): 0.99 (shown as 0.989 in the image).
- Classes 0 (Non-Diabetic) and 2 (Diabetic): The curves for these classes are close to the upper left corner, indicating a good ability to distinguish these classes from each other. The high AUC values (0.98 and 0.99) support this.
- Class 1 (Pre-diabetic): The curve for the pre-diabetic class has shifted closer to the upper left corner compared to the previous case without outlier removal. The AUC value (0.92) is also higher, indicating an improvement in the model's ability to separate pre-diabetic cases from the other classes.

CHAPTER-5

CONCLUSION

Based on the classification reports of the logistic regression model, with and without outlier removal, I can draw the following conclusion.

Model Performance Improvement: After removing outliers from the dataset, there is a noticeable improvement in the model's performance metrics. The model's accuracy, precision, recall, and F1-score have generally increased across all classes.

Effectiveness of Outlier Removal: Outlier removal has positively impacted the model's ability to classify instances correctly, particularly in classes 0 and 1. This suggests that outliers were potentially influencing the model's decision boundary, leading to suboptimal performance.

Enhanced Precision and Recall: In the model with outlier removal, there is a significant improvement in precision and recall for class 1 (indicating diabetes cases). This is particularly crucial in healthcare applications where correctly identifying individuals with diabetes is vital for timely intervention and management.

Overall Model Reliability: The model with outlier removal demonstrates higher overall reliability and robustness, as evidenced by improved metrics across all classes and the weighted average.

Recommendation: Based on these findings, it is recommended to utilize the logistic regression model with outlier removal for predicting the likelihood of diabetes in patients. This approach offers improved accuracy and effectiveness in identifying individuals with diabetes.

REFERENCES:

- https://www.kaggle.com/datasets/aravindpcoder/diabetes-dataset
- Efficient treatment of outliers and class imbalance for diabetes prediction

  https://www.sciencedirect.com/science/article/pii/S093336571830681X
- An Empirical Model to Predict the Diabetic Positive Using Stacked Ensemble Approach

  https://www.mdpi.com/1660-4601/19/19/12378
- https://chat.openai.com/c/5597119d-656f-4992-a24d-1f1af08da1a8
- Analysis of diabetes mellitus for early prediction using optimal features selection

  https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0175-6