# Model for Diabetes Prediction Using LightGBM Classifier

**D.HARSHITHA[1], K.DIVYA[2], A.SATYA ANUSHA[3], M.V.HARINI[4]**

[1]BVRIT HYDERABAD College of Engineering for Women,Bachupally, 8-5/4, Nizampet Rd, Opposite Rajiv Gandhi Nagar Colony, Hyderabad, Telangana 500090
[2]Department of Information Technology.(e-mail: 18wh1a1201@bvrithyderabad.edu.in)
[3]Department of Information Technology.(e-mail: 18wh1a1258@bvrithyderabad.edu.in)
[4]Department of Information Technology.(e-mail: 18wh1a1253@bvrithyderabad.edu.in)
[5]Department of Information Technology.(e-mail: 18wh1a1233@bvrithyderabad.edu.in)

**ABSTRACT** Diabetes mellitus is a chronic disease characterized by hyperglycemia. It may cause many complications. According to the growing morbidity in recent years, in 2040, the world's diabetic patients will reach 642 million, which means that one of the ten adults in the future is suffering from diabetes. There is no doubt that this alarming figure needs great attention. With the rapid development of the latest technology , many aspects of medical health can be improved . In this research, we focused on models to determine whether a patient admitted to an ICU has been diagnosed with a particular type of diabetes, Diabetes mellitus. The dataset contains 130,157 patients data, where 70 percent are used for training and 30 percent are used for testing. Since the dataset has 371 attributes, we used feature engineering to optimize the data and then implemented various Machine Learning algorithms like XGB, Random forest and LightGBM to predict diabetes mellitus which resulted in AUC score 0.86. For improving results we have applied Hyper Parameter tuning which resulted in 0.872 AUC score.

## I. INTRODUCTION

The evolution in the digital era has led to the confluence of healthcare and technology resulting in the emergence of newer data-related applications. Diabetes is a common chronic disease and poses a great threat to human health. The characteristic of diabetes is that the blood glucose is higher than the normal level, which is caused by defective insulin secretion or its impaired biological effects, or both . Diabetes can lead to chronic damage and dysfunction of various tissues, especially eyes, kidneys, heart, blood vessels and nerves.In medicine, the diagnosis of diabetes is according to fasting blood glucose, glucose tolerance, and random blood glucose levels. The constant hyperglycemia of diabetes is related to long-haul harm, brokenness, and failure of various organs, particularly the eyes, kidneys, nerves, heart, and veins. The earlier the diagnosis is obtained, the much easier we can control it. Machine learning can help people make a preliminary judgment about diabetes mellitus according to their daily physical examination data, and it can serve as a reference for doctors. Doctors rely on common knowledge for treatment. When common knowledge is lacking, studies are summarized after some number of cases have been studied. But this process takes time, whereas if machine learning is used, the patterns can be identified earlier. The huge volume of data can be pooled together and analyzed effectively using machine-learning algorithms. Analyzing the details and understanding the patterns in the data can help in better decision-making resulting in a better quality of patient care. The objective of this paper is to make use of significant features, design a prediction algorithm using Machine learning and find the optimal classifier to give the closest result comparing to clinical outcomes.This work focuses on improvising the outcome of medical care, life expectancy, early detection, and identification of diabetes mellitus disease at an initial stage and required treatment at an affordable cost.

## II. RESEARCH

The Acute Physiology and Chronic Health Evaluation (APACHE II) is a severity score and mortality estimation tool developed from a large sample of ICU patients in the United States.

1) APACHE II score: It is a general measure of disease severity based on current physiologic measurements, age previous health conditions. The score can help in the assessment of patients to determine the level degree of diagnostic therapeutic intervention.
2) Oxygenation Index: The oxygenation index or Horowitz index is the ratio of partial pressure of oxygen in blood (PaO2) and the fraction of oxygen in the inhaled air (FiO2). Therefore, we can fill the missing values of d1_pao2fio2ratio_max with pao2_apache/fio2_apache.

3) Encoding: The purpose of encoding is to transform data so that it can be properly (and safely) consumed by a different type of system, e.g. binary data being sent over email, or viewing special characters on a web page. The goal is not to keep information secret, but rather to ensure that it's able to be properly consumed.
4) Stratified K-Folds cross-validation : This helps train/test indices to split data in train/test sets. This cross-validation object is a variation of KFold that returns stratified folds. The folds are made by preserving the percentage of samples for each class.
5) Hyperparameter optimization : In machine learning, hyperparameter optimization or tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm. A hyperparameter is a parameter whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are learned. Specifically, the learning rate is a configurable hyperparameter used in the training of neural networks that has a small positive value, often in the range between 0.0 and 1.0. The learning rate controls how quickly the model is adapted to the problem.

## III. DATA DESCRIPTION

This data is obtained from MIT's GOSSIS (Global Open Source Severity of Illness Score) initiative. The dataset consists of 180 features. It has 1,30,157 rows and 317 columns with different data types like int, float, string etc. According to the dictionary, there are 7 categories of features in the data. Due to a large number of features in the dataset, a category-wise analysis of the data was done.

A closer look into the data shows that our data is imbalanced.There are a lot of missing values in the feature set. It is also observed that some features are missing in pairs. For example, the number of missing values for d1_glucose_max and d1_glucose_min is the same. We also noticed there are a lot of common values in some apache and vital columns.

## IV. METHODOLOGY

This approach goes initially, by understanding the dataset, exploring the attributes present in the dataset for focusing on necessary features. Next analysis phase comes where preprocessing of data happens by removing the missing values, replacing the NaN values with the appropriate ones etc. Learning algorithms like random forest, linear regression, Lgbm, xg-boost, etc to predict whether the person is suffering from diabetes or not.

1) Exploratory Data Analysis : In this section, some basic Exploratory Data Analysis has been done to get the "feel" of the data, by checking the distributions, the correlations etc of the different columns and trying to remove the null values present.
2) Data Processing : As the base dataset, the PIMA Indian dataset is analyzed for finding the missing columns,correlated features and outliers. Below is the image of the columns having missing values and percentage of it.

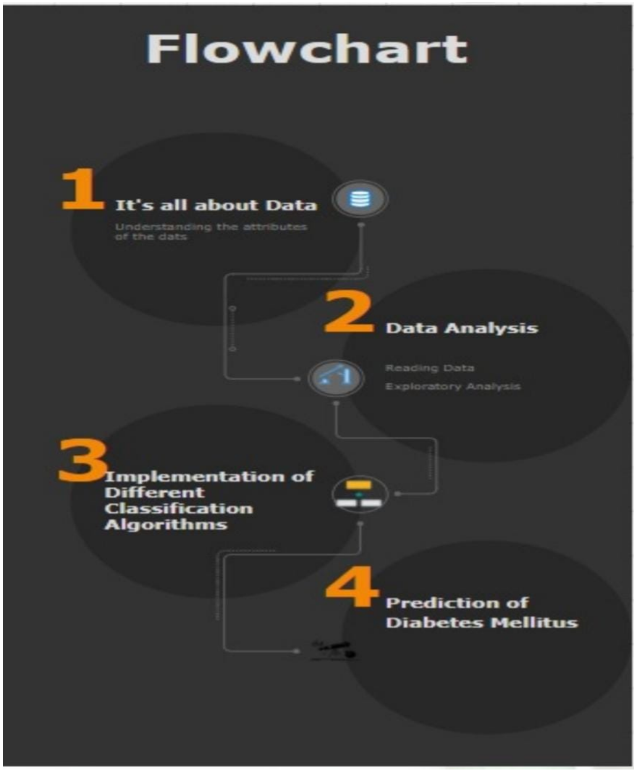

Flowchart
1
It's all about Data
Understanding the attributes of the dats
2
Data Analysis
Reading Data
Exploratory Analysis
3
Implementation of Different Classification Algorithms
4
Prediction of Diabetes Mellitus

FIGURE 1. Flow of the model

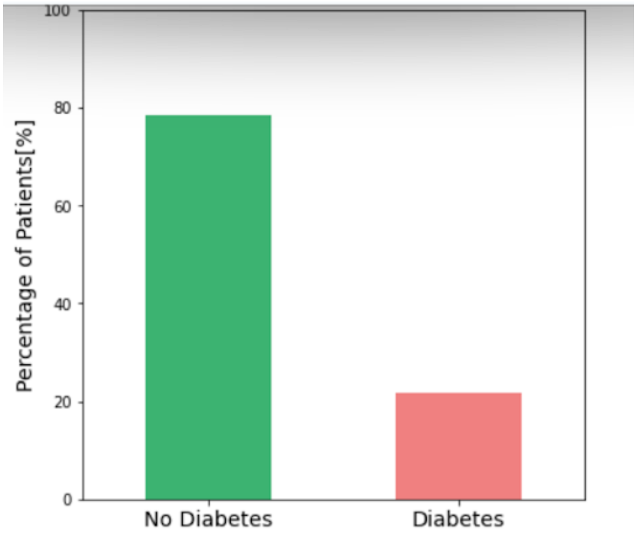


FIGURE 2. Dataset visualization

3) Feature Engineering :
Depending on the model, the dataset may contain too many features for it to handle.Hence features with less importances can be eliminated. The below figure shows the sorted features according to their importance.
4) Model Building :

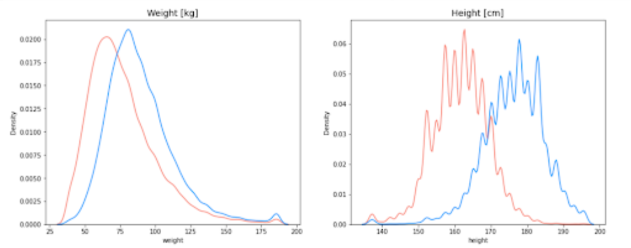
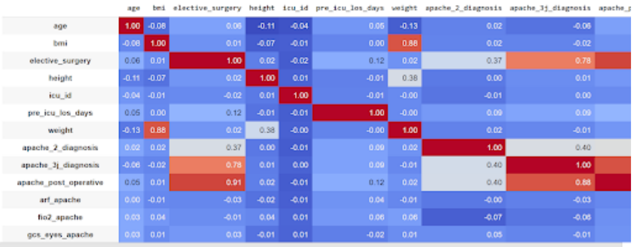
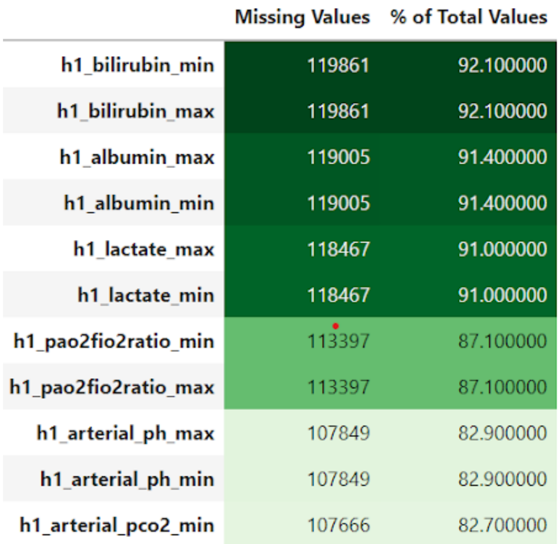


FIGURE 3. Data visualization with respective to height and weight



| | age | bmi | elective_surgery | height | icu_id | pre_icu_los_days | weight | apache_2_diagnosis | apache_3j_diagnosis | apache_p |
|---|---|---|---|---|---|---|---|---|---|---|
| age | 1.00 | -0.08 | 0.06 | -0.11 | -0.04 | 0.05 | -0.13 | 0.02 | -0.06 | |
| bmi | -0.08 | 1.00 | 0.01 | -0.07 | -0.01 | 0.00 | 0.88 | 0.02 | -0.02 | |
| elective_surgery | 0.06 | 0.01 | 1.00 | 0.02 | -0.02 | 0.12 | 0.02 | 0.37 | 0.78 | |
| height | -0.11 | -0.07 | 0.02 | 1.00 | 0.01 | -0.01 | 0.38 | 0.00 | 0.01 | |
| icu_id | -0.04 | -0.01 | -0.02 | 0.01 | 1.00 | -0.01 | -0.00 | -0.01 | 0.00 | |
| pre_icu_los_days | 0.05 | 0.00 | 0.12 | -0.01 | -0.01 | 1.00 | -0.00 | 0.09 | 0.09 | |
| weight | -0.13 | 0.88 | 0.02 | 0.38 | -0.00 | -0.00 | 1.00 | 0.02 | -0.01 | |
| apache_2_diagnosis | 0.02 | 0.02 | 0.37 | 0.00 | -0.01 | 0.09 | 0.02 | 1.00 | 0.40 | |
| apache_3j_diagnosis | -0.06 | -0.02 | 0.78 | 0.01 | 0.00 | 0.09 | -0.01 | 0.40 | 1.00 | |
| apache_post_operative | 0.05 | 0.01 | 0.91 | 0.02 | -0.01 | 0.12 | 0.02 | 0.40 | 0.88 | |
| arf_apache | 0.00 | -0.01 | -0.03 | -0.02 | -0.01 | 0.04 | -0.01 | -0.00 | -0.03 | |
| fio2_apache | 0.03 | 0.04 | -0.01 | 0.04 | 0.01 | 0.06 | 0.06 | -0.07 | -0.06 | |
| gcs_eyes_apache | 0.03 | 0.01 | 0.03 | -0.01 | 0.01 | -0.02 | 0.01 | 0.05 | -0.01 | |

FIGURE 4. Correlation of features.



| | Missing Values | % of Total Values |
|---|---|---|
| h1_bilirubin_min | 119861 | 92.100000 |
| h1_bilirubin_max | 119861 | 92.100000 |
| h1_albumin_max | 119005 | 91.400000 |
| h1_albumin_min | 119005 | 91.400000 |
| h1_lactate_max | 118467 | 91.000000 |
| h1_lactate_min | 118467 | 91.000000 |
| h1_pao2fio2ratio_min | 113397 | 87.100000 |
| h1_pao2fio2ratio_max | 113397 | 87.100000 |
| h1_arterial_ph_max | 107849 | 82.900000 |
| h1_arterial_ph_min | 107849 | 82.900000 |
| h1_arterial_pco2_min | 107666 | 82.700000 |

FIGURE 5. List of missing columns with their details

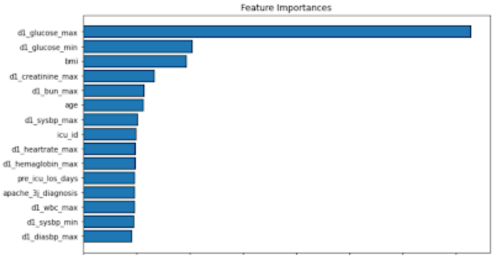


FIGURE 6. important features

After extracting the new columns from the data all the unnecessary columns and correlated columns are dropped. Features with less importance, target permutations are dropped and some that are highly correlated or have different distributions in train and test datasets have also been dropped. Categorical columns have been encoded with the get_dummies function.Finally the data was given as input to a LightGBM Classifier.

5) Performance :
The performance metric used here is AUC(Area Under the Curve).The performance of the model was measured using the average AUC metric after all folds.The average fold AUC for the resultant model was 0.86542 in this case.

6) Hyperparameter Tuning :
Hyperparameter optimization or tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm. The hyperparameters like weights,learning rates are tuned in the model to generalize different optimal data patterns. Hyperparameter optimization finds a tuple of hyperparameters that yields an optimal model which minimizes a predefined loss function on given independent data. The performance of the model after tuning the data was measured using the average AUC metric and the result was 0.8724.

## V. RESULTS

The Results of the Model by using various machine learning algorithms can be considered in two phases:

- Prediction
- AUC Score

XGB: XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. XGBoost is an algorithm that has recently been dominating applied machine learning algorithms.

The results after using the XGB Algorithm on this data are shown in fig 7 and the measured AUC score is 0.847

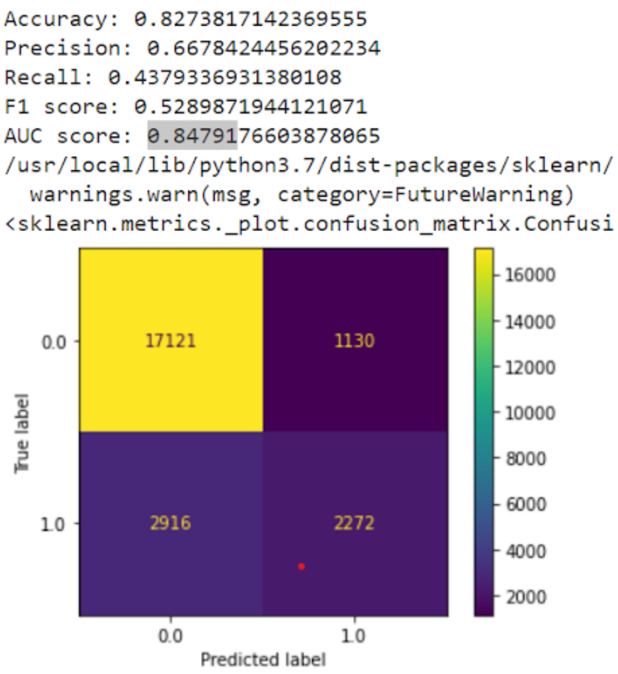
Light GBM: Light GBM is a fast, distributed, high-performance gradient boosting framework based on a decision tree algorithm, used for ranking, classification and many other machine learning tasks to increase the efficiency of the model and reduce memory usage. LightGBM splits the tree leaf-wise as opposed to other boosting algorithms that grow tree level-wise.

The results after using the XGB Algorithm on this data are shown in fig 8 and the measured AUC score is 0.86509.

The performance of XGB after implementing Hyper Parameter Tuning is shown in fig 9 and the measured AUC score is 0.8671.

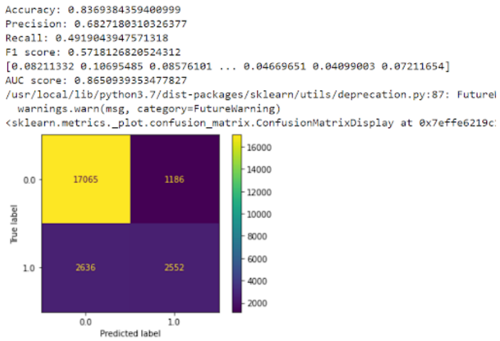
The performance of Light GBM after implementing Hyper Parameter Tuning is shown in fig 10 and the measured AUC score is 0.87246.

Streamlit: Streamlit is an open source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. The implementation of the predictions in Streamlit are shown in below figures.



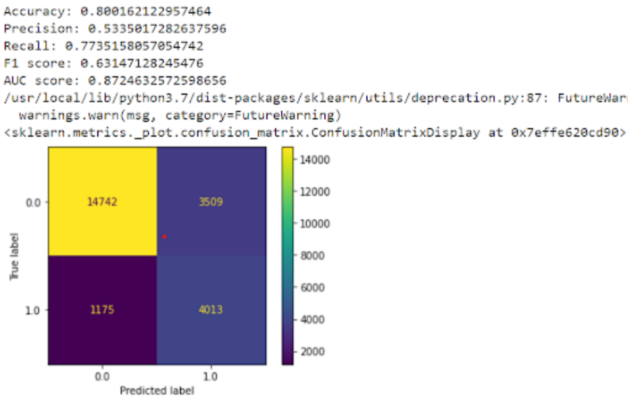
```
Accuracy: 0.8273817142369555
Precision: 0.6678424456202234
Recall: 0.4379336931380108
F1 score: 0.5289871944121071
AUC score: 0.8479176603878065
/usr/local/lib/python3.7/dist-packages/sklearn/
  warnings.warn(msg, category=FutureWarning)
<sklearn.metrics._plot.confusion_matrix.Confusi
```

FIGURE 7. performance of XGB Model



```
Accuracy: 0.8369384359400999
Precision: 0.6827180310326377
Recall: 0.4919043947571318
F1 score: 0.5718126820524312
[0.08211332 0.10695485 0.08576101 ... 0.04669651 0.04099003 0.07211654]
AUC score: 0.8650939353477827
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureW
  warnings.warn(msg, category=FutureWarning)
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7effe6219c
```

FIGURE 8. performance of Light GBM Model

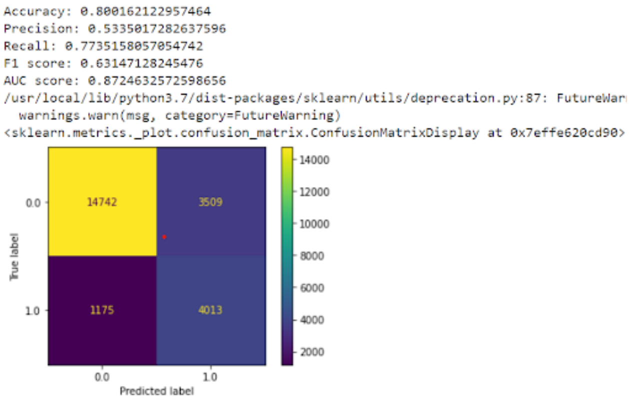


```
Accuracy: 0.800162122957464
Precision: 0.5335017282637596
Recall: 0.7735158057054742
F1 score: 0.63147128245476
AUC score: 0.8724632572598656
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWar
  warnings.warn(msg, category=FutureWarning)
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7effe620cd90>
```

FIGURE 9. performance of XGB with Hyperparameter Tuning

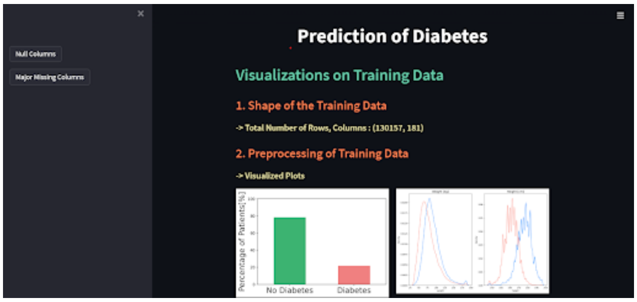
## VI. CONCLUSION

The Experimental results can assist health care to take early prediction and make early decision to cure diabetes and save humans life. As of now the risk of diabetes was predicted using different algorithms like XGBoost and Light GBM. LightGBM classifier performs better on integer encoded categorical values.Some techniques like resampling, extraction of indicators using the blood pressure columns into new features, missing value handling of BMI and Blood



```
Accuracy: 0.800162122957464
Precision: 0.5335017282637596
Recall: 0.7735158057054742
F1 score: 0.63147128245476
AUC score: 0.8724632572598656
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWar
  warnings.warn(msg, category=FutureWarning)
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7effe620cd90>
```

FIGURE 10. performance of Light GBM with Hyperparameter Tuning



Prediction of Diabetes

Visualizations on Training Data

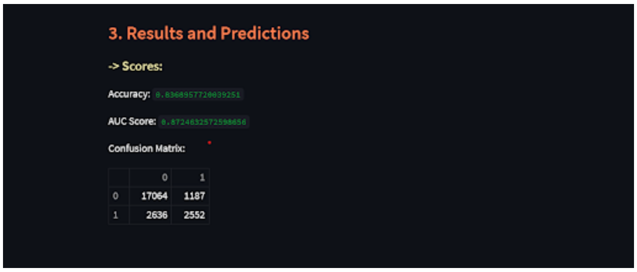
1. Shape of the Training Data

-> Total Number of Rows, Columns : (130157, 181)

2. Preprocessing of Training Data

-> Visualized Plots

FIGURE 11. Streamlit Output



3. Results and Predictions

-> Scores:

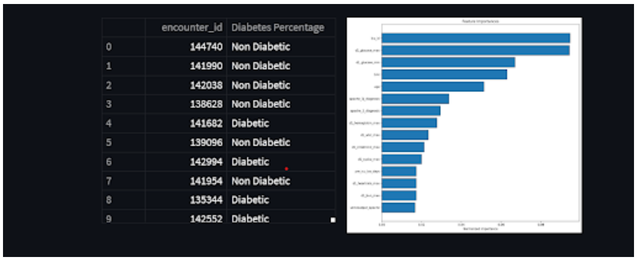
Accuracy: 0.8368957720039251

AUC Score: 0.8724632572598656

Confusion Matrix:

|   | 0 | 1 |
|---|---|---|
| 0 | 17064 | 1187 |
| 1 | 2636 | 2552 |

FIGURE 12. Result displayed using streamlit



|   | encounter_id | Diabetes Percentage |
|---|---|---|
| 0 | 144740 | Non Diabetic |
| 1 | 141990 | Non Diabetic |
| 2 | 142038 | Non Diabetic |
| 3 | 138628 | Non Diabetic |
| 4 | 141682 | Diabetic |
| 5 | 139096 | Non Diabetic |
| 6 | 142994 | Diabetic |
| 7 | 141954 | Non Diabetic |
| 8 | 135344 | Diabetic |
| 9 | 142552 | Diabetic |

FIGURE 13. Features shown using streamlit

pressure columns, 5-fold cross validation are also the major reasons behind the better accuracy. A multicenter study with more variables can make a large difference in efficiency of results.implementation of feature engineering with use of domain knowledge extract features from raw data. The motivation is to use these extra features to improve the quality of results. Hyper Parameter Optimization helps in choosing hyper parameters which are used in controlling the learning process. A combination of both feature engineering and Hyper Parameter Optimization can result in best accurate results. Streamlit is used to build a interactive Data Science web application with all visualization part and predictions.

## VII. REFERENCES

1) N. Sneha Tarun Gangil , “Analysis of diabetes mellitus for early prediction using optimal features selection” Journal of Big Data, 2019.
2) Quan Zou1, Kaiyang Qu, Yamei Luo, Dehui Yin, “Predicting Diabetes Mellitus With Machine Learning Techniques”Front. Genet, 2018.
3) DeeptiSisodia, aDilip SinghSisodia “Prediction of Diabetes using Classification Algorithms” Procedia Computer Science Volume 132 2018.
4) Diabetes prediction model based on an enhanced deep neural network, Huaping Zhou, Raushan Myrzashova Rui Zheng, EURASIP Journal on Wireless Communications and Networking volume 2020, Article number: 148 ,2020
5) Machine Learning Based Diabetes Classification and Prediction for Healthcare Applications, Umair Muneer Butt , Sukumar Letchmunan ,Mubashir Ali ,Fadratul Hafinaz Hassan, Volume 2021,
6) Research on Diabetes Prediction Method Based on Machine Learning, Jingyu Xue,a, Fanchao Min,b, Fengying, Journal of Physics: Conference Series, 2020
7) LGBM Classifier based Technique for Predicting Type-2 Diabetes, B. Shamreen Ahamed Dr. Meenakshi Sumeet Arya, European Journal of Molecular Clinical Medicine, Volume 08, Issue 03, 2021.
8) Prediction of Gestational Diabetes Based on LightGBM, Fan Hou, ZhiXhang Cheng, 2020