STARK HEALTH CLINIC -DIABETES PREDICTION PROJECT

BY:
Sesay Isaac Santous

AUGUST 2024

Background and Problem Statement

- Stark Health Clinic is a healthcare provider
- Leverages technology and predictive modelling to enhance its operations
- Integrating machine learning into its systems
- Identifies diseases early, improving patient outcomes and resource allocation.
- Health and financial challenges due to Diabetes
- Current methods for early detection lack precision
- Missed opportunities for timely interventions

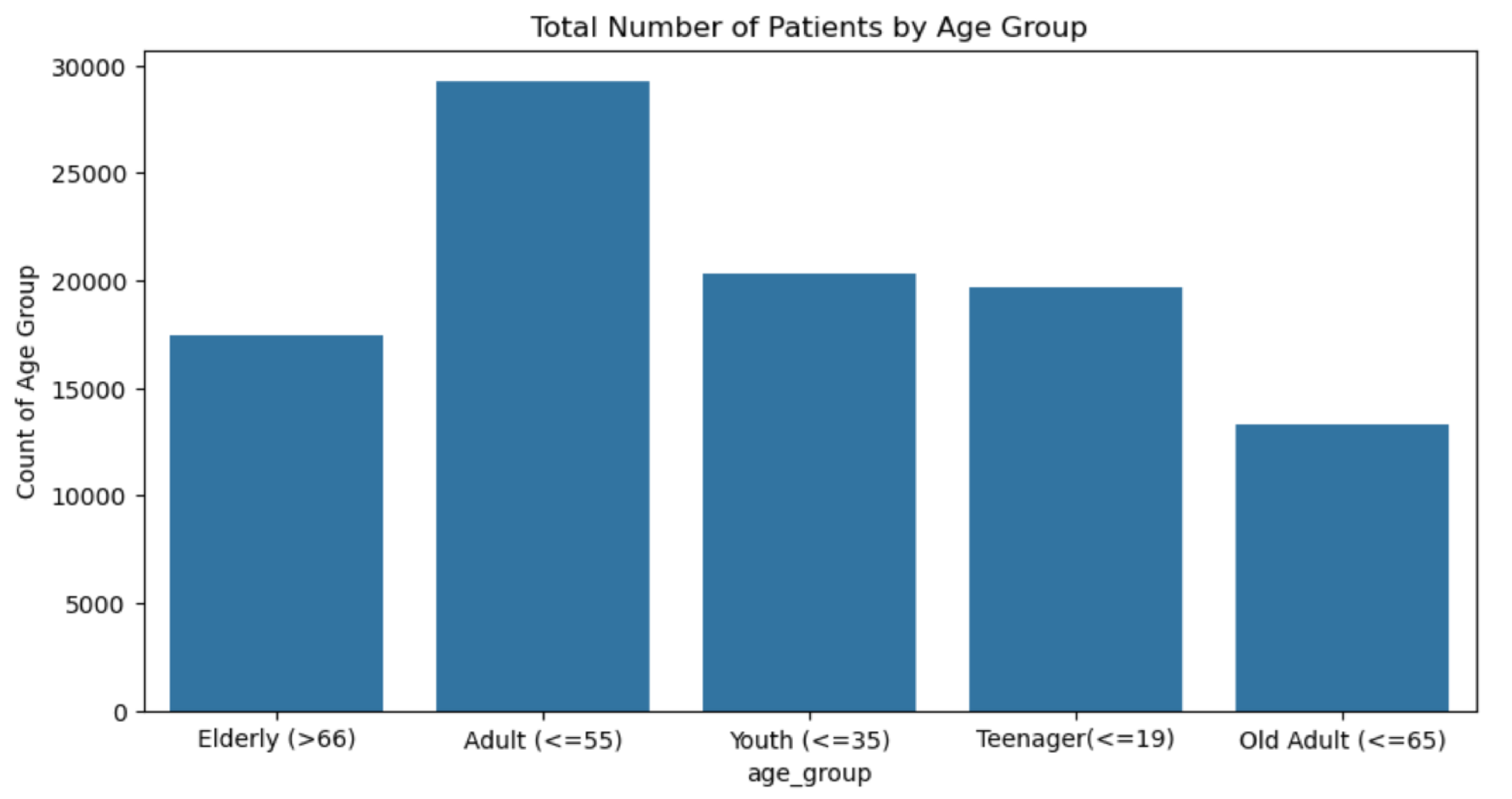
Rationale and Objective of the Project

- Accurate prediction of Diabetes through advanced Machine Learning
- Improve patient care, reduce costs, and take a proactive role in combating diabetes
- Aims at developing a robust diabetes prediction model to accurately identify individuals at risk of diabetes.
- To predict the likelihood of diabetes onset

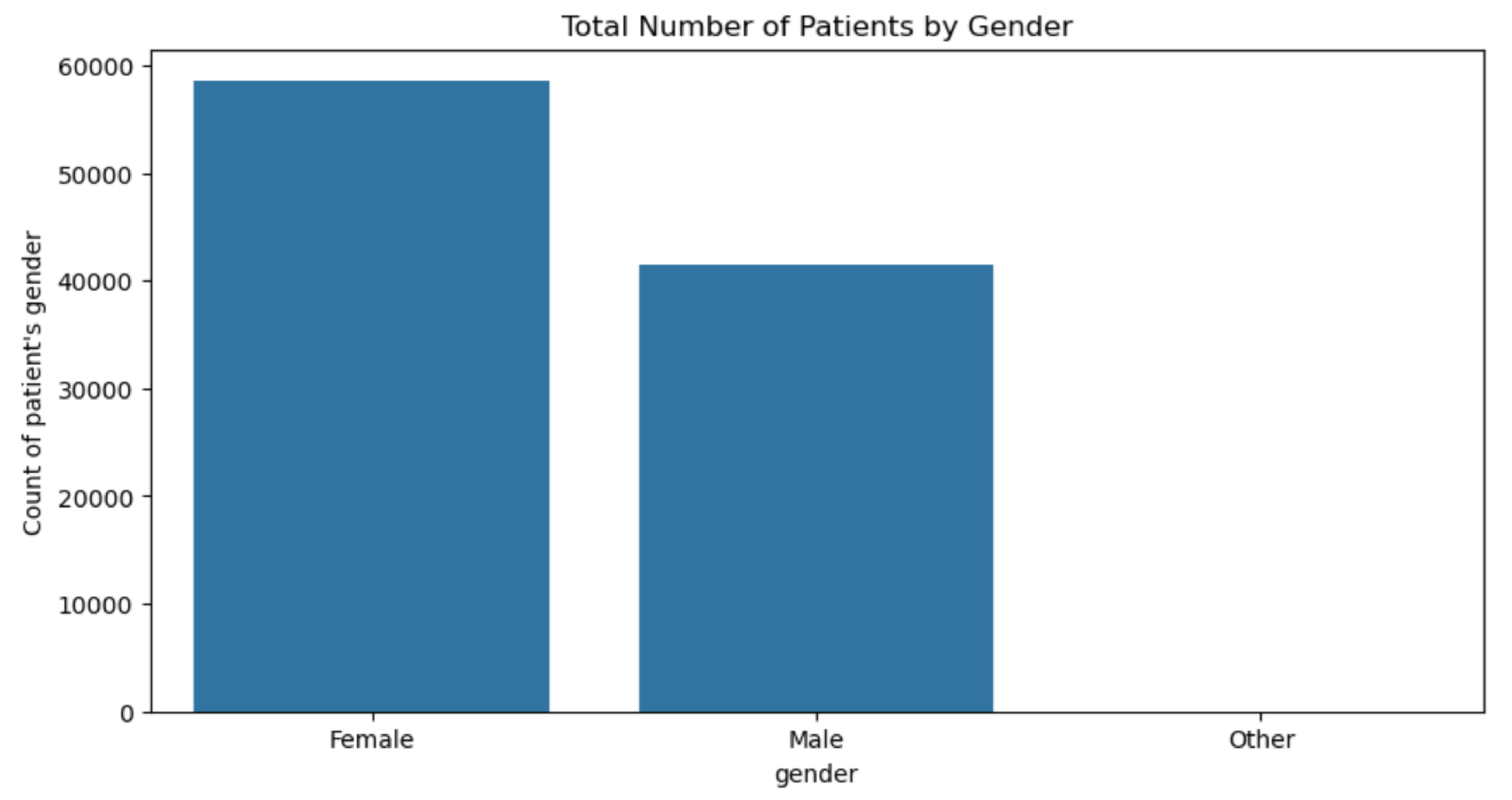
Steps

- Data cleaning
- Import Required Libraries
- Load the data
- Data verification - type, features, rows, missing data etc
- Check for missing values
- Perform exploratory Data Analysis (EDA)
- Data pre-processing/feature engineering
- Machine Learning

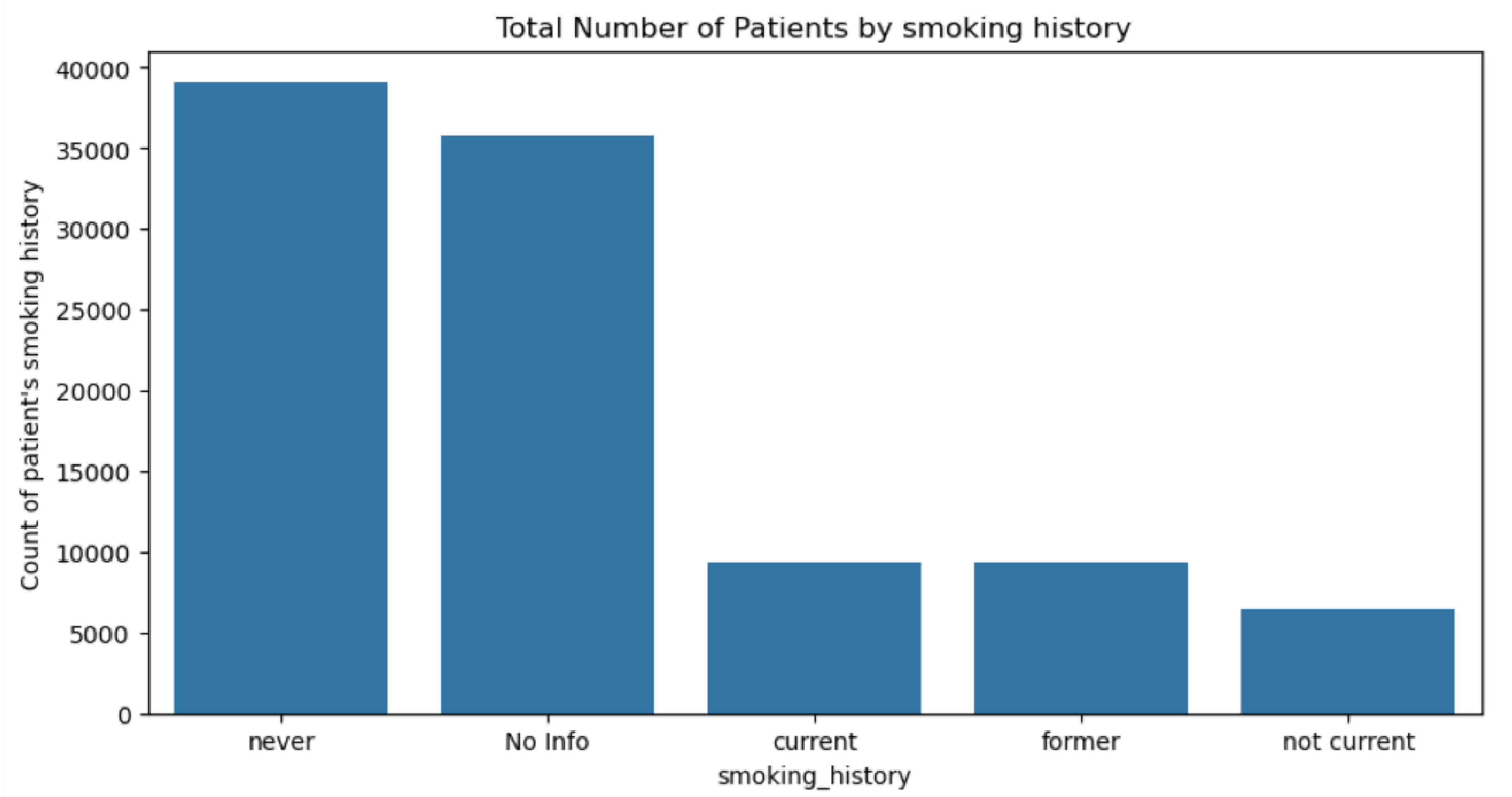
Categorical Analysis



Analysis


Total Number of Patients by Gender
Count of patient's gender
60000
50000
40000
30000
20000
10000
0
Female
Male
Other
gender

Analysis



Total Number of Patients by smoking history
Count of patient's smoking history
40000
35000
30000
25000
20000
15000
10000
5000
0
never
No Info
current
former
not current
smoking_history

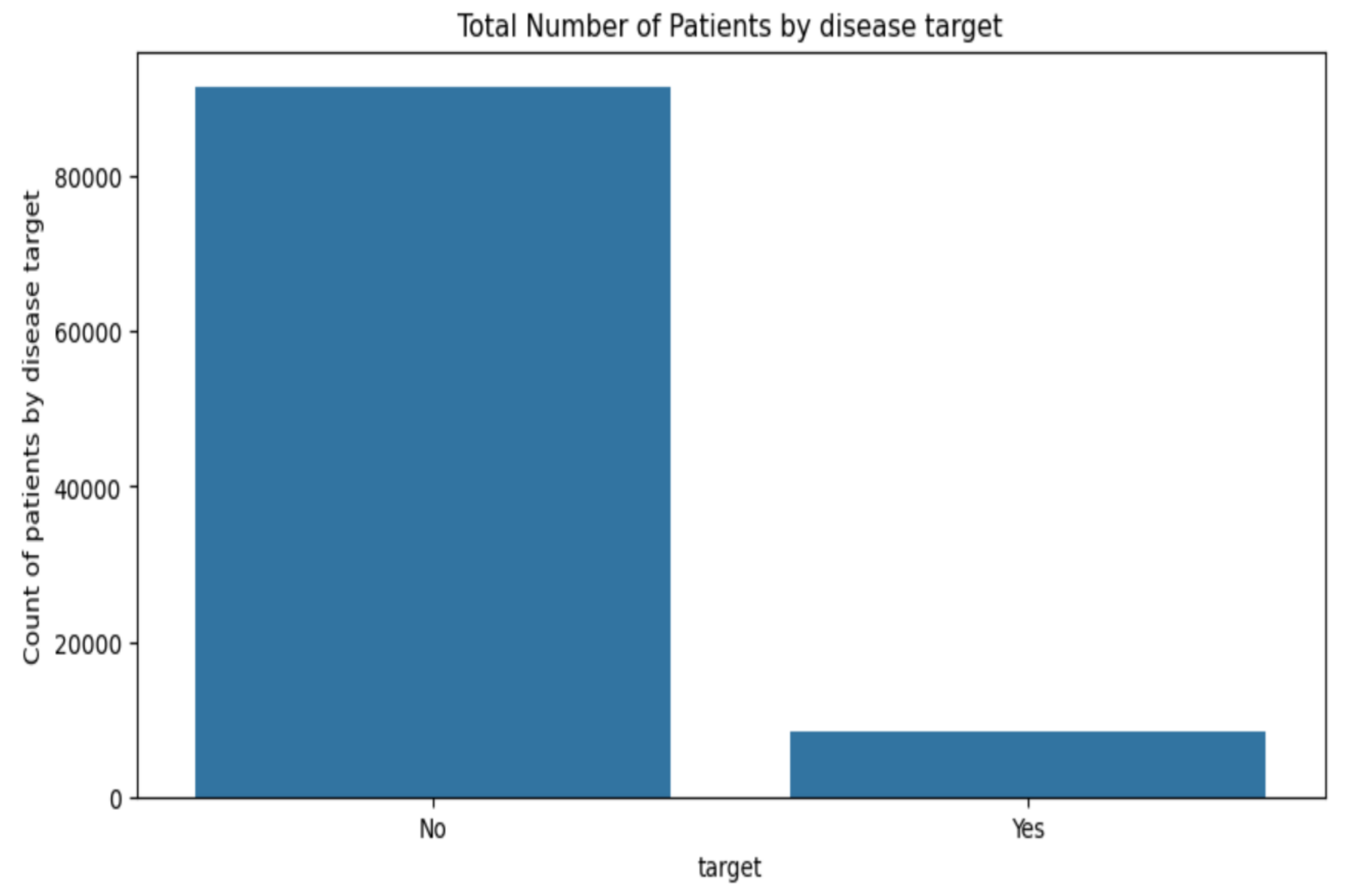
Analysis

Target counts

No	*91500*
Yes	*8500*

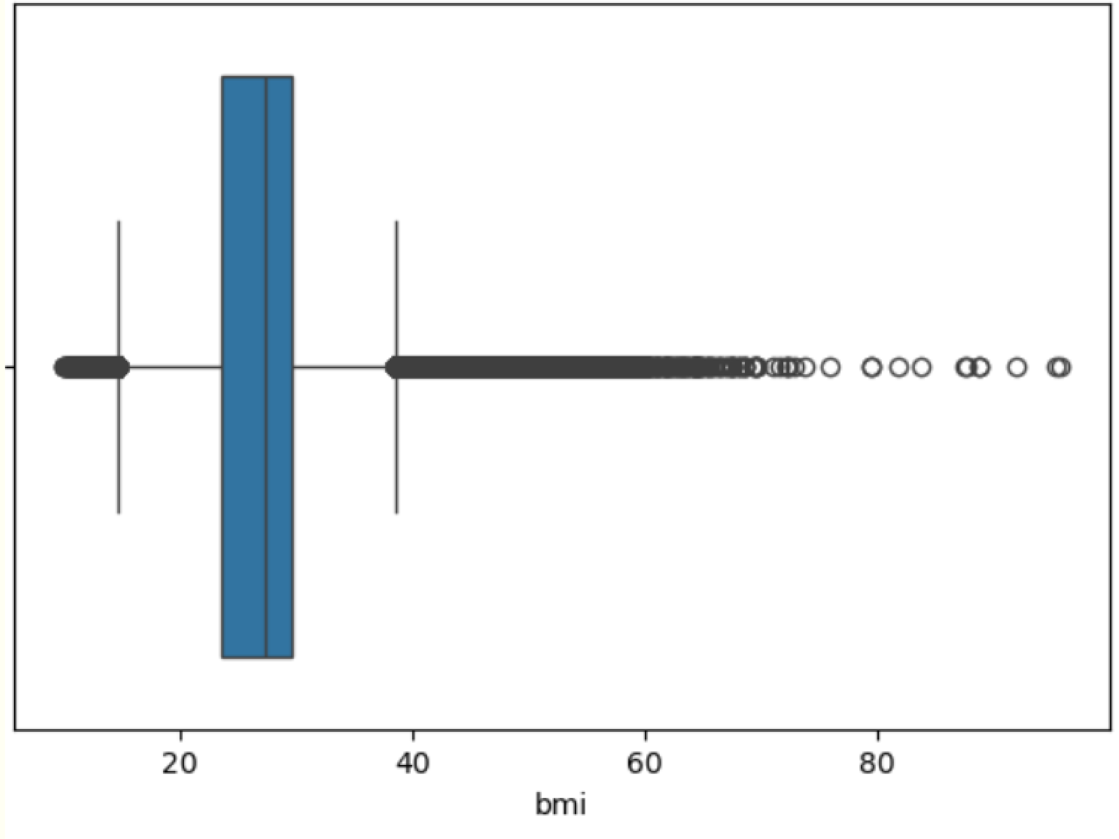
Target Percentage

No	*0.915*
Yes	*0.085*

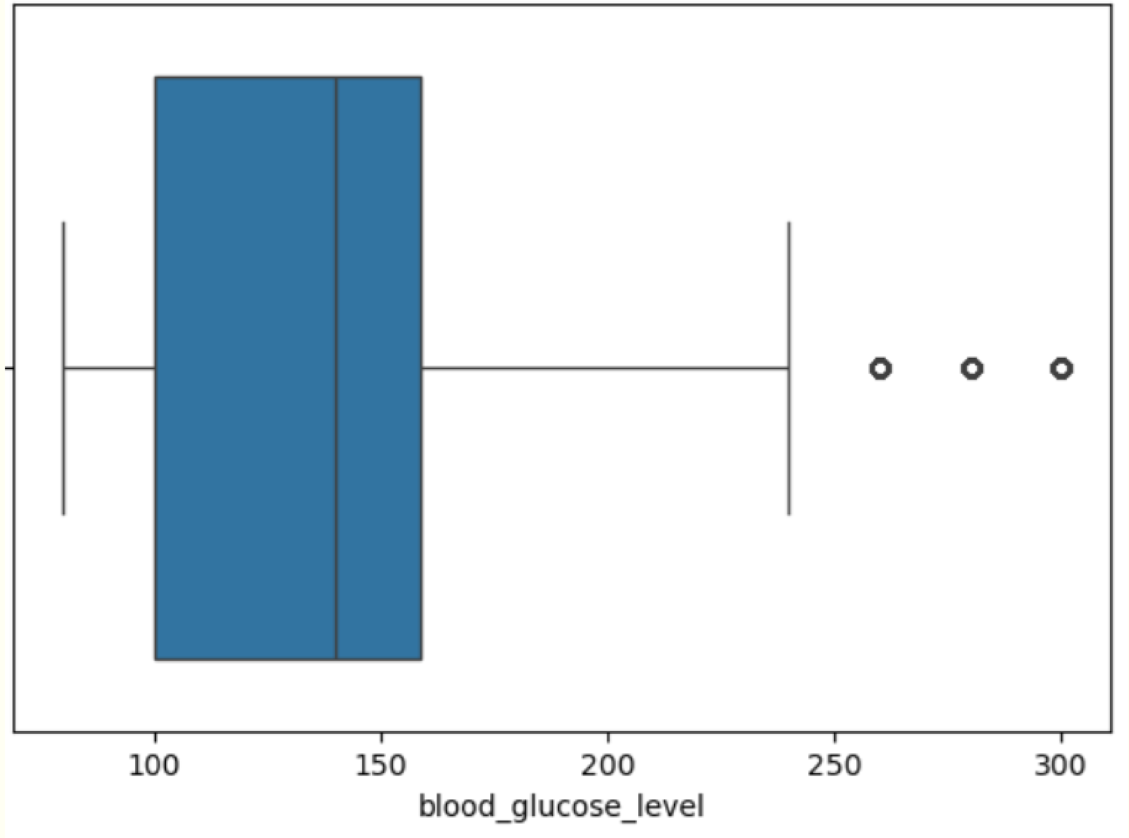


Univariate Analysis

Boxplot on BMI

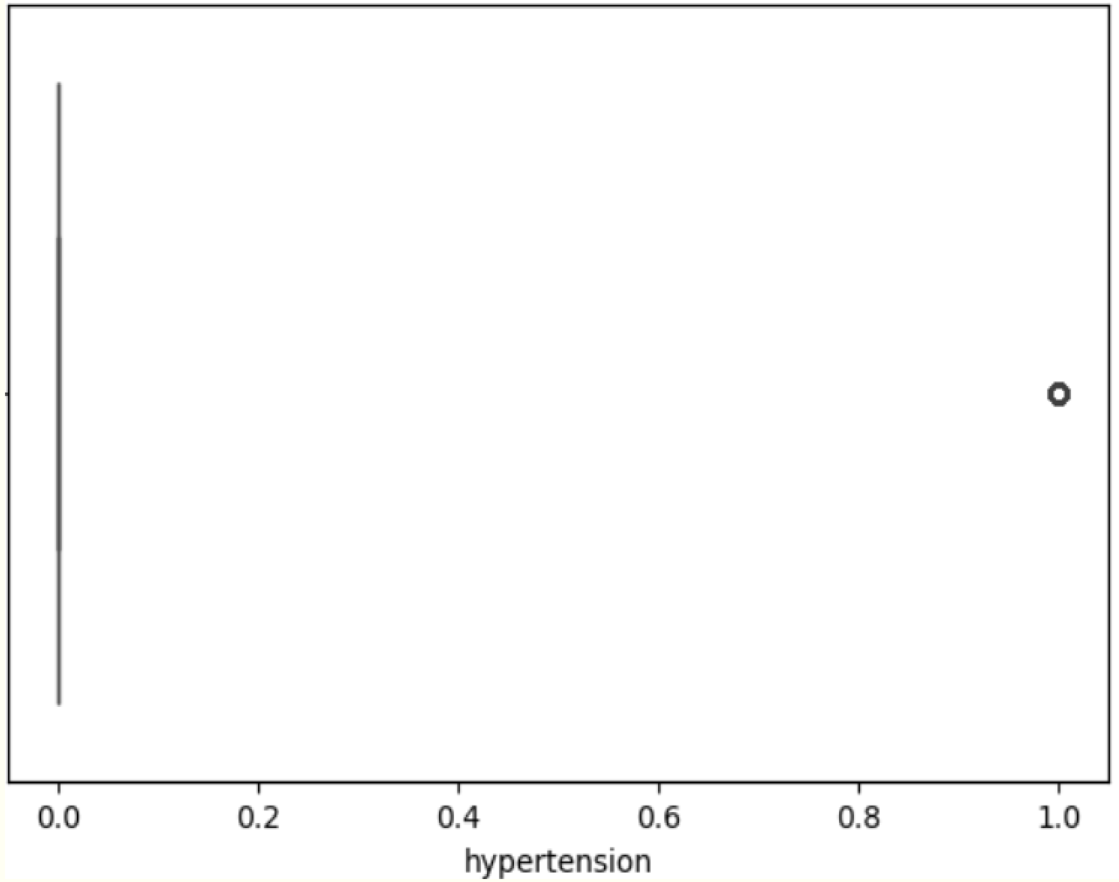


Boxplot on blood glucose level

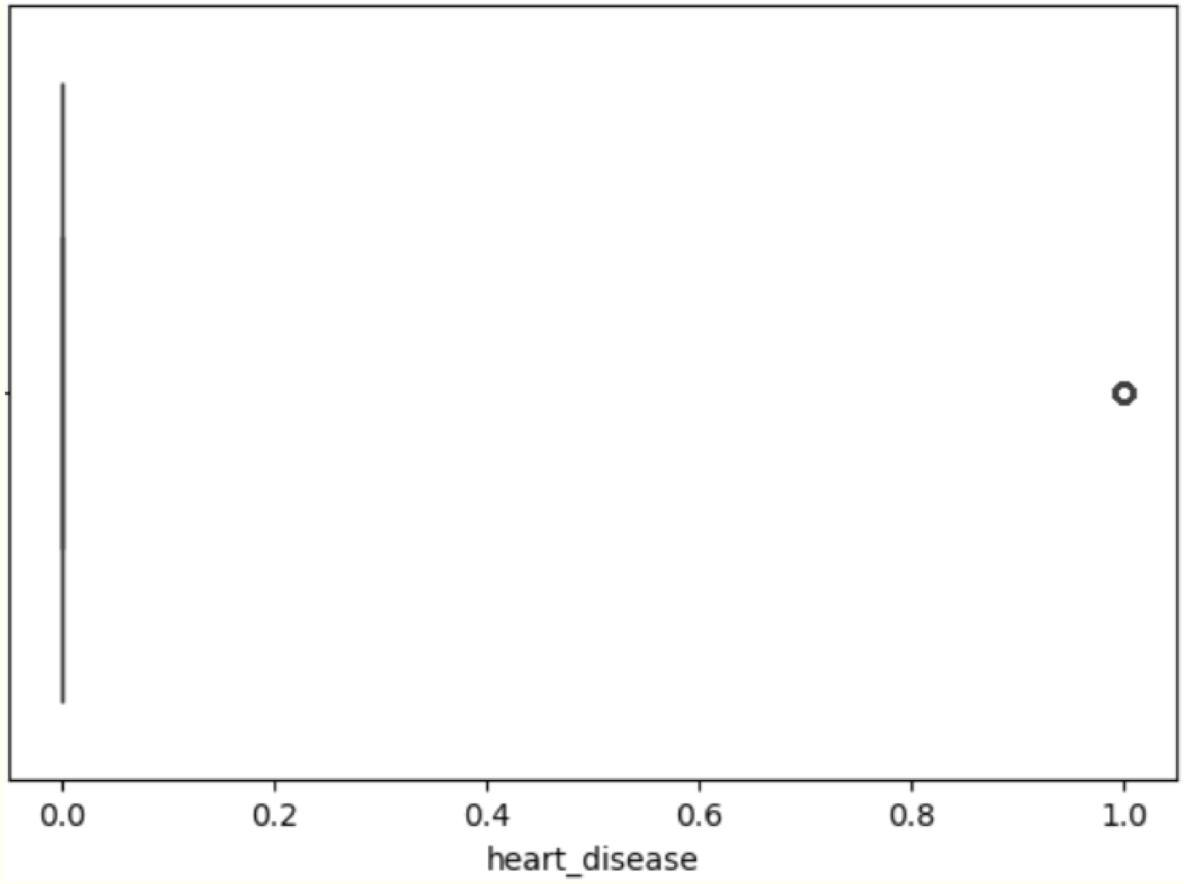


Univariate Analysis

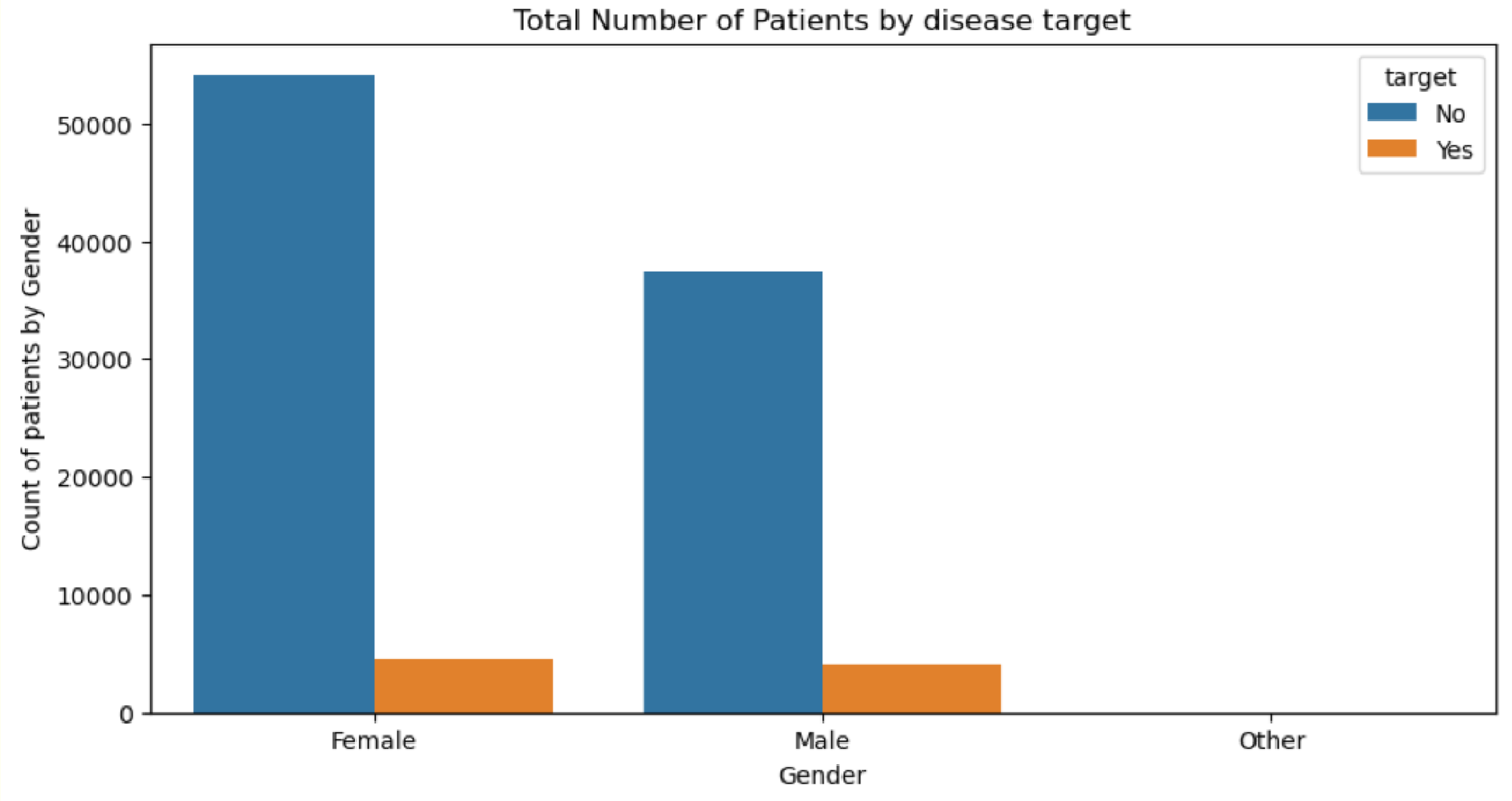
Boxplot on Hypertension



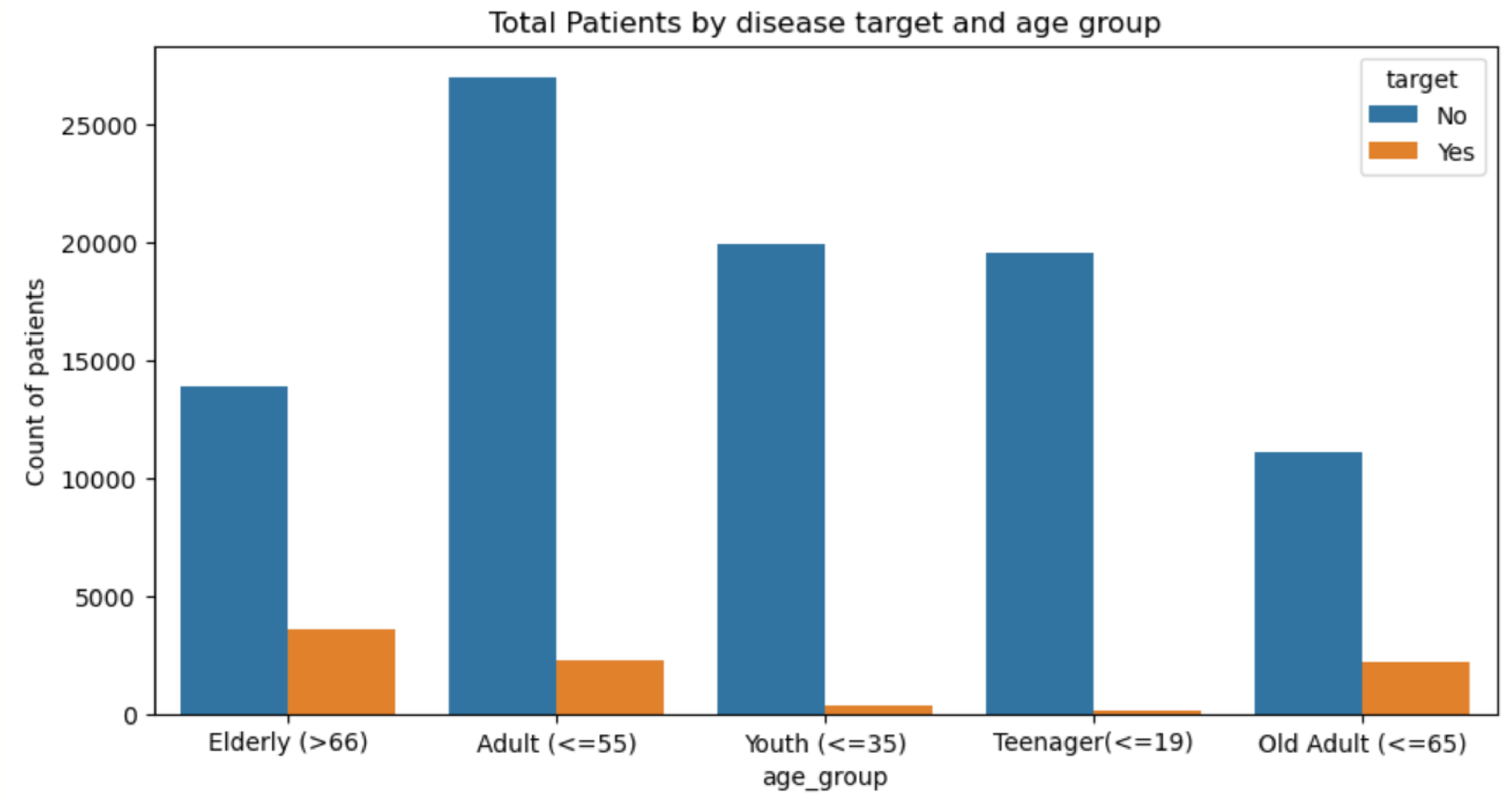
Boxplot on Heart disease



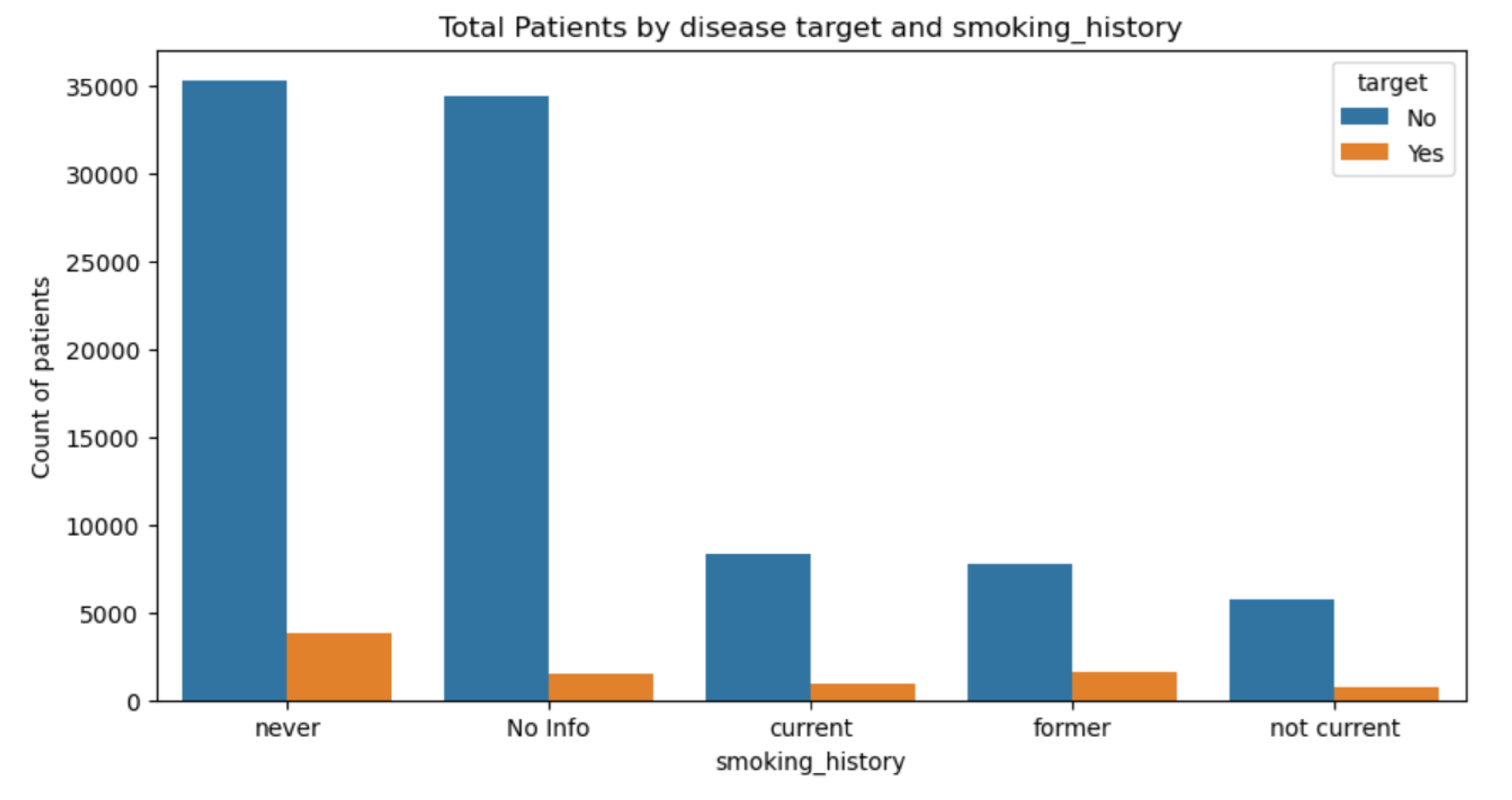
Bivariate Analysis



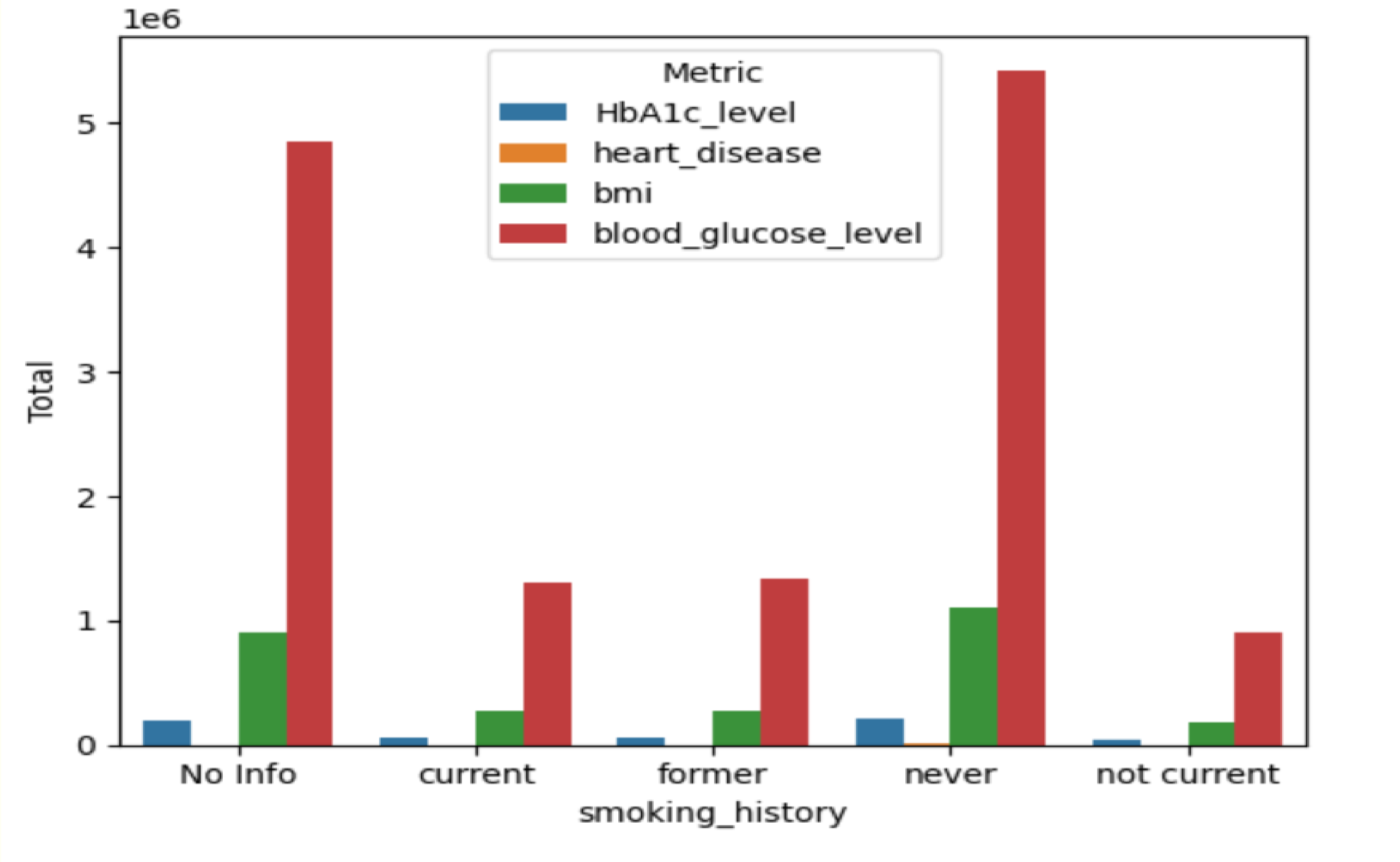
Bivariate Analysis



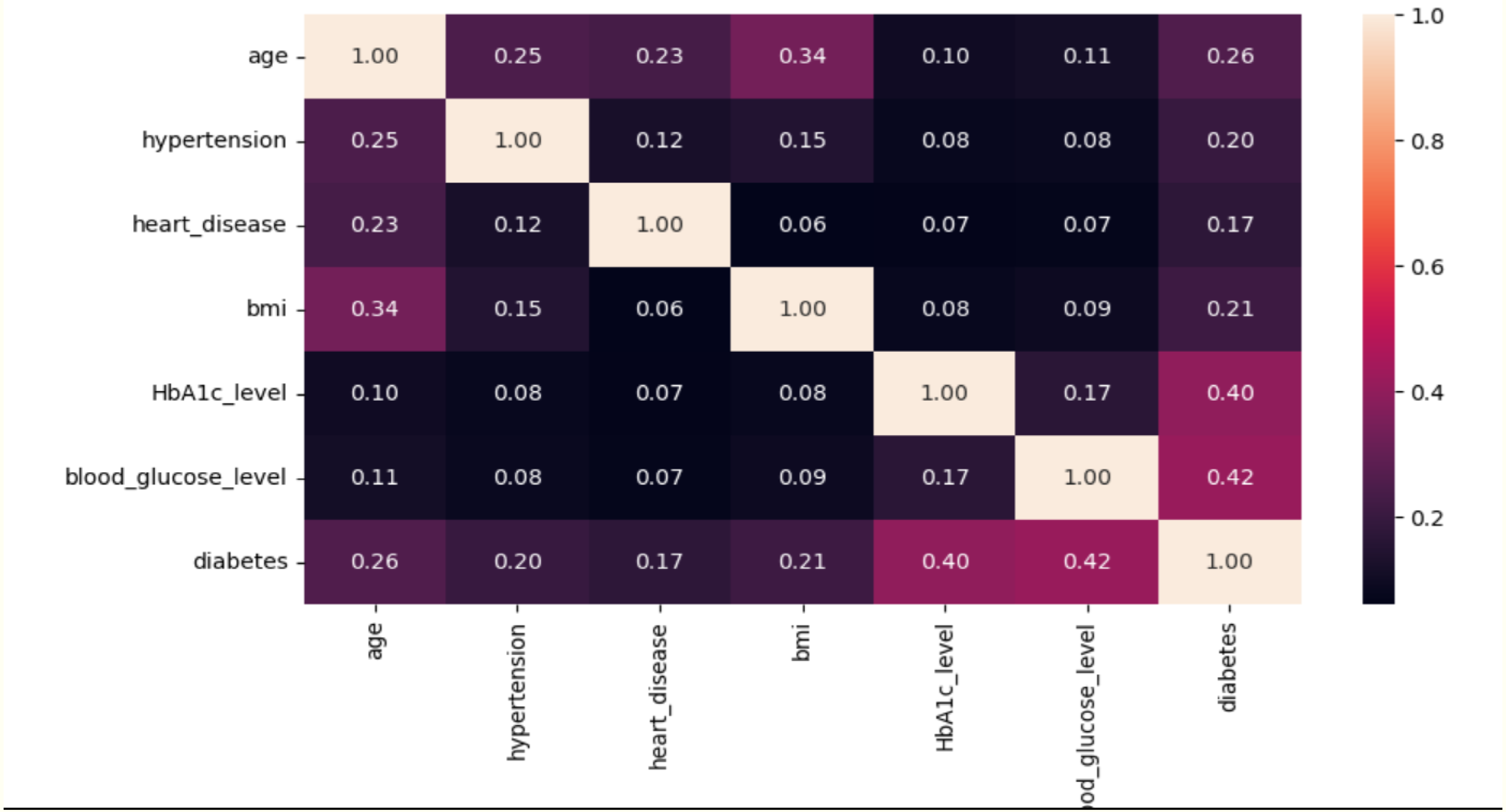
Bivariate Analysis



Multivariate Analysis

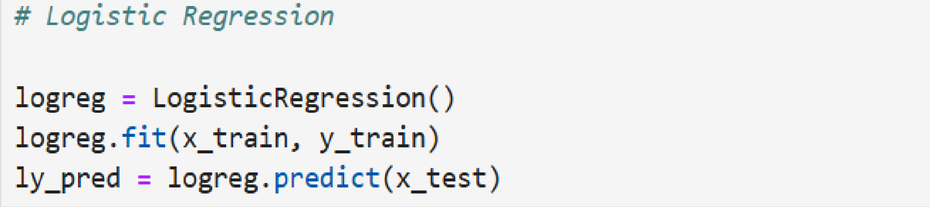


Multivariate Analysis

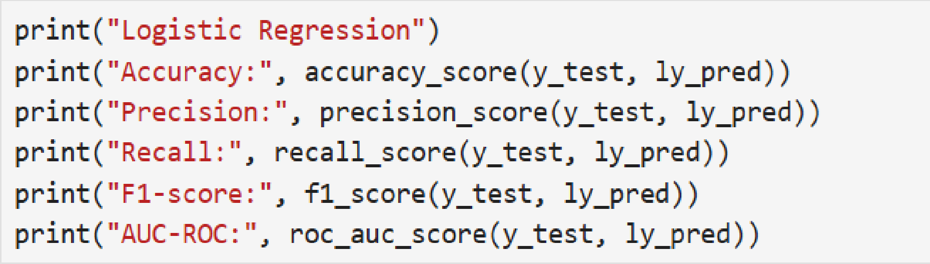


	age	hypertension	heart_disease	bmi	HbA1c_level	od_glucose_level	diabetes
age	1.00	0.25	0.23	0.34	0.10	0.11	0.26
hypertension	0.25	1.00	0.12	0.15	0.08	0.08	0.20
heart_disease	0.23	0.12	1.00	0.06	0.07	0.07	0.17
bmi	0.34	0.15	0.06	1.00	0.08	0.09	0.21
HbA1c_level	0.10	0.08	0.07	0.08	1.00	0.17	0.40
blood_glucose_level	0.11	0.08	0.07	0.09	0.17	1.00	0.42
diabetes	0.26	0.20	0.17	0.21	0.40	0.42	1.00

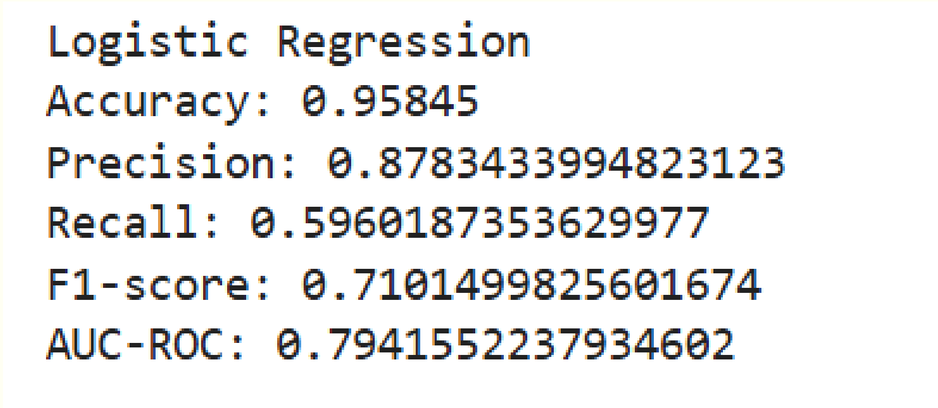
Predictive Models



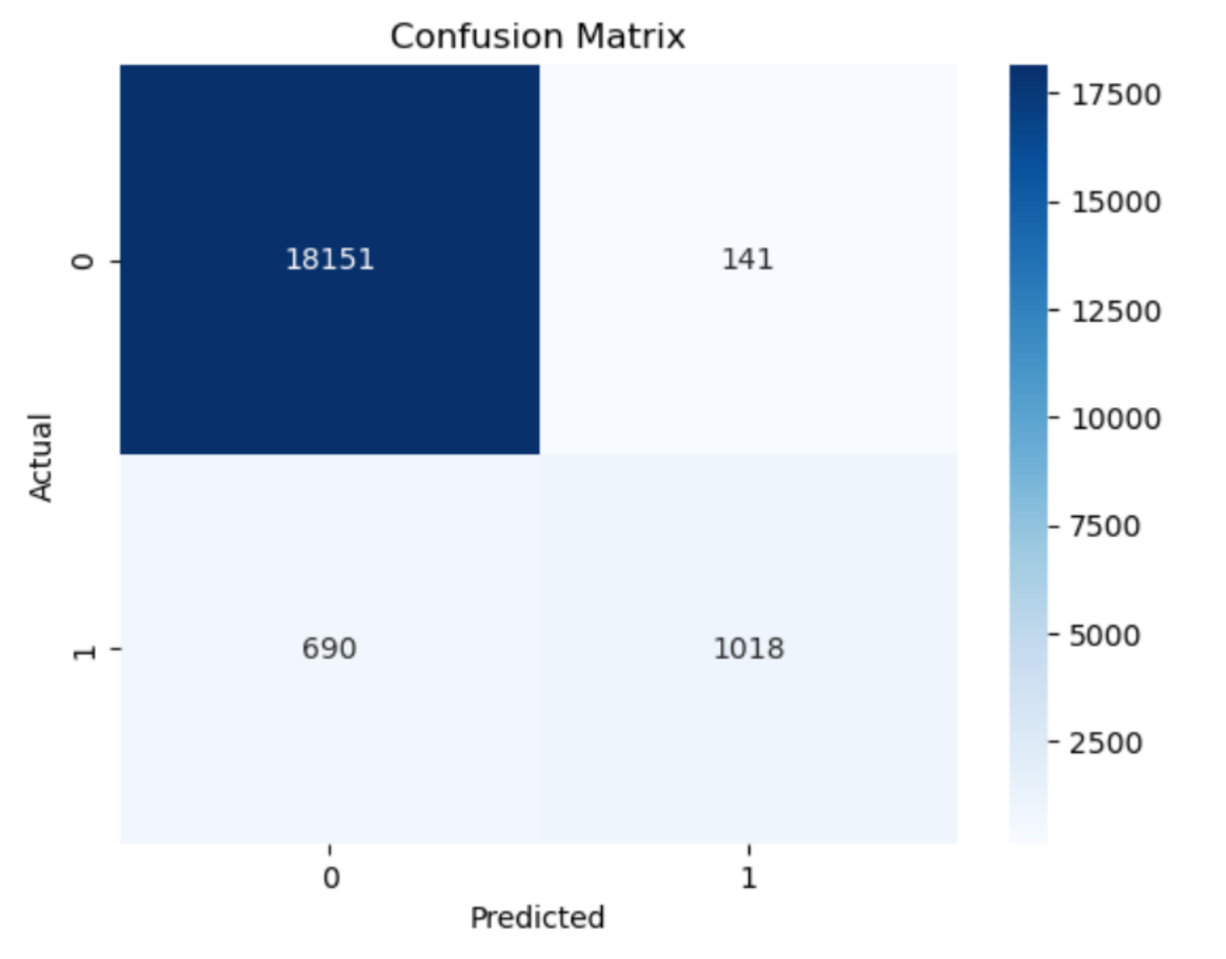
```python
# Logistic Regression

logreg = LogisticRegression()
logreg.fit(x_train, y_train)
ly_pred = logreg.predict(x_test)
```



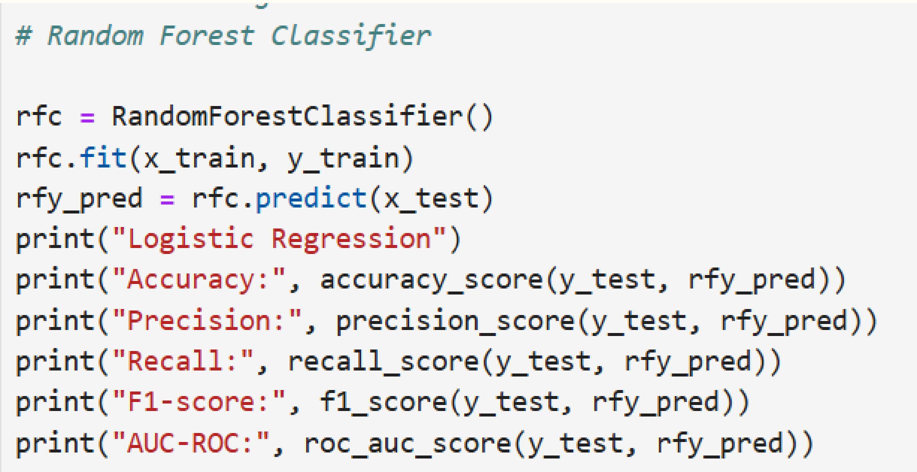
```python
print("Logistic Regression")
print("Accuracy:", accuracy_score(y_test, ly_pred))
print("Precision:", precision_score(y_test, ly_pred))
print("Recall:", recall_score(y_test, ly_pred))
print("F1-score:", f1_score(y_test, ly_pred))
print("AUC-ROC:", roc_auc_score(y_test, ly_pred))
```



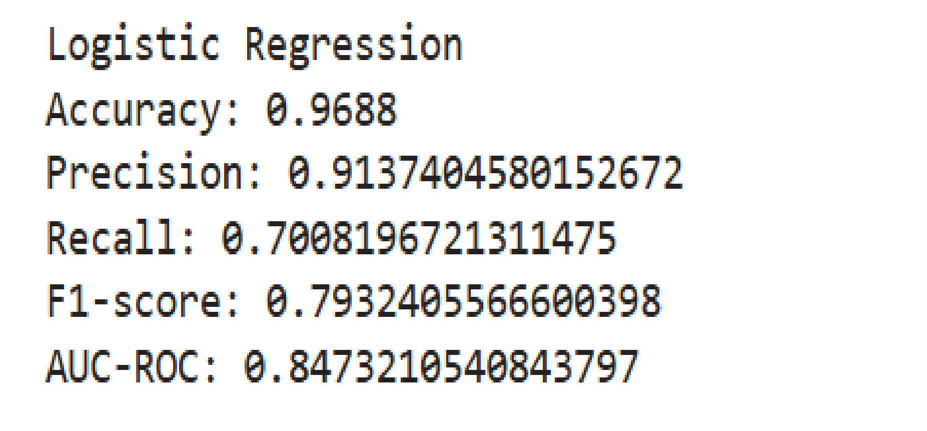
```
Logistic Regression
Accuracy: 0.95845
Precision: 0.8783433994823123
Recall: 0.5960187353629977
F1-score: 0.7101499825601674
AUC-ROC: 0.7941552237934602
```



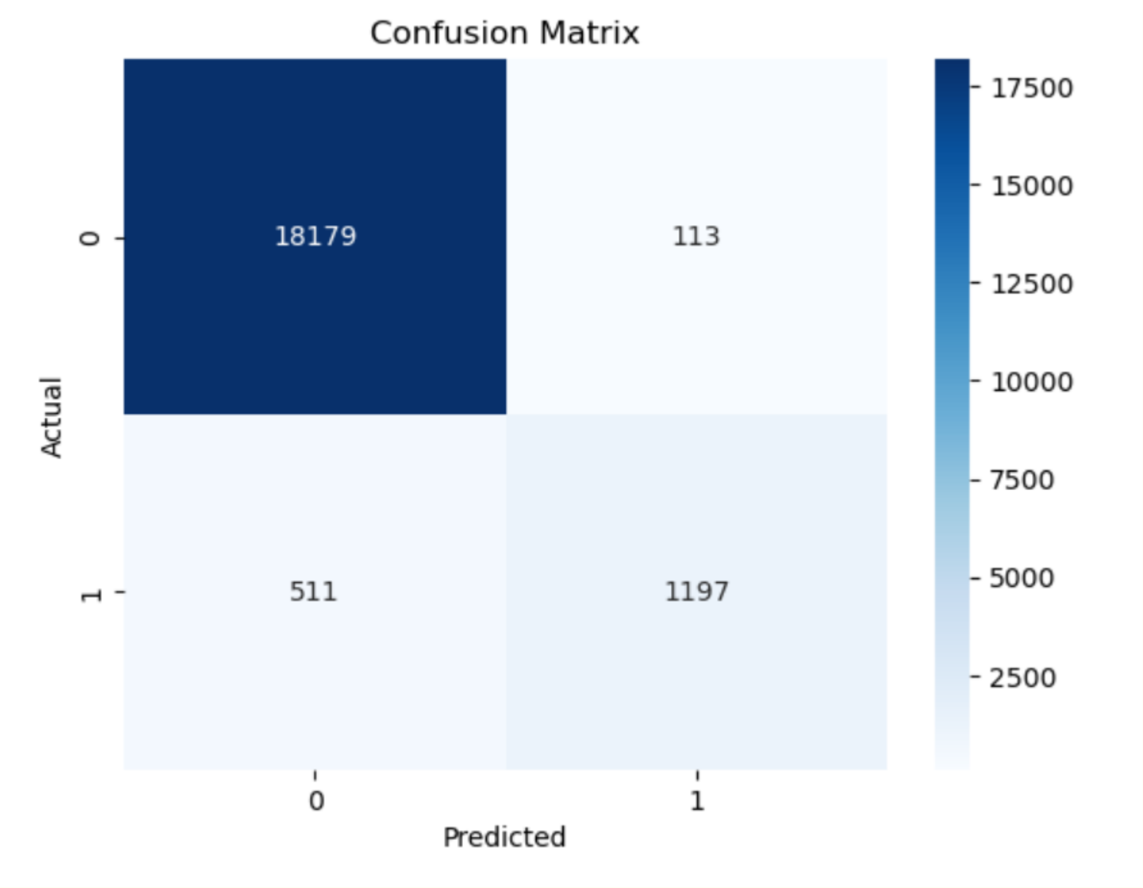
Predictive Models



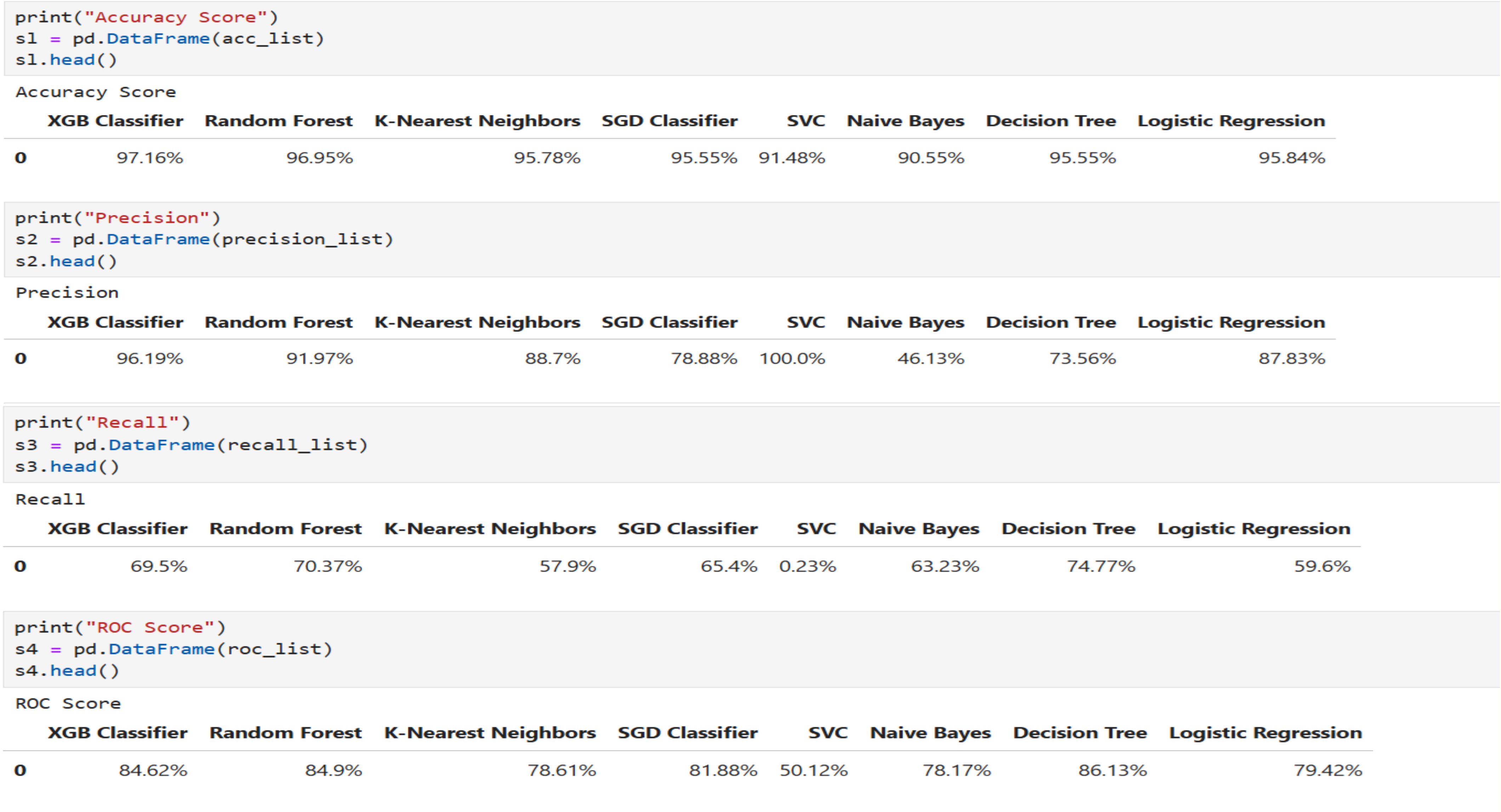
```
# Random Forest Classifier

rfc = RandomForestClassifier()
rfc.fit(x_train, y_train)
rfy_pred = rfc.predict(x_test)
print("Logistic Regression")
print("Accuracy:", accuracy_score(y_test, rfy_pred))
print("Precision:", precision_score(y_test, rfy_pred))
print("Recall:", recall_score(y_test, rfy_pred))
print("F1-score:", f1_score(y_test, rfy_pred))
print("AUC-ROC:", roc_auc_score(y_test, rfy_pred))
```



```
Logistic Regression
Accuracy: 0.9688
Precision: 0.9137404580152672
Recall: 0.7008196721311475
F1-score: 0.7932405566600398
AUC-ROC: 0.8473210540843797
```



Applied 8 ML Algorithms to the dataset



```python
print("Accuracy Score")
s1 = pd.DataFrame(acc_list)
s1.head()
```

```
Accuracy Score
```

	XGB Classifier	Random Forest	K-Nearest Neighbors	SGD Classifier	SVC	Naive Bayes	Decision Tree	Logistic Regression
0	97.16%	96.95%	95.78%	95.55%	91.48%	90.55%	95.55%	95.84%

```python
print("Precision")
s2 = pd.DataFrame(precision_list)
s2.head()
```

```
Precision
```

	XGB Classifier	Random Forest	K-Nearest Neighbors	SGD Classifier	SVC	Naive Bayes	Decision Tree	Logistic Regression
0	96.19%	91.97%	88.7%	78.88%	100.0%	46.13%	73.56%	87.83%

```python
print("Recall")
s3 = pd.DataFrame(recall_list)
s3.head()
```

```
Recall
```

	XGB Classifier	Random Forest	K-Nearest Neighbors	SGD Classifier	SVC	Naive Bayes	Decision Tree	Logistic Regression
0	69.5%	70.37%	57.9%	65.4%	0.23%	63.23%	74.77%	59.6%

```python
print("ROC Score")
s4 = pd.DataFrame(roc_list)
s4.head()
```

```
ROC Score
```

	XGB Classifier	Random Forest	K-Nearest Neighbors	SGD Classifier	SVC	Naive Bayes	Decision Tree	Logistic Regression
0	84.62%	84.9%	78.61%	81.88%	50.12%	78.17%	86.13%	79.42%

Conclusion and Recommendation

- According to the dataset, most patients were female
- Only a small proportion (8.5%) of patients have diabetes
- The disease was more prevalent among the adult (36 yrs above) population
- There was no strong correlation between the patient's smoking history and diabetes
- There was a high level of blood glucose among patients
- The XGB Classifier Model proves to be a better model with an accuracy of 96.12%.
- The most important metrics based on the models executed are accuracy and precision
- However, 551 patients (false negative) were wrongly predicted
- Hence, more attention and further analysis is required on the 551 falsely predicted patients.