

Prediction of cognitive impairment using type 2 diabetes dataset using machine learning

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Engineering degree in Computer Science and Engineering

by

Sugandham Satya Sai(37110741)
Vanga Mohan Brahma Harsha(37110806)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC I12B Status by UGC Approved by AICTE

JEPPIAARNAGAR, RAJIV GANDHI SALAI, CHENNAI-600119

March- 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC I12B Status by UGC I Approved by AICTE

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONA FIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Satya Sai (37110741) and Brahma Harsha (37110806) who carried out the project entitled "**Prediction of cognitive impairment using type 2 diabetes dataset using machine learning**" under my supervision from November 2020 to March 2021.

Internal Guide

Head of the Department
(No need to specify the name)

Submitted for Vivavoce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We **S Satya Sai** (Reg. No. 37110741) and **Brahma Harsha**(Reg. No. 37110806) hereby declare that the Project Report entitled “**Prediction of cognitive impairment using type 2 diabetes dataset using machine learning**” done by us under the guidance of **Mrs M S RoobiniM.E.,(Ph.D)**. is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATUREOFTHECANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. T. Sasikala M.E., Ph.D., Dean, School of Computing and Dr. L. Lakshmanan M.E., Ph.D., and Dr. S. Vigneshwari M.E., Ph.D., Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Mrs M S Roobini M.E., (Ph.D.) for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Due to its ceaseless increasing occurrence, more and more families are influenced by this diabetes. Most diabetics do not know about their health quality or the risk factors they face prior to diagnosis. In this case we proposed a model for predicting this type 2 diabetes. The problem that we are trying to solve and improve the accuracy of this model and make this model adaptive. By comparing with different algorithms and their accuracy we have to choose the best algorithm for predicting this type 2 diabetes. Our model ensures that dataset quality is sufficient. And its shows good performance in predicting the type 2 diabetes with higher accuracy. As a result our model is preferable to realistic health management of diabetes.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF ABBREVIATIONS	vii
	LIST OF FIGURES	viii
1.	INTRODUCTION	01
	1.1. INTRODUCTION	01
2.	LITERATURE SURVEY	03
	2.1. RELATED WORK	03
3.	AIM AND SCOPE	04
	3.1. AIM OF THE PROJECT	04
	3.2. OBJECTIVE AND SCOPE	04
4.	SYSTEM IMPLEMENTATION	05
	4.1. SYSTEM ARCHITECTURE	05
	4.2. METHODS AND MODEL DETAILS	06
5.	RESULTS AND DISCUSSION	14
6.	SUMMARY AND CONCLUSION	21
7.	APPENDIX	23
	A) SOURCE CODE	23
	LIST OF SYMBOLS AND ABBREVIATIONS	
	B) REFERENCES	32

ABBREVIATION	FULLFORM
ML	MACHINE LEARNING
SVMS	SUPPORT VECTOR MACHINE

LIST OF FIGURES

FIG.NO	FIG NAME	PAGE NO
4.1.1	ARCHITECTURE DIAGRAM	05
4.2.2.1	RANDOM FOREST CLASSIFIER	10
4.2.2.2	DECISION TREE	12

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Diabetes is an illness caused because of high glucose level in a human body. Diabetes should not be ignored if it is untreated then Diabetes may cause some major issues in a person like: heart related problems, kidney problem, blood pressure, eye damage and it can also affects other organs of human body. Diabetes can be controlled if it is predicted earlier. To achieve this goal this project work we will do early prediction of Diabetes in a human body or a patient for a higher accuracy through applying, Various Machine Learning Techniques. Machine learning techniques Provide better result for prediction by constructing models from datasets collected from patients. Nowadays, diabetes is the most common disease that is faced by majority of the population. Basically there are two types of diabetes, they are type 1 and type 2. Our project is regarding the prediction of type 2 diabetes. This model is for predicting exact result to the particular individual which helps in controlling deaths due to diabetes. A type 2 diabetic can get a proper treatment for his/her disease which helps them to control their disease and keep them healthy. In this type 2, insulin is produced in human body but its usage will not be effective. There are several symptoms for this diabetes like increased thirst, frequent urination, hunger, fatigue and blurred vision. Various factors are gathered from the person to predict whether him/her is a diabetic or not. Diabetes is noxious diseases in the world. Diabetes caused because of obesity or high blood glucose level, and so forth. It affects the hormone insulin, resulting in abnormal metabolism of carbs and improves level of sugar in the blood. Diabetes occurs when body does not make enough insulin. According to (WHO) World Health Organization about 422 million people suffering from diabetes particularly from low or idle income countries. And this could be increased to 490 billion up to the year of 2030. However prevalence of diabetes is found among various Countries like Canada, China, and India etc. Population of India is now more than 100 million so the actual number of diabetics in India is 40 million. Diabetes is major cause of death in the world. Early prediction of disease like diabetes can be controlled and save the human life. To accomplish this, this work explores prediction of diabetes by taking various attributes related to diabetes disease.

CHAPTER 2

LITERATURE SURVEY

2.1 RELATED WORK

K.VijayaKumar proposed random Forest algorithm for the Prediction of diabetes develop a system which can perform early prediction of diabetes for a patient with a higher accuracy by using Random Forest algorithm in machine learning technique. The proposed model gives the best results for diabetic prediction and the result showed that the prediction system is capable of predicting the diabetes disease effectively, efficiently and most importantly, instantly. Nonso Nnamoko presented predicting diabetes onset: an ensemble supervised learning approach they used five widely used classifiers are employed for the ensembles and a meta-classifier is used to aggregate their outputs. The results are presented and compared with similar studies that used the same dataset within the literature. It is shown that by using the proposed method, diabetes onset prediction can be done with higher accuracy. TejasN. Joshi presented Diabetes Prediction Using Machine Learning Techniques aims to predict diabetes via three different supervised machine learning methods including: SVM, Logistic regression, ANN. This project proposes an effective technique for earlier detection of the diabetes disease. Deeraj Shetty proposed diabetes disease prediction using data mining assemble Intelligent Diabetes Disease Prediction System that gives analysis of diabetes malady utilizing diabetes patients database. In this system, they propose the use of algorithms like Bayesian and KNN (K-Nearest Neighbour) to apply on diabetes patients database and analyze them by taking various attributes of diabetes for prediction of diabetes disease. Muhammad Azeem Sarwar proposed study on prediction of diabetes using machine learning algorithms in healthcare they applied six different machine learning algorithms Performance and accuracy of the applied algorithms is discussed and compared. Comparison of the different machine learning techniques used in this study reveals which algorithm is best suited for prediction of diabetes. Diabetes Prediction is becoming the area of interest for researchers in order to train the program to identify the patient are diabetic or not by applying proper classifier on the dataset. Based on previous research work, it has been observed that the classification

process is not much improved. Hence a system is required as Diabetes Prediction is important area in computers, to handle the issues identified based on previous research.

CHAPTER 3

AIM AND SCOPE

3.1 AIM OF THE PROJECT

Aim of this project is to investigate for model to predict diabetes with better accuracy. We experimented with different classification and ensemble algorithms to predict diabetes. By taking the dataset from diabetic centres and tested. On applying different algorithms we will get different outcomes.

3.2 OBJECTIVE AND SCOPE

To predict the outcome of Diabetes. It also aims to analyse Diabetes causing factors. The data has to be pre-processed and fed to various supervised machine learning algorithms and analysed in accordance to their accuracies. The best possible outcome will be predicted using a perfect model.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 SYSTEM ARCHITECTURE:

The proposed system aims to analyse the data collected from diabetic centres .The steps followed are –

- Collecting data by scraping
- Data pre-processing
- Prediction using supervised learning algorithms

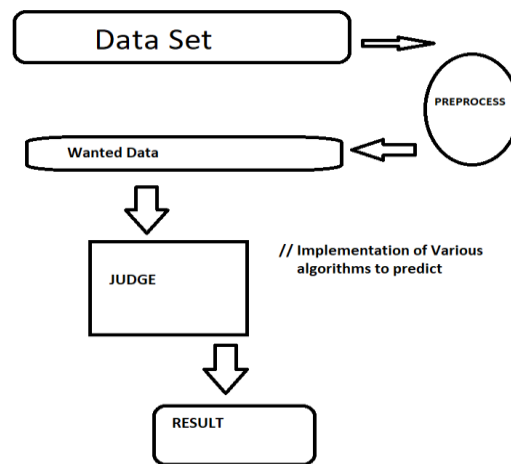


Figure 4.1.1: Architecture Diagram

4.2 METHODS AND MODEL DETAILS:

Logistic Regression: Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm and based on the concept of probability. We can call a Logistic Regression a Linear Regression model but the Logistic

Regression uses a more complex cost function, this cost function can be defined as the '**Sigmoid function**' or also known as the 'logistic function' instead of a linear function. The hypothesis of logistic regression tends to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

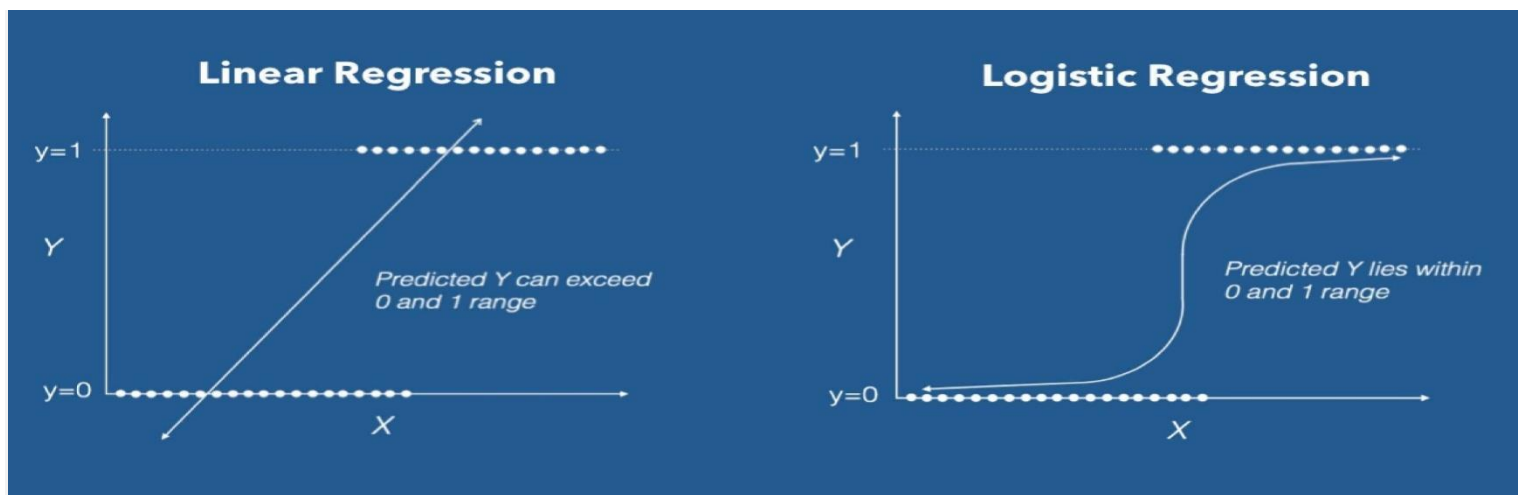


Figure : Logistic Regression

Random Forest Algorithm: Random forest algorithm is a flexible machine learning algorithm that produces great results even without hyper-parameter tuning. Apart from being simple to use, it is extremely accurate also. It is basically a supervised learning algorithm. A large number of decision trees operate together; an individual tree in a random forest model gives some prediction and finally the one with most votes becomes the prediction of the model. Random forest can be used for both classification as well as regression problems. A large number of relatively uncorrelated trees operating as a committee will perform better than individual constituent tree. Uncorrelated trees can

produce better and more accurate results than individual decision trees, the reason for this is that the trees protect each other from their individual errors, thus even if some trees in the group are wrong, all the trees are able to move in correct direction given that many other trees will be right.

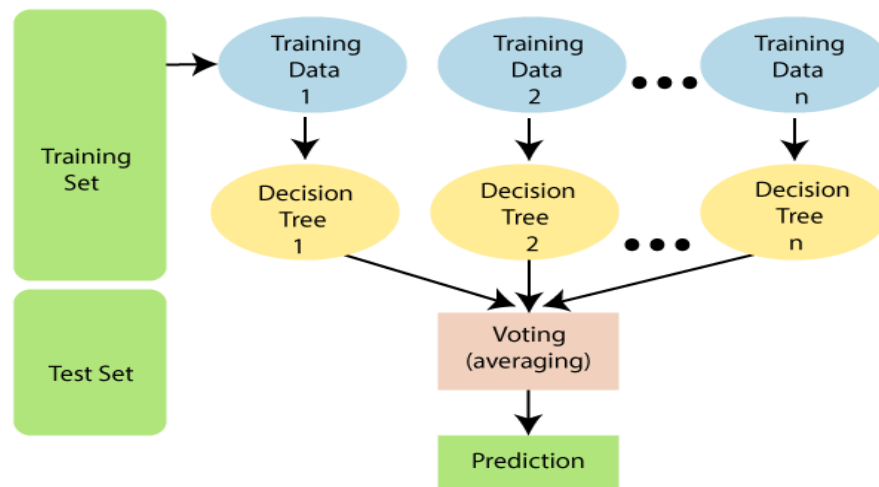


Figure 4.2.2.1: Random Forest Classifier

Random Forest works in four steps:

- Select random samples from a given dataset.
- Construct a decision tree for each sample and get a prediction result from each decision tree.
- Perform a vote for each predicted result.
- Select the prediction result with the most votes as the final prediction.

Decision Trees:

The classification technique is a systematic approach to build classification models from an input dataset. For example, decision tree classifiers, rule-based classifiers, neural networks, support vector machines, and naive Bayes classifiers are different techniques to solve a classification problem. Each technique adopts a learning algorithm to identify a model that best fits the relationship between the attribute set and class label of the input data. Therefore, a key objective of the learning algorithm is to build a predictive model that accurately predicts the class labels of previously unknown records.

Decision Tree Classifier is a simple and widely used classification technique. It applies a straightforward idea to solve the classification problem. Decision Tree Classifier poses a series of carefully crafted questions about the attributes of the test record. Each time it receives an answer, a follow-up question is asked until a conclusion about the class label of the record is reached.

Build an optimal decision tree is key problem in decision tree classifier. In general, many decision trees can be constructed from a given set of attributes. While some of the trees are more accurate than others, finding the optimal tree is computationally infeasible because of the exponential size of the search space.

However, various efficient algorithms have been developed to construct a reasonably accurate, albeit suboptimal, decision tree in a reasonable amount of time. These algorithms usually employ a greedy strategy that grows a decision tree by making a series of locally optimum decisions about which attribute to use for partitioning the data. For example, Hunt's algorithm, ID3, C4.5, CART, SPRINT are greedy decision tree induction algorithms.

The decision tree inducing algorithm must provide a method for specifying the test condition for different attribute types as well as an objective measure for evaluating the goodness of each test condition.

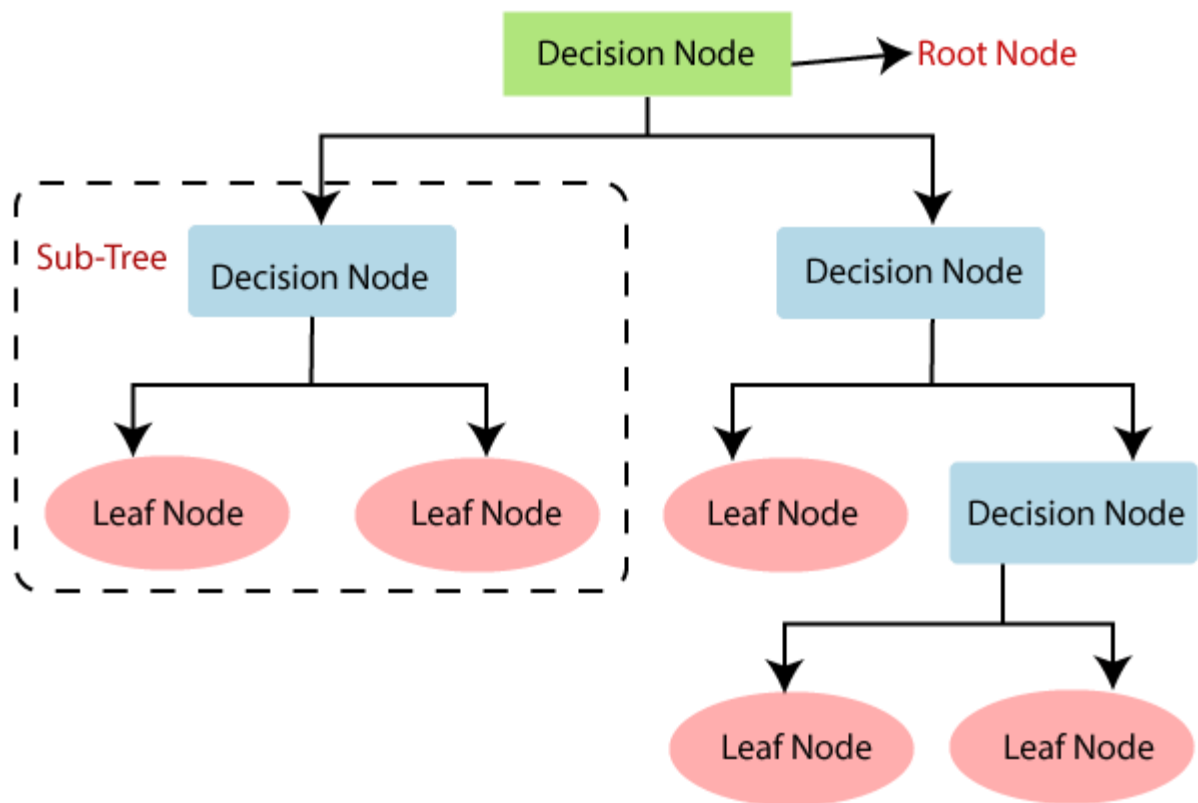


Figure 4.2.2.2: Decision Tree

First, the specification of an attribute test condition and its corresponding outcomes depends on the attribute types. We can do two-way split or multi-way split, discretize or group attribute values as needed. The binary attributes lead to two-way split test condition. For nominal attributes which have many values, the test condition can be expressed into multi way split on each distinct value, or two-way split by grouping the attribute values into two subsets. Similarly, the ordinal attributes can also produce binary or multi way splits as long as the grouping does not violate the order property of the attribute values. For continuous attributes, the test condition can be expressed as a comparison test with two outcomes, or a range query. Or we can discretize the continuous value into nominal attribute and then perform two-way or multi-way split.

Since there are many choices to specify the test conditions from the given training set, we need use a measurement to determine the best way to split the records. The goal of best test conditions is whether it leads a homogenous class distribution in the nodes, which is the purity of the child nodes before and after splitting. The larger the degree of purity, the better is the class distribution.

To determine how well a test condition performs, we need to compare the degree of impurity of the parent before splitting with degree of the impurity of the child nodes after splitting. The larger their difference, the better is the test condition. The measurements of node impurity/purity are:

- Gini Index
- Entropy
- Misclassification Error

$$\text{Gini index: } G(E) = 1 - \sum_{j=1}^C p_j^2,$$

$$\text{entropy: } H(E) = - \sum_{j=1}^C p_j \log p_j.$$

Support Vector Machine: Support Vector Machine also known as svm is a supervised machine learning algorithm. Svm is most popular classification technique. Svm creates a hyperplane that separate two classes. It can create a hyperplane or set of hyperplane in high dimensional space. This hyper plane can be used for classification or regression also. Svm differentiates instances in specific classes and can also classify the entities which are not supported by data. Separation is done by through hyperplane performs the separation to the closest training point of any class.

Algorithm-

- Select the hyper plane which divides the class better.
- To find the better hyper plane you have to calculate the distance between the planes and the data which is called Margin.
- If the distance between the classes is low then the chance of miss conception is high and vice versa. So, we need to
- Select the class which has the high margin. Margin = distance to positive point + Distance to negative point

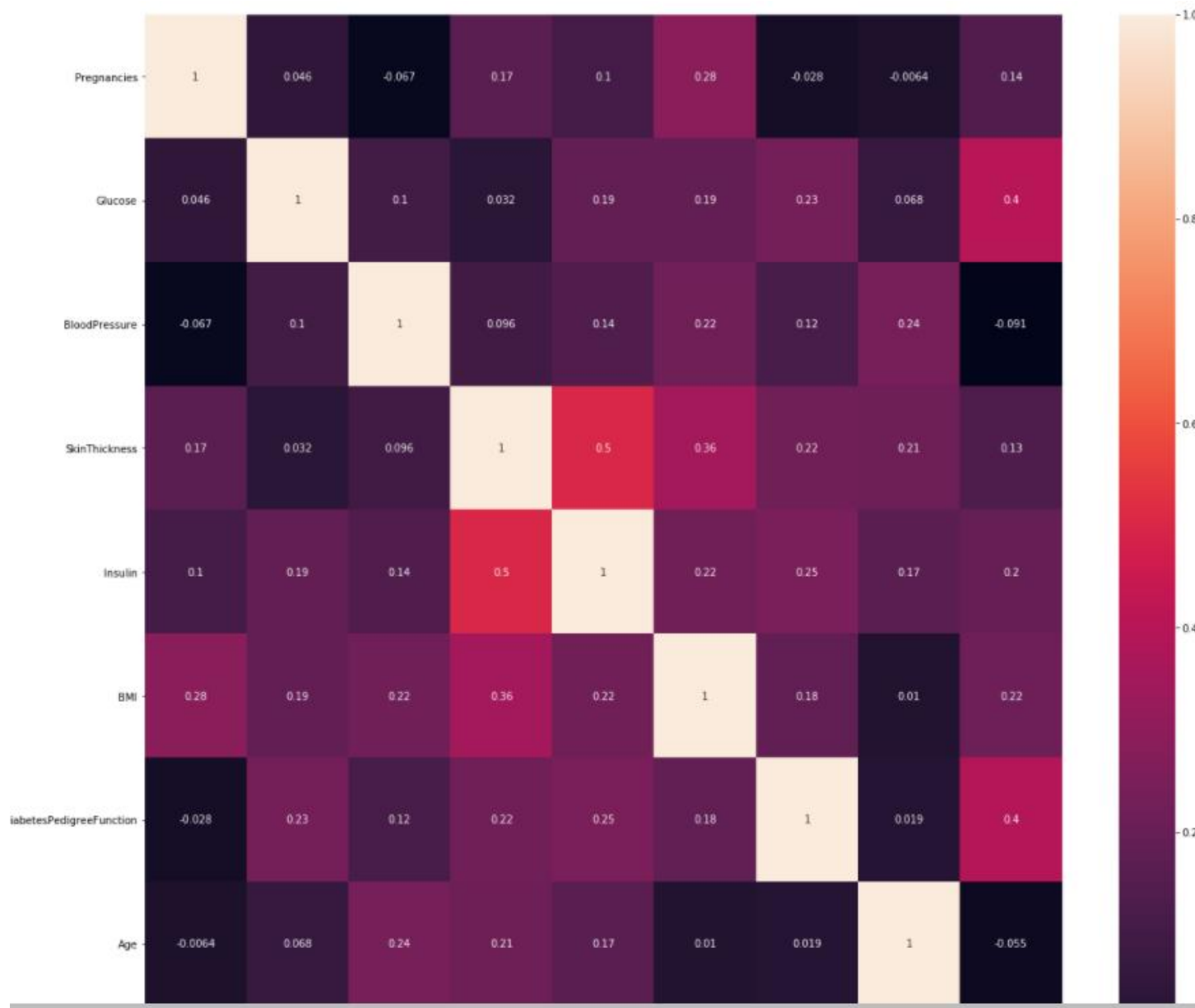
In this project, the outcome of the dataset provided has predicted with four algorithms. The main aim of this project was to design and implement Diabetes Prediction Using Machine Learning Methods and Performance Analysis of that methods and it has been achieved successfully

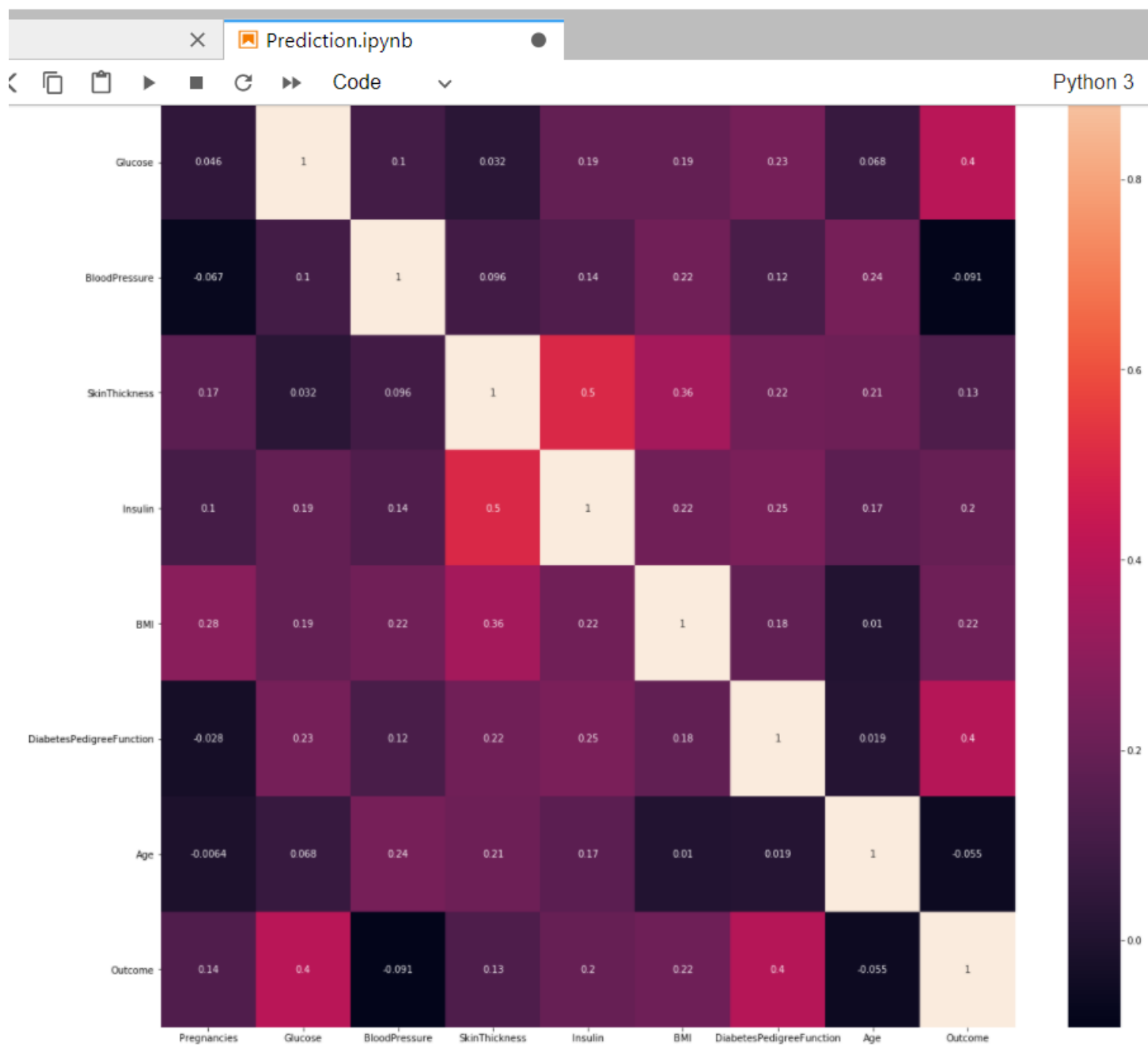
CHAPTER 5

RESULTS AND DISCUSSION

RESULTS:

As a result we have shown the following relation between the factors like BMI, Insulin etc





Prediction:

	model	best_parameters	score
0	logistic_regression	{'C': 1}	0.696
1	decision_tree	{'criterion': 'gini', 'max_depth': 10}	0.816
2	random_forest	{'n_estimators': 50}	0.848
3	svm	{'C': 1, 'kernel': 'rbf'}	0.848

```
: model=RandomForestClassifier(criterion='gini',n_estimators=100)
model.fit(X_train,y_train)
```

```
: RandomForestClassifier()
```

```
: y_pred=model.predict(X_test)
```

```
: from sklearn.metrics import accuracy_score
acc=accuracy_score(y_pred,y_test)
print(acc)
```

```
0.9146341463414634
```

Future Work:

In future this all can be implemented in web interface for our ease and clear cut understanding and we can use non diabetic patients dataset for testing to get result whether they can get diabetes or not in future.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 CONCLUSION:

Models	Accuracy
Logistic Regression	69%
Decision tree	81%
Support Vector Machine(SVM)	84%
Random Forest Classifier	84%
Random Forest Classifier (Tuned)	91%

This project is deployed in Anaconda navigator platform in jupyter notebook

APPENDIX

A) SOURCE CODE:

```
import pandas as pd

df=pd.read_csv("diabetes.csv")

df

df.info()

df.describe()

Q1 = df.Pregnancies.quantile(0.25)

Q3 = df.Pregnancies.quantile(0.75)

Q1, Q3

IQR = Q3 - Q1

IQR

lower_limit = Q1 - 1.5*IQR

upper_limit = Q3 + 1.5*IQR

lower_limit, upper_limit

df=df[(df.Pregnancies<lower_limit)|(df.Pregnancies>(upper_limit-5))]

df

import seaborn as sns

import matplotlib.pyplot as plt

plt.figure(figsize=(20,20))
```

```

#plot heat map

g=sns.heatmap(df.corr(),annot=True)

plt.show()

df.drop(columns=['Insulin','DiabetesPedigreeFunction'],axis=1,inplace=True)

df

X=df.iloc[:, :-1]

y=df.iloc[:, -1]

y

from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.40,random_state=0,shuffle=True)

from sklearn.model_selection import GridSearchCV

from sklearn.model_selection import ShuffleSplit

from sklearn.linear_model import LogisticRegression

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.svm import SVC

def find_best_model(X, y):

    models = {

        'logistic_regression': {

            'model': LogisticRegression(solver='lbfgs', multi_class='auto'),

            'parameters': {

                'C': [1,5,10]

            }

        }

```

```
},
```

```
'decision_tree': {
```

```
    'model': DecisionTreeClassifier(splitter='best'),
```

```
    'parameters': {
```

```
        'criterion': ['gini', 'entropy'],
```

```
        'max_depth': [5,10]
```

```
    }
```

```
},
```

```
'random_forest': {
```

```
    'model': RandomForestClassifier(criterion='gini'),
```

```
    'parameters': {
```

```
        'n_estimators': [10,15,20,50,100,200]
```

```
    }
```

```
},
```

```
'svm': {
```

```
    'model': SVC(gamma='auto'),
```

```
    'parameters': {
```

```
        'C': [1,10,20],
```

```
        'kernel': ['rbf','linear']
```

```
    }
```

```
}
```

```
}
```

```
scores = []
```

```
cv_shuffle = ShuffleSplit(n_splits=5, test_size=0.20, random_state=0)
```

```
for model_name, model_params in models.items():
```

```
    gs = GridSearchCV(model_params['model'], model_params['parameters'], cv =  
cv_shuffle, return_train_score=False)
```

```
    gs.fit(X, y)
```

```
    scores.append({
```

```
        'model': model_name,
```

```
        'best_parameters': gs.best_params_,
```

```
        'score': gs.best_score_
```

```
    })
```

```
return pd.DataFrame(scores, columns=['model','best_parameters','score'])
```

```
find_best_model(X_train, y_train)
```

```
model=RandomForestClassifier(criterion='gini',n_estimators=100)
```

```
model.fit(X_train,y_train)
```

```
y_pred=model.predict(X_test)
```

```
from sklearn.metrics import accuracy_score
```

```
acc=accuracy_score(y_pred,y_test)
print(acc)
model.predict([[2, 81, 72, 15, 30.1, 25]])
```

References:

- [1] Gauri D. Kalyankar, Shivananda R. Poojara and Nagaraj V. Dharwadkar," Predictive Analysis of Diabetic Patient Data Using Machine Learning and Hadoop", International Conference On I-SMAC,978-1-5090-3243-3,2017.
- [2] Ayush Anand and Divya Shakti," Prediction of Diabetes Based on Personal Lifestyle Indicators", 1st International Conference on Next Generation Computing Technologies, 978-1-4673-6809-4, September 2015.
- [3] B. Nithya and Dr. V. Ilango," Predictive Analytics in Health Care Using Machine Learning Tools and Techniques", International Conference on Intelligent Computing and Control Systems, 978-1-5386-2745-7,2017.
- [4] Dr Saravana kumar N M, Eswari T, Sampath P and Lavanya S," Predictive Methodology for Diabetic Data Analysis in Big Data", 2nd International Symposium on Big Data and Cloud Computing,2015