

“PREDICTING DIABETES WITH MACHINE LEARNING”

S. Lavan Karthik 2011CS010266

M. Vamshi Reddy 2011CS010265

P. Jayanth 2011CS010216

S. Hari Sai Ram 2011CS010264

Guided by

Mr. M. Gouthamm, Assistant Professor

**SCHOOL OF ENGINEERING**

**COMPUTER SCIENCE & ENGINEERING**

**MALLA REDDY UNIVERSITY**

(As per Telangana State Private Universities Act No.13 of 2020 and

G.O.Ms.No.14, Higher Education (UE) Department)

**DEPARTMENT OF**

**COMPUTER SCIENCE & ENGINEERING**

Logo

Description automatically generated

CERTIFICATE

This is to certify that the Project Work titled, **“PREDICTING DIABETES WITH MACHINE LEARNING”** a bonafide work of **S. Lavan Karthik (2011CS010266), M.VamshiReddy (2011CS010265), P.Jayanth (2011CS010216), HariSairam (2011CS010264)** who carried out the work under my supervision and submitted in partial fulfillment of the requirements for the award of the degree of BACHELOR TECHNOLOGY in COMPUTER SCIENCE and ENGINEERING by **Malla Reddy University** during the academic year 2022- 2023.

**Guide Name: Signature of Guide:**

**Internal Examiner Name: Signature Internal Examiner:**

**External Examiner Name: Signature External Examiner:**

**HOD Signature**

# DECLARATION

We hereby inform that this main project entitled “**PREDICTING DIABETES WITH MACHINE LEARNING”** has been carried out and submitted in partial fulfillment for the award to the Degree of **Bachelor of Technology in Computer Science and Engineering** to **MALLA REDDY UNIVERSITY, Hyderabad** under the guidance of Mr. M. Gouthamm (M Tech), Assistant Professor, Department of COMPUTER SCIENCEAND ENGINEERING. The work embodied in this project work is original and has not been submitted in part or full for any degree of this or any degree of any other university.

Submitted By

S. Lavan Karthik (2011CS010266)

M. Vamshi Reddy (2011CS010265)

P. Jayanth (2011CS010216)

S. Hari Sairam (2011CS010264)

**INDEX**

|  |  |
| --- | --- |
| **CONTENTS** | **PAGE NO** |
| **ABSTRACT** | **5** |
| **1. INTRODUCTION** | **6-12** |
| 1.1 Introduction of the Project | 6 |
| 1.1.1 Objective | 7 |
| 1.1.2 Methodology | 9 |
| 1.1.3 Algorithm | 9 |
| 1.2 Existing System | 10 |
| 1.3 Issues in Existing Systems | 10 |
| 1.4 Proposed System | 11 |
| 1.4.1 Advantages of Proposed System | 11 |
| 1.5 Dataset Description | 11 |
| 1.6 Dataset Contains | 12 |
| **2. ANALYSIS** | 12-24 |
| 2.1 Requirements Analysis | **12** |
| 2.1.1 Functional Requirements Analysis | 13 |
| 2.1.1.1 Graphs of the model or Confusion Matrix | 14 |
| 2.1.1.2 Histogram | 13 |
| 2.1.1.3 Box Plot | 14 |
| 2.1.2 User and System Requirements | 15 |
| 2.1.3 Non-Functional Requirements | 15 |
| 2.2 Modules Description | 16 |
| 2.2.1 Flow of Algorithm | 17 |
| 2.3 Feasibility Study | 18 |
| 2.3.1 Technical Feasibility | 18 |
| 2.3.2 Operational Feasibility | 19 |
| 2.3.3 Behavioral Feasibility | 20 |
| 2.4 Process Model used | 22 |
| 2.5 Diagrams | 22 |
| 2.6 Hardware and Software Requirements | 24 |
| **3. Design** | 25-30 |
| 3.1 Introduction to Design | 25 |
| 3.2 DFD/UML diagram | 26 |
| 3.2.1 Sequence diagram | 27 |
| 3.2.2 Use case diagram | 28 |
| 3.2.3 Activity diagram | 30 |
| **4. Implementation and Coding** | 31-39 |
| 4.1 Add Algorithm theory and algorithm | 31 |
| 4.2 Coding | 32-38 |
| **5. Output (Screen shot)** | 39,40 |
| **6. Conclusion and future Enhancement** | 41 |
| **7. Reference** | 42 |

**LIST OF FIGURES**

|  |  |
| --- | --- |
| **FIGURE NAMES** | **Page No** |
| Figure 1: methodology | 9 |
| Figure 1.2: flow of algorithm | 18 |
| Figure 1.3: n\_neighbors | 20 |
| Figure 1.4: decision tree features | 21 |
| Figure 1.5: randomforest | 22 |
| Figure 1.6: svm accuracy | 22 |
| Figure 1.7: process model | 23 |
| Figure 2: implementation model | 24 |
| Figure 3: uml diagrams | 26 |
| Figure 3.1: sequence diagram | 27 |
| Figure 3.2: use case diagram | 28 |
| Figure 3.3: activity diagram | 29 |

**LIST OF TABLES**

|  |  |
| --- | --- |
| **TABLES NAMES** | **Page No** |
| Table 1: dataset description | 12 |
| Table 2: dataset contains | 13 |

**LIST OF OUTPUT SCREENS**

|  |  |
| --- | --- |
| **OUTPUT SCREENS** | Page No |
| Figure 4.1: implementing of code libraries | 30 |
| Figure 4.2: EDA | 30 |
| Figure 4.3: modifying the code | 31 |
| Figure 4.4: retrieveing of data | 31 |
| Figure 4.5: histogram | 32 |
| Figure 4.6: box plot | 32 |
| Figure 4.7: dist plot | 33 |
| Figure 4.8: heatmap | 33 |
| Figure 4.9: declaring of x and y | 34 |
| Figure 4.10: importing of randomforest classifier | 34 |
| Figure 4.11: importing pickle file | 35 |
| Figure 4.12: prediction | 35 |
| Figure 5: python code | 36 |
| Figure 6: AWS environment | 37 |
| Figure 7: EC2 instance dashboard | 38 |
| Figure 8: frontend page | 39 |
| Figure 9: outcome (you have diabetes) | 39 |
| Figure 10: outcome (you don’t have diabetes) | 39 |

# ABSTRACT

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 machine learning, machine learning has been applied to many aspects of medical health. In this study, we used decision tree, random forest to predict diabetes mellitus. The results showed that prediction with random forest could reach the highest accuracy when all the attributes were used. Type 2 diabetes occurs more commonly in middle-aged and elderly people, which is often associated with the occurrence of obesity, hypertension, dyslipidemia, arteriosclerosis, and other diseases. Machine learning methods are widely used in predicting diabetes, and they get preferable results. Decision tree is one of popular machine learning methods in medical field, which has grateful classification power.

Random forest generates many decision trees. So in this study, we used decision tree, random forest (RF) to predict the diabetes. Machine learning (ML) algorithms methods are used in diabetes prediction in our research. Predictive learning is a technique of machine learning in which an agent tries to build a model of its environment by trying out different actions in various circumstances. It uses knowledge of the effects its actions appear to have, turning them into planning operators. At its most basic, machine learning uses programmed algorithms that receive and analyze input data to predict output values within an acceptable range. As new data is fed to these algorithms, they learn and optimize their operations to improve performance, developing 'intelligence’ over time.

# 1.INTRODUCTION

# 1.1 INTRODUCTION OF THE PROJECT

First before we learn about diabetes predictor, let us clearly know what diabetes actually is, Diabetes, also known as diabetes mellitus, is a group of metabolic disorders characterized by a high blood sugar level over a prolonged period of time. Symptoms often include frequent urination, increased thirst and increased appetite If left untreated, diabetes can cause many health complications. Acute complications can include diabetic ketoacidosis, hyperosmolar hyperglycaemic state, or death. Serious long-term complications include cardiovascular disease, stroke, chronic kidney disease, foot ulcers, damage to the nerves, damage to the eyes, and cognitive impairment. Diabetes is due to either the pancreas not producing enough insulin, or the cells of the body not responding properly to the insulin produced. Insulin is a hormone which is responsible for helping glucose from food get into cells to be used for energy. There are three main types of diabetes mellitus. Apart from this there are mainly two types of Diabetes and let us understand them in detail.

1. [Type 1 diabetes](https://en.wikipedia.org/wiki/Type_1_diabetes) results from failure of the pancreas to produce enough insulin due to loss of [beta cells](https://en.wikipedia.org/wiki/Beta_cell). his form was previously referred to as "insulin-dependent diabetes mellitus" or "juvenile diabetes" he loss of beta cells is caused by an [autoimmune](https://en.wikipedia.org/wiki/Autoimmunity) response. The cause of this autoimmune response is unknown. Although Type 1 diabetes usually appears during childhood or adolescence, it can also develop in adults. [Type 2 diabetes](https://en.wikipedia.org/wiki/Type_2_diabetes) begins with [insulin resistance](https://en.wikipedia.org/wiki/Insulin_resistance), a condition in which cells fail to respond to insulin properly. As the disease progresses, a lack of insulin may also develop. This form was previously referred to as "non-insulin-dependent diabetes mellitus" or "adult-onset diabetes".
2. Type 2 diabetes is more common in older adults, but a significant increase in the prevalence of obesity among children has led to more cases of type 2 diabetes in younger people. The most common cause is a combination of [excessive body weight](https://en.wikipedia.org/wiki/Obesity) and [insufficient exercise](https://en.wikipedia.org/wiki/Sedentary_lifestyle).
3. [Gestational diabetes](https://en.wikipedia.org/wiki/Gestational_diabetes) is the third main form and occurs when [pregnant women](https://en.wikipedia.org/wiki/Pregnancy) without a previous history of diabetes develop high blood sugar levels. In women with gestational diabetes, blood sugar usually returns to normal soon after delivery. However, women who had gestational diabetes during pregnancy have a higher risk of developing type 2 diabetes later in life.
4. Coming to our predictor it is mainly used to predict the diabetes in a person based on all the results that are obtained by various tests performed during, however our predictor does not provide any treatment or medication about the disease and it just conforms whether the person has the disease or not, this prediction is very helpful up to a great extent in this present corporate world where everything is related to money mainly.

**1.1.1 OBJECTIVE**

This research work aims to analyse the Diabetes dataset, design, and implement a Diabetes prediction and

recommendation system utilizing machine learning classification techniques. The specific objectives of this project work are:

1. To review existing literature along the area of diabetes diagnosis and prediction.
2. Design and develop a model using machine learning techniques.
3. To analyse the Diabetes dataset and use Support Vector Machine and Random forest algorithms to develop a prediction engine.
4. To identify and discuss the benefits of the designed system along with effective applications.

An effective framework for knowledge required Machine learning has been used successfully by researchers for the prognosis and/or diagnosis of diabetes for active and accurate decision making. Therefore, this project focuses on the use of machine learning techniques on a set of data collected which is an online dataset to uncover hidden patterns and predict diabetes based on the dataset collected. Support Vector Machine and random forest are proposed for use in the prediction of diabetes in a patient to ensure that the information gotten from the system built based on these techniques are reliable. Good demonstration of a real-world machine learning process. As a clinician I have concerns about using this dataset without some medical expertise. For example, you can't have triceps thickness or insulin levels of zero. This means the test was not done. Is imputation legitimate in this situation? A pregnancy level of zero could mean it was not asked or it could mean no pregnancies. We don't know which. The column on pedigree is complicated and probably should be deleted and not used.

Predicting Diabetes Mellitus in Patients Using Machine Learning. Diabetes Mellitus, the disease caused by hyperglycemia and can be classified as type 1 and type 2.

1. Type 1 diabetes is often discovered in early diagnosis in young children and adolescents. However, given recent evidence, it is shown that the condition can develop at any age and therefore people of all ages are susceptible to development.
2. Type 2 diabetes is an autoimmune disease that causes the insulin producing beta cells in the pancreas to be eradicated thereby causing prevention for the body to produce adequate levels of insulin to regulate blood glucose levels.

Another terminology for type 1 diabetes is Insulin Dependent Diabetes. On the contrary, type 2 diabetes, which is the most common diagnosis in the general population, is a metabolic disorder resulting in hyperglycemia. This is often the prognosis of the body’s inability to utilize the insulin it produces or ability to produce insulin. Gestational diabetes appears in pregnant women who suddenly develop high blood sugar. In two thirds of the cases, it will reappear during subsequent pregnancies. There is a great chance that type 1 or type 2 diabetes will occur after a pregnancy affected by gestational diabetes.

**Symptoms of Diabetes**

1. Frequent Urination
2. Increased thirst
3. Tired/Sleepiness
4. Weight loss
5. Blurred vision
6. Mood swings
7. Confusion and difficulty concentrating
8. frequent infection

**1.1.2 METHODOLOGY**

In this section we shall learn about the various classifiers used in machine learning to predict diabetes. We shall also explain our proposed methodology to improve the accuracy. Five different methods were used in this paper. The different methods used are defined below. The output is the accuracy metrics of the machine learning models. Then, the model can be used in prediction.

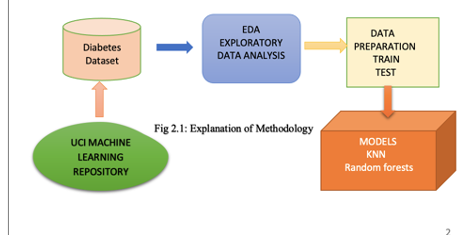


Fig 1

**1.1.3 ALGORITHM**

STEP1: Gathering the data from the Kaggle and understand about the dataset and preparing the data by removing the missing values.

STEP 2: Now we split the gathered data into training and testing dataset.

STEP 3: Now using training data we create a Random Forest Classification model.

STEP 4: Using the testing data we test the created Random Forest Classification model and import the pickle file.

STEP 5: Using the model now we predict diabetes.

**1.2 EXISTING SYSTEM**

Existing many research handled for the diabetes detection. Data mining approach like clustering, classification was studied in existing system. Diabetes prediction using algorithm such as K-NN, K-Means, branch and bound algorithm was proposed. A basic diabetic dataset is chosen for carrying out the comparative analysis. The importance of feature analysis for predicting diabetes by employing machine learning technique is discussed.

**1.3 ISSUES IN EXISTING SYSTEM**

High false positives. There is no interactive tool for users to predict diabetes.

1. Costing: The Existing system is high cost and this is main reason most of the system is failed.
2. Technology Complexity: Most of system is complex to understand, not user friendly as compare to our proposed system.
3. Time Consuming Feature: In existing system, the performance is low and most of the time system gets hanged due to load.
4. Not Easy to Understand: Systems re complex to understand and they were not user friendly.
   1. **PROPOSED SYSTEM**
5. The Proposed System study is classification of Indian PIMA dataset for diabetes as a binary classification problem.
6. This is proposed to achieve through machine learning classification algorithms and Regression.
7. For machine learning SVM algorithm, K-NN algorithm, Random Forest algorithm and Logistic Regression are proposed.
8. The proposed system improves accuracy of prediction through Random Forest Classification.
9. The proposed is also has the frontend with the angular code. The website is fully dynamic and the we can user friendly website.
10. Then the project has deployed in the AWS EC2 backend is deployed and the front is deploy in the s3 bucket.

**1.4.1 ADVANTAGES OF PROPOSED SYSTEM**

1. Interactive application, in which user can give a single input to arrive the prediction.
2. Accuracy is improved using machine learning techniques.
3. By this project we can know about the machine learning algorithms and AWS basics.

**1.5 DATASET DESCRIPTION**

The diabetes data set was originated from <https://www.kaggle.com/johndasilva/diabetes>. Diabetes dataset containing 2000 cases. The objective is to predict based on the measures to predict if the patient is diabetic or not.

Table

Description automatically generated

Table 1

1. The diabetes data set consists of 2000 data points, with 9 features each.
2. “Outcome” is the feature we are going to predict, 0 means No diabetes, 1 means diabetes.

Table

Description automatically generated

There are no null values in dataset.

**1.6 DATASET CONTAINS**

|  |  |  |
| --- | --- | --- |
| Serial no | Attributes names | description |
| 1 | Pregnancies | Number of times of pregnant. |
| 2 | Glucose | Plasma glucose concentration. |
| 3 | Blood Pressure | Diastolic blood pressure. |
| 4 | Skin Thickness | Triceps skin fold thickness(mm). |
| 5 | Insulin | 2-h serum insulin. |
| 6 | BMI | Body mass index. |
| 7 | Diabetes pedigree function | Diabetes pedigree function |
| 8 | Age | Age of patient. |
| 9 | Outcome | Class variable(0 or 1). |

# Table 2

# 2.ANALYSIS

# 2.1 REQUIREMENTS ANALYSIS

The requirement analysis of the entire tool, features, software, hardware, and logistics were untaken at this step. The major feature of the system dells on the benefit of automatizing the process of the information given to the patients for their prevention. This will ensure not the information also can know about their body status and the fitness. The reliability is more important in this analysis.

**2.1.1 FUNCTIONAL REQUIREMENT ANALYSIS**

The programming level is used in the website is only purely depend on the frontend if we want to add the database, we will use the backend but for now the website no need of the backend for giving the information we need only the frontend and look and feel and information we give to the user is more enough for the cultivation of the crop form the cultivation of land to yielding of the crop. The website contains the details of the patient and as well as the person can get the information the health organization. It is easy to see that there is no single feature that has a very high correlation with our outcome value. Some features have a negative correlation with the outcome value and some have positive.

**2.1.1.1 Graphs of the model**

**Confusion matrix**

**Chart

Description automatically generated**

Graph 1

**Chart, box and whisker chart

Description automatically generated2.1.1.2 Histogram**

Graph 2

Let’s take a look at the plots. It shows how each feature and label is distributed along different ranges, which further confirms the need for scaling. Next, wherever you see discrete bars, it basically, means that each of these is actually a categorical variable. We will need to handle these categorical variables before applying Machine Learning. Our outcome labels have two classes, 0 for no disease and 1 for disease.

**2.1.1.3 Box plot**

**Chart, bar chart

Description automatically generated**

Graph 3

The above graph shows that the data is biased towards datapoints having outcome value as 0 where it

means that diabetes was not present actually. The number of non-diabetics is almost twice the number

of diabetic patients.

# 2.1.2 USER AND SYSTEM REQUIREMENTS

User requirements are the only login with the name and email or phone number and go through the and collect the information from the website. User requirements also include the user suggestions will be taken and they can encourage the other users also by giving the information about their crop yielding. System requirements are minimal which is a basic computer/ phone with access to the internet on any operating system

# NON-FUNCTIONAL REQUIREMENTS

1. Speed: The speedy responsive website is essential for customer satisfaction and to retain the customer on the platform.
2. Portability: The application must be portable and must not require a lot of resources from the users’ devices.
3. Compatibility: The web application is compatible with all the major operating systems with an internet connection and with any modern browser.
4. Information: The user is satisfactory with the given information or not is the main thing in the website.
5. Security: The given data of the user is secure are not no one can access the data of the user.

# 2.2 MODULES DESCRIPTION

We have only the single page application to know that the diabetes is present or not for the patients.

**USER**

The User can register the first. While registering he required a valid User email and mobile for further communications. Once the User registers, then the admin can activate the User. Once the admin activates the User then the User can login into our system. After login User will add the data to predict diabetes prediction is present or not.

**ADMIN**

Admin can login with his credentials. Once he logs in he can activate the users. The activated users only login in our applications. The admin will store csv data into our database. we can implement Random Forest classification algorithm to predict diabetes and also we can perform cross validation.

**MACHINE LEARNING**

Machine learning refers to the computer’s acquisition of a kind of ability to make predictive judgments and make the best decisions by analyzing and learning a large number of existing data. The representation algorithms include deep learning, artificial neural networks, decision trees, enhancement algorithms and so on. The key way for computers to acquire artificial intelligence is machine learning. Nowadays, machine learning plays an important role in various fields of artificial intelligence. Whether in aspects of internet search, biometric identification, auto driving, Mars robot, or in American presidential election, military decision assistants and so on, basically, as long as there is a need for data analysis, machine learning can be used to play a role.

# 2.2.1 FLOW OF ALGORITHM:

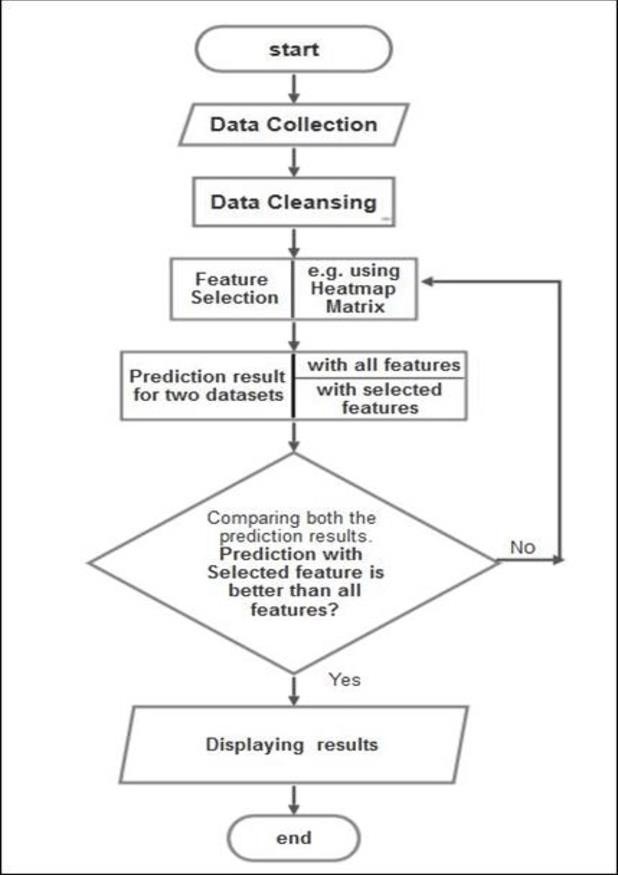


Fig 1.2

## 2.3 FEASIBILITY STUDY

A feasibility study is an assessment of the practicality of a proposed plan or project. A feasibility study analyzes the viability of a project to determine whether the project or venture is likely to succeed.

# TECHNICAL FEASIBILITY

Technical feasibility inspects whether software can be built at all with available tools and experts. The proposed system will be built using the following technologies:

1. HTML, CSS, Angular for front-end framework.
2. Machine learning code and FastAPI for the back-end runtime environment.
3. AWS for deploying of the frontend in S3 bucket and backend code in EC2 instance.

The prototype is largely built on the Machine Learning and allows for dynamic and easy updates of changes in the prototype.

# 2.3.2 OPERATIONAL FEASIBILITY

Operational feasibility is the measure of how well a proposed system solves the problems and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. Operational Feasibility reviews the willingness of the organization to support the proposed system. This is probability the most difficult of the feasibilities to gauge. In order to determine this feasibility, It is important to understand the management commitment to the proposed project. If the request was initiated by management, it is likely that there is management support and the system will be accepted and used.

# BEHAVIORAL FEASIBILITY

The prototype will evaluate and estimates the user attitude or behavior towards the development of a new system. It helps in determining if the system requires special effort to educate the user or not. The prototype

is proposed with the behavioral feasibility in consideration.

# PROCESS MODEL USED

MACHINE LEARNING MODELS USED IN THE PROJECT

**K-Nearest Neighbors**

The K-NN algorithm is arguably the simplest machine learning algorithm. Building the model consists only of storing the training data set. To make a prediction for a new data point, the algorithm finds the closest data points in the training data set, its “nearest neighbors.”

First, let’s investigate whether we can confirm the connection between model complexity and accuracy:

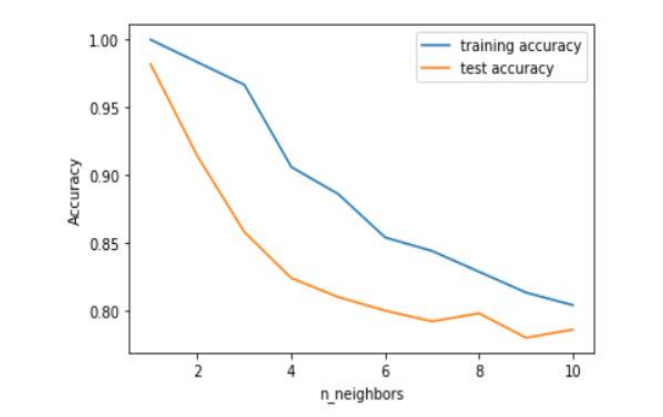
****

Fig1.3

The above plot shows the training and test set accuracy on the y-axis against the setting of n-neighbors on the x-axis. Considering if we choose one single nearest neighbor, the prediction on the training set is perfect. But when more neighbors are considered, the training accuracy drops, indicating that using the single nearest neighbor leads to a model that is too complex. The best performance is somewhere around 9 neighbors.

|  |  |
| --- | --- |
| Training Accuracy | 0.81 |
| Testing Accuracy | 0.78 |

**Decision Tree**

This classifier creates a decision tree based on which, it assigns the class values to each data point. Here, we can vary the maximum number of features to be considered while creating the model.

**Feature Importance in Decision Trees**

Feature importance rates how important each feature is for the decision a tree makes. It is a number between 0 and 1 for each feature, where 0 means “not used at all” and 1 means “perfectly predicts the target”.

|  |  |
| --- | --- |
| Training Accuracy | 0.78 |
| Testing Accuracy | 0.75 |

**Chart, bar chart

Description automatically generated**

Fig 1.4

Feature “Glucose” is by far the most important feature.

**Random Forest Classification**

This classifier takes the concept of decision trees to the next level. It creates a forest of trees where each tree is formed by a random selection of features from the total features.We divide the data into training and testing. Then we use the Random Forest Classifier to classify the patient is having diabetes or not.This classifier takes the concept of decision trees to the next level. It creates a forest of trees where each tree is formed by a random selection of features from the total features.

**Feature importance in Random Forest**

**Chart, bar chart

Description automatically generated**

Fig 1.5

Similarly, to the single decision tree, the random forest also gives a lot of importance to the “Glucose” feature, but it also chooses “BMI” to be the 2nd most informative feature overall.

**Support Vector Machine**

This classifier aims at forming a hyper plane that can separate the classes as much as possible by adjusting the distance between the data points and the hyper plane. There are several kernels based on which the hyper plane is decided. I tried four kernels namely, linear, poly, rbf and sigmoid.

Chart, bar chart

Description automatically generated

Fig 1.6

As can be seen from the plot above, the linear kernel performed the best for this dataset and achieved a score of 77%.

**Accuracy Comparison**

|  |  |  |
| --- | --- | --- |
| Models | Training accuracy | Testing accuracy |
| Decision Tree | 78% | 78% |
| k-Nearest Neighbors | 81% | 78% |
| Random Forest | 94% | 99% |
| SVM | 76% | 77% |

There is more accuracy for the random forest classification, so we use the random forest classification to create the pickle file.

**2.5 DIAGRAMS**

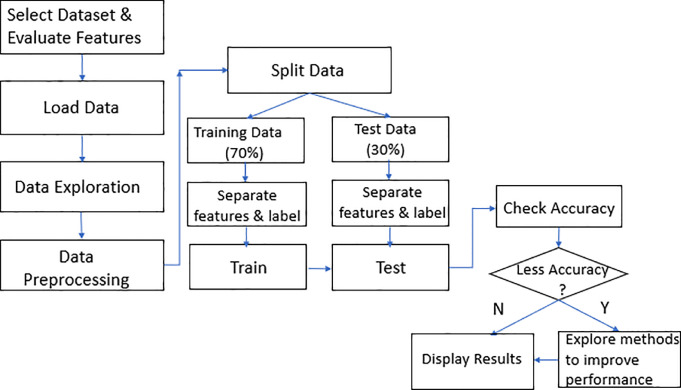
****

Fig 1.7: process model

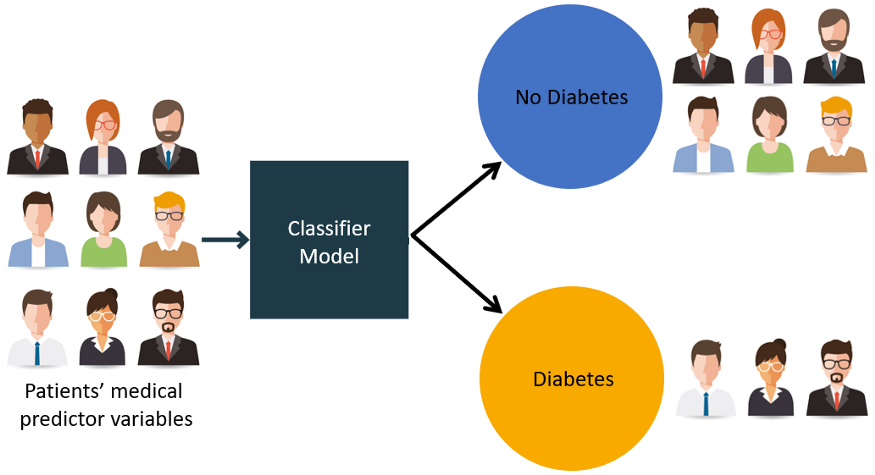


Fig 2

**REQUIREMENTS**

**SOFTWARE REQUIREMENTS**

1. Python(backend)
2. FastAPI(backend)
3. Uvicorn(backend)
4. Angular(frontend)
5. Node Js(frontend)

**HARDWARE REQUIREMENTS**

1. Processor: minimum 1 GHz or more.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi).
3. Hard drive: Minimum 32 GB; Recommended 64 GB.
4. Memory (RAM): Minimum 1 GB; Recommended 4 GB.

# 3.DESIGN

**3.1 INTRODUCTION TO DESIGN**

The design is to be created in the angular environment for the dynamic website we used the angular for the best website and the look and feel. If you want an amazing agriculture website that stands out from the competition, work with professional designers from Fire art. Hire designers to make your vision come to life or host a design contest and get ideas from designers around the world. A great website shows the world who you are, makes people remember you, and helps potential customers understand if they found what they were looking for. Websites communicate all of that through color, shape and other design elements. Learn how to make your agriculture website tell your brand’s story. you are looking for visual design concepts and inspirations to develop the next agriculture website or app, you might be at the right place. Have a look at our range of the most effective agriculture web site design concepts below to discover a website look that works for you. We use the angular for the frontend. To represent the outcome for the machine learning we use the angular application. In the angular we create the service component and we connect the machine learning model to the service by deploying the model in the AWS EC2 instance and that IP address is to be call in the service component in the angular.

After completing the connection, we have to run the angular project. To check the project then after You have run the IP address then the angular project should run then after the machine learning model Should run in the instances also; we can deploy the angular application in the instance as well as in theS3 bucket of the AWS management console. We see in the instances as well as in the S3 bucket also.

Deploying in the instances: IP address should run with the port at which it has running.

# DFD/UML DIAGRAM

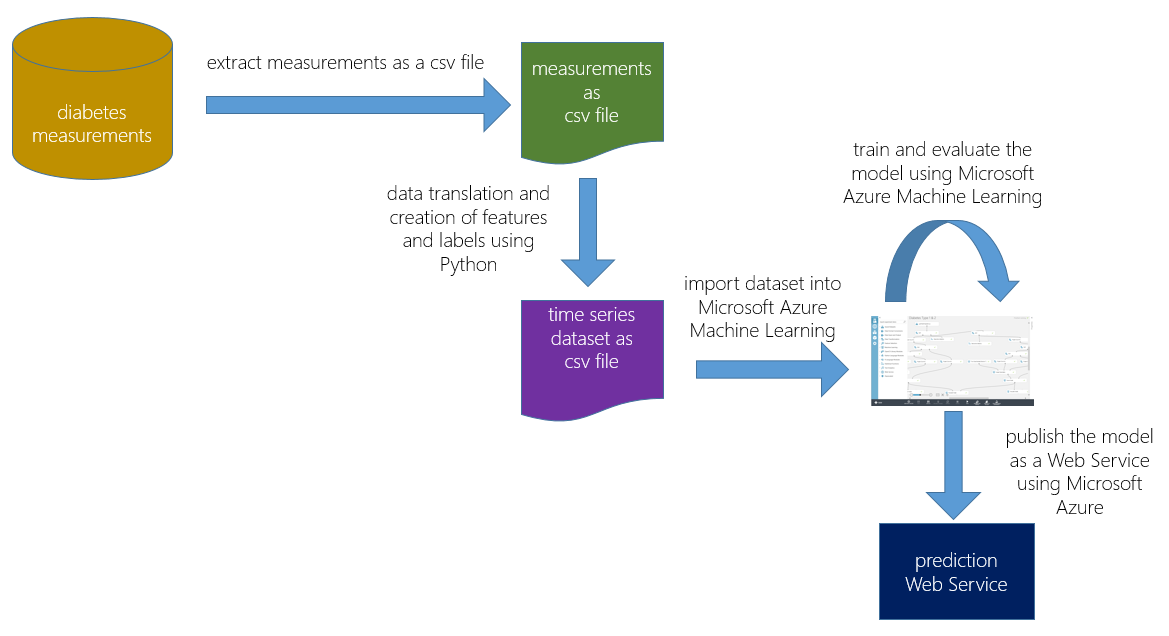
****

Fig 3: uml diagram

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

* + 1. **SEQUENCE DIAGRAM**

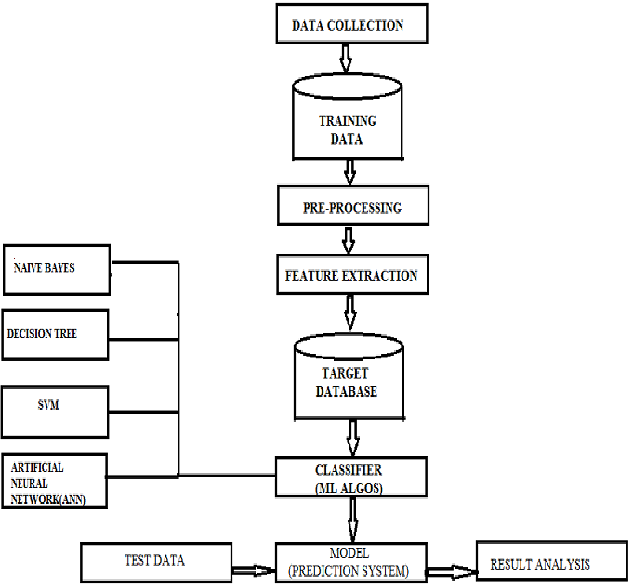
****

Fig 3.1: sequence diagram

Sequence Diagram represents the sequence of steps as follows:

1. User will submit review.
2. The review gets analyzed by the analyzer.
3. After analysis, the result is displayed to the user whether review is positive or negative

# USECASE DIAGRAM

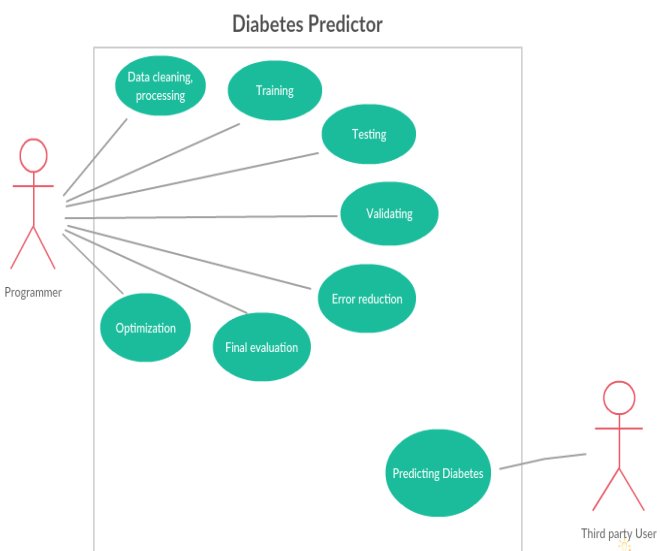
****

Fig 3.2: use case diagram

UML UseCase Diagram Tutorial the UML Class diagram is a graphical notation used to construct and visualize object oriented systems. A class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the systems.

# 3.2.3 ACTIVITY DIAGRAM

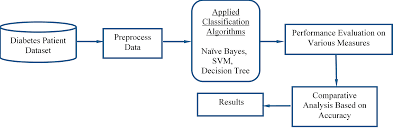
****

Fig 3.3: activity diagram

Activity diagram for online shopping system The activity diagram used to describe flow of activity through a series of actions. Activity diagram is an important diagram to describe the system. The activity described as a action or operation of the system.

# 4.IMPLEMENTATION OF CODING

* 1. **ADD ALGORITHM THEORY AND ALGORITHM**

An Algorithm is a procedure to solve a particular problem in a finite number of steps for a finite- sized input. The algorithms can be classified in various ways. The first thing we'll have to do is agree on what an algorithm is. Simply put, an algorithm is a recipe used to solve a problem. From a human's point of view, an example of an algorithm could be how to get up in the morning. Although it might sound simple, it can get very complicated, very quickly. Computers are machines and machines don't think. Therefore, we have to describe all of the algorithm's steps precisely. With this, we get to the first property of an algorithm - **it must be elementary** (consist of the final number of simple and comprehensible steps, commands). "Get out of bed" certainly isn't an algorithm. "Open your eyes, take the blanket off, sit up, put your legs on the ground and stand up" - this one sound concrete enough to be a true algorithm. However, we're going to keep things relevant to IT, so we'll solve tasks like sorting items by their size or searching elements by their values. Sorting and searching are two basic tasks which computers do all the time, so they have to be thought out and optimized very well. Otherwise, they'd take up too much time that could be used for other processes.

**Random Forest Classification**

This classifier takes the concept of decision trees to the next level. It creates a forest of trees where each

tree is formed by a random selection of features from the total features. We divide the data into training

and testing. Then we use the Random Forest Classifier to classify the patient is having diabetes or not.

This classifier takes the concept of decision trees to the next level. It creates a forest of trees where each

tree is formed by a random selection of features from the total features.

|  |  |
| --- | --- |
| Training Accuracy | 1.00 |
| Testing Accuracy | 0.975 |

**4.2 CODE REQUIRED**

The project is created in the angular for the more dynamic process. The code is given below for

the execution of the project. Create angular project and the place the code in the following methods.

The algorithm used in the model is RANDOM FOREST CLASSIFICATION the Jupiter

notebook is used in the pickle file is created and the application is deploy in the cloud.

The backend code for the machine learning:

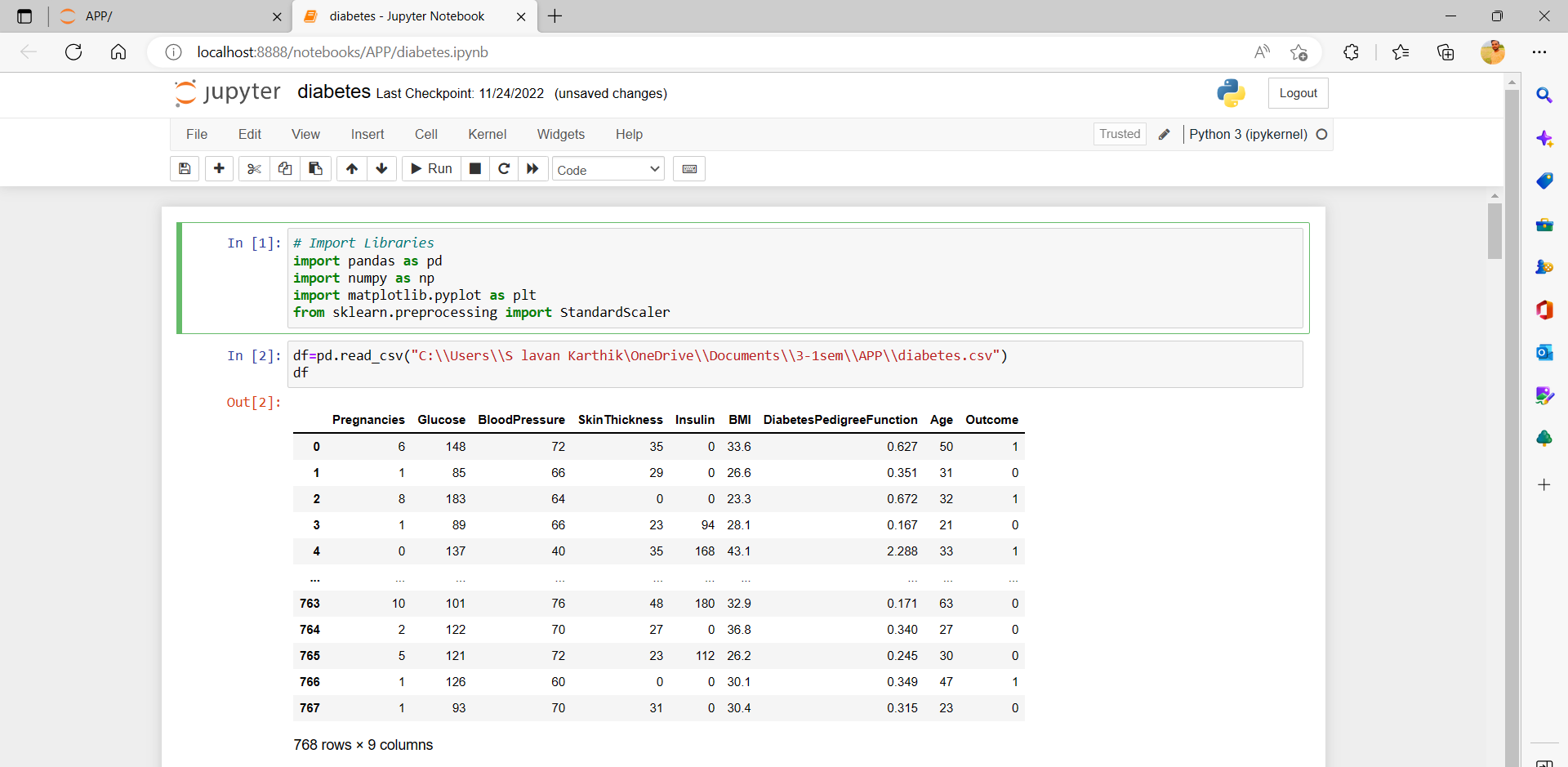
****

Fig 4.1: importing of libraries and loading dataset.

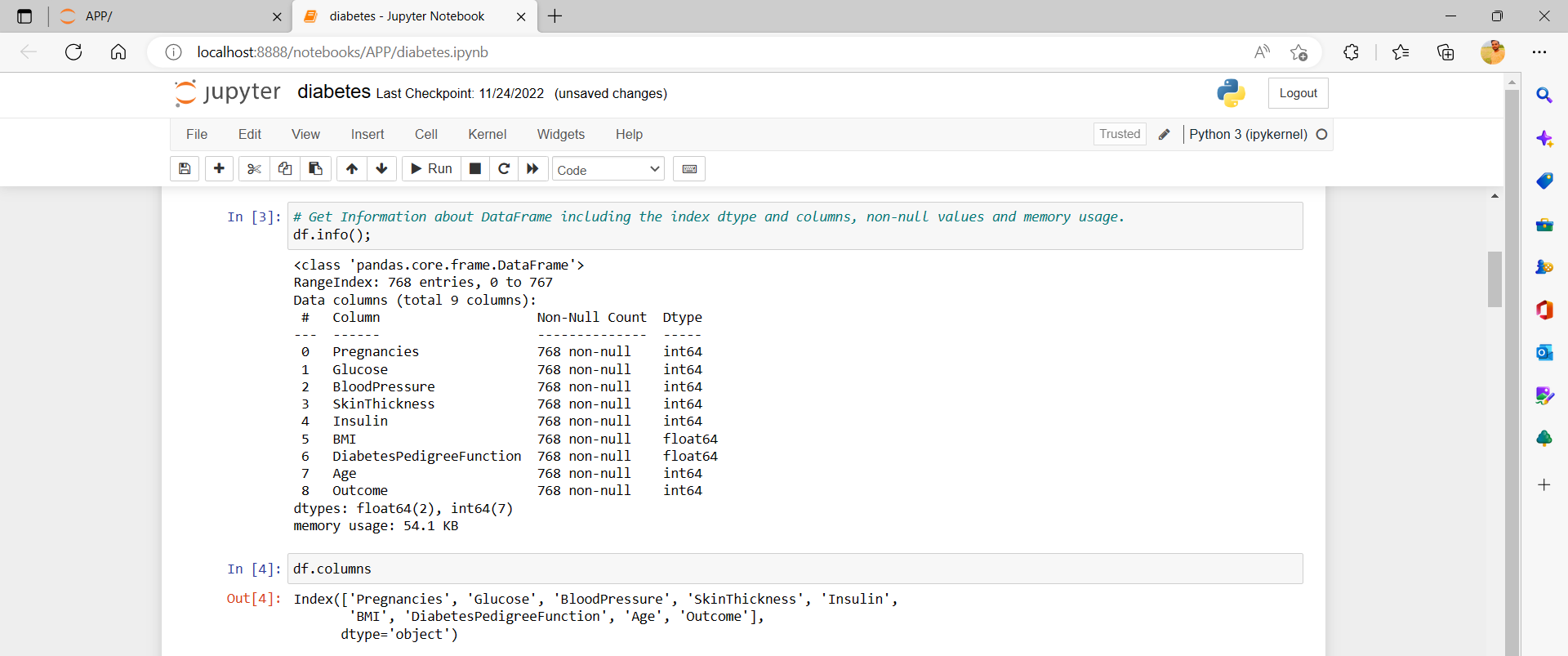
****

Fig 4.2: EDA

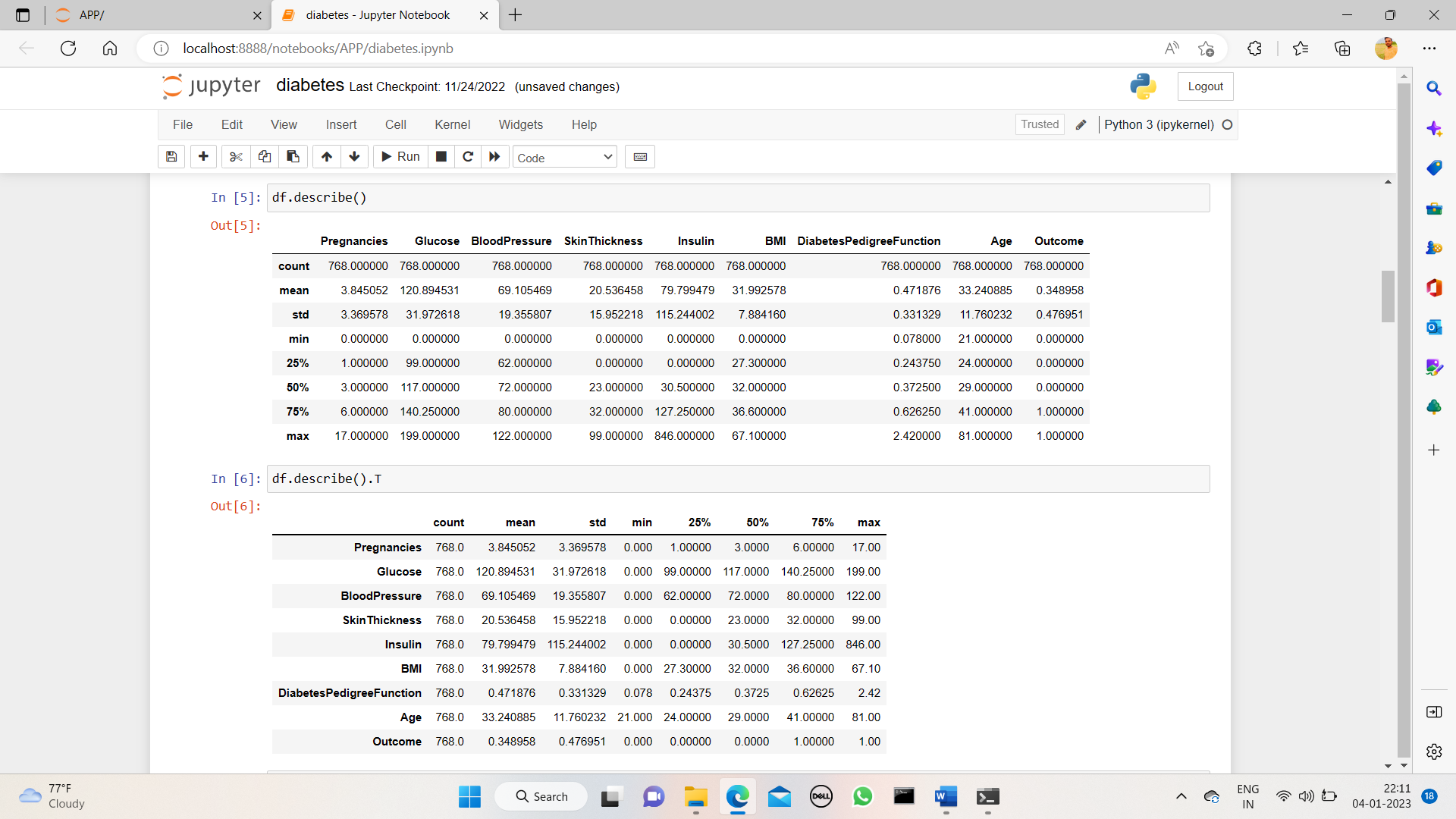


Fig 4.3: modifying of code

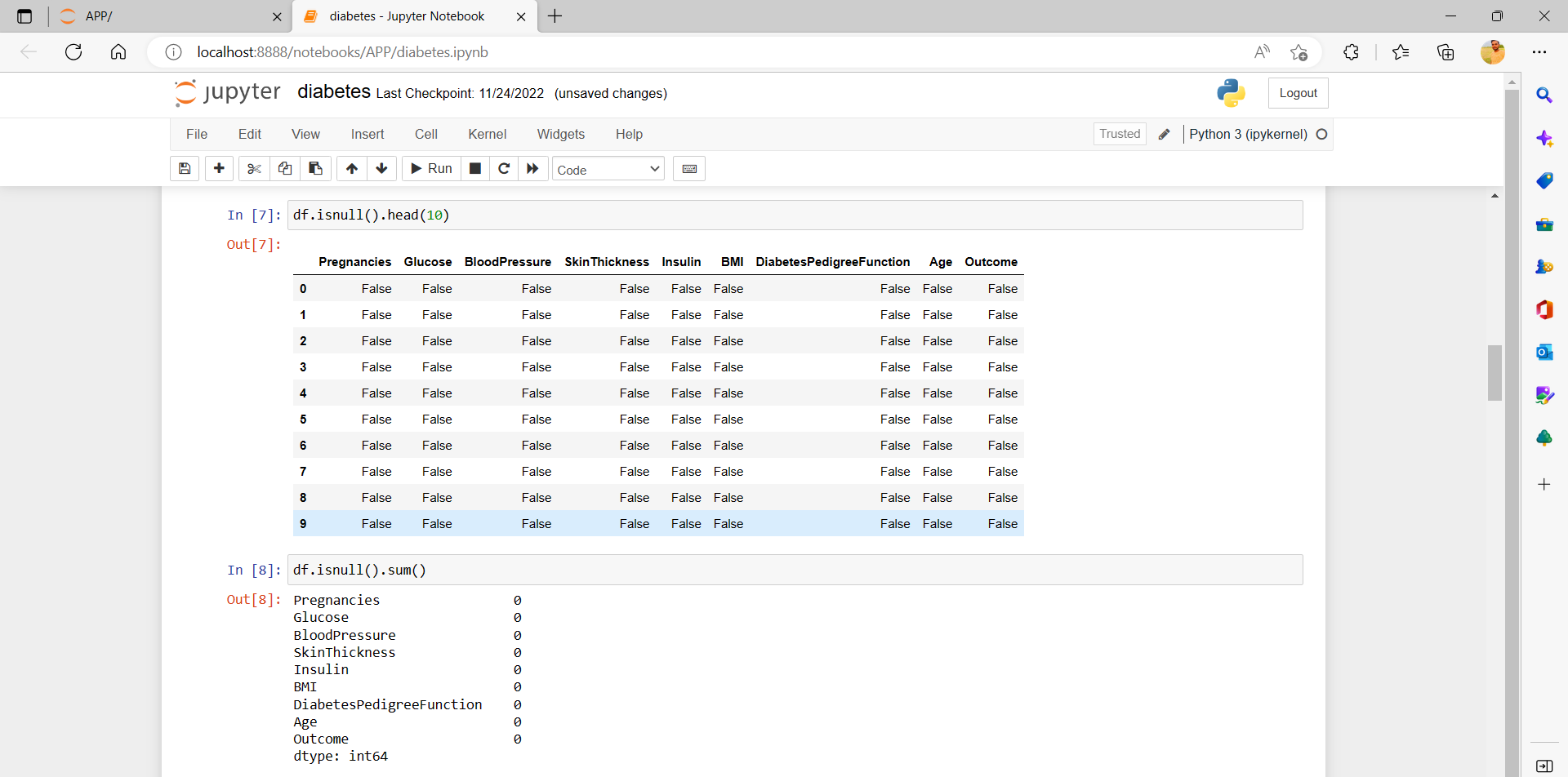


Fig 4.4: retrieveing of data

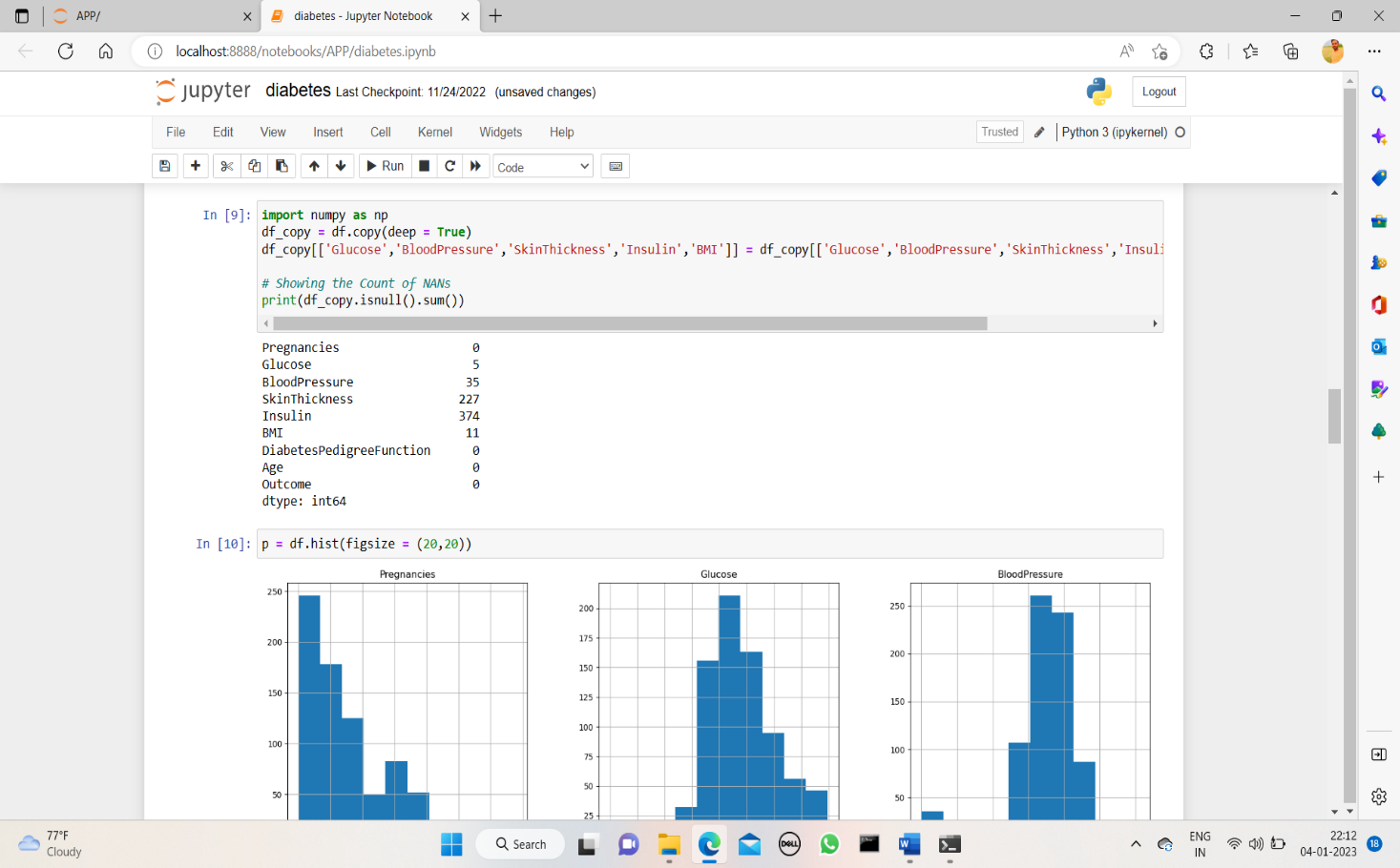


Fig 4.5: histogram

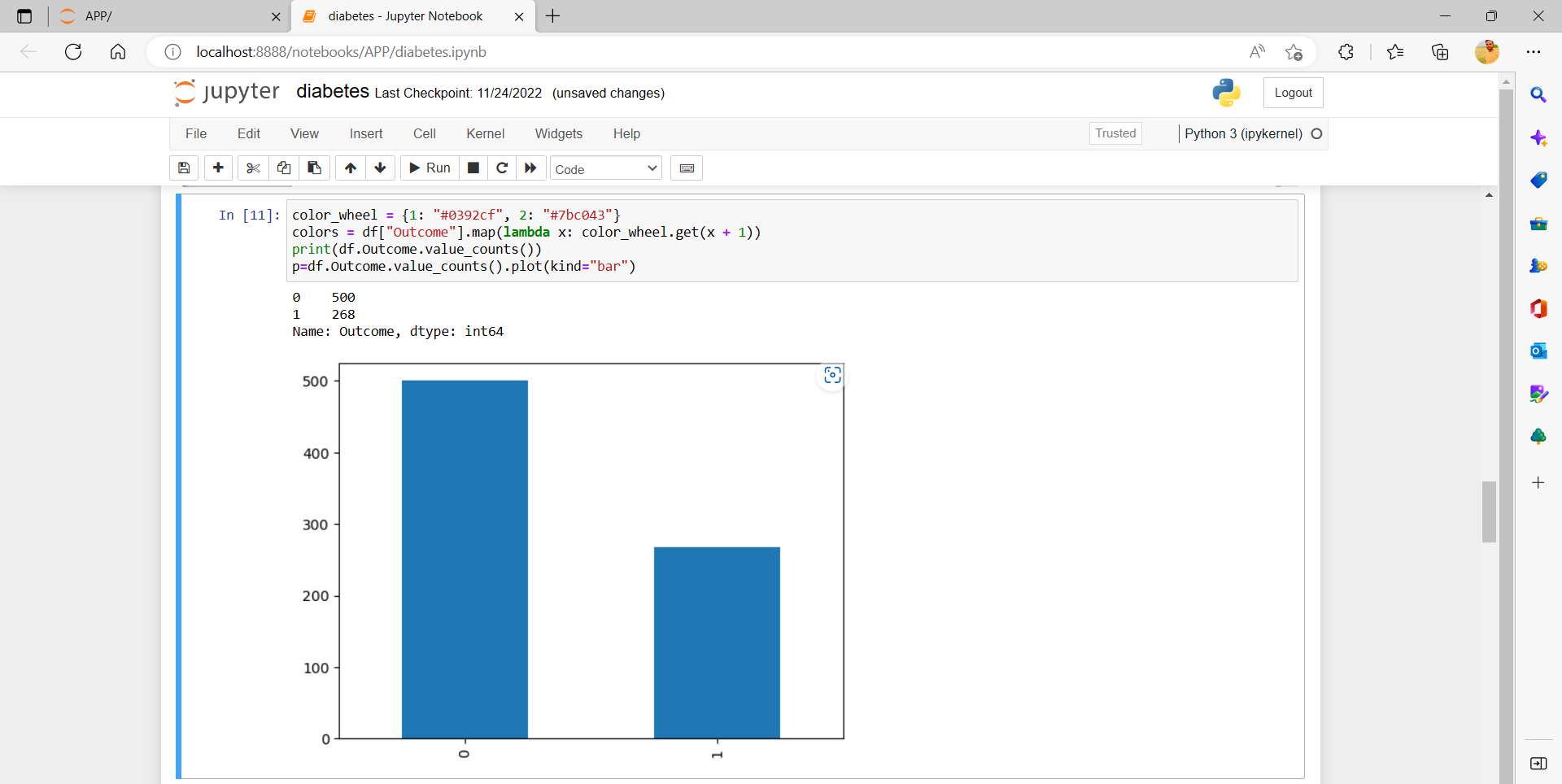


Fig 4.6: box plot

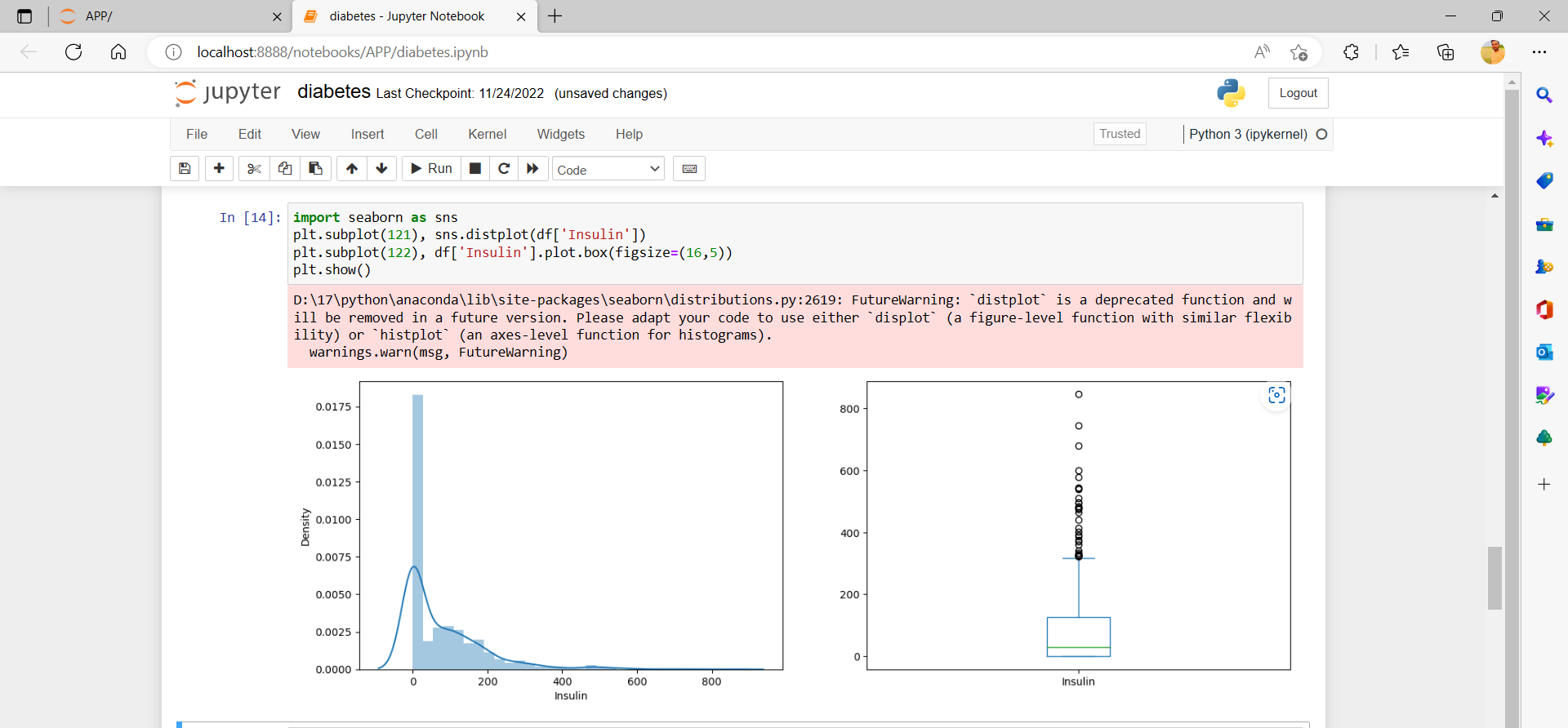


Fig 4.7: dist plot

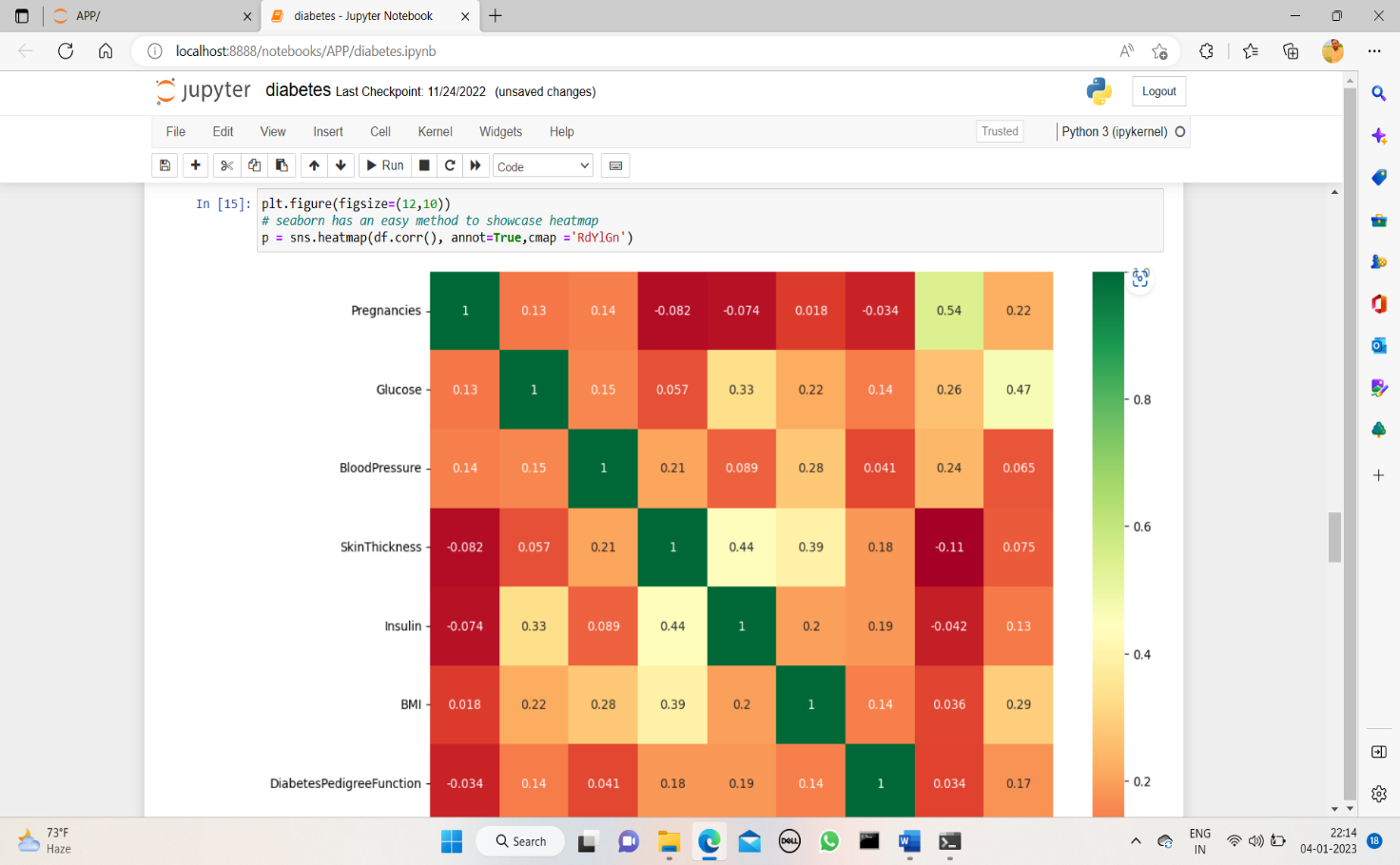


Fig 4.8: heatmap

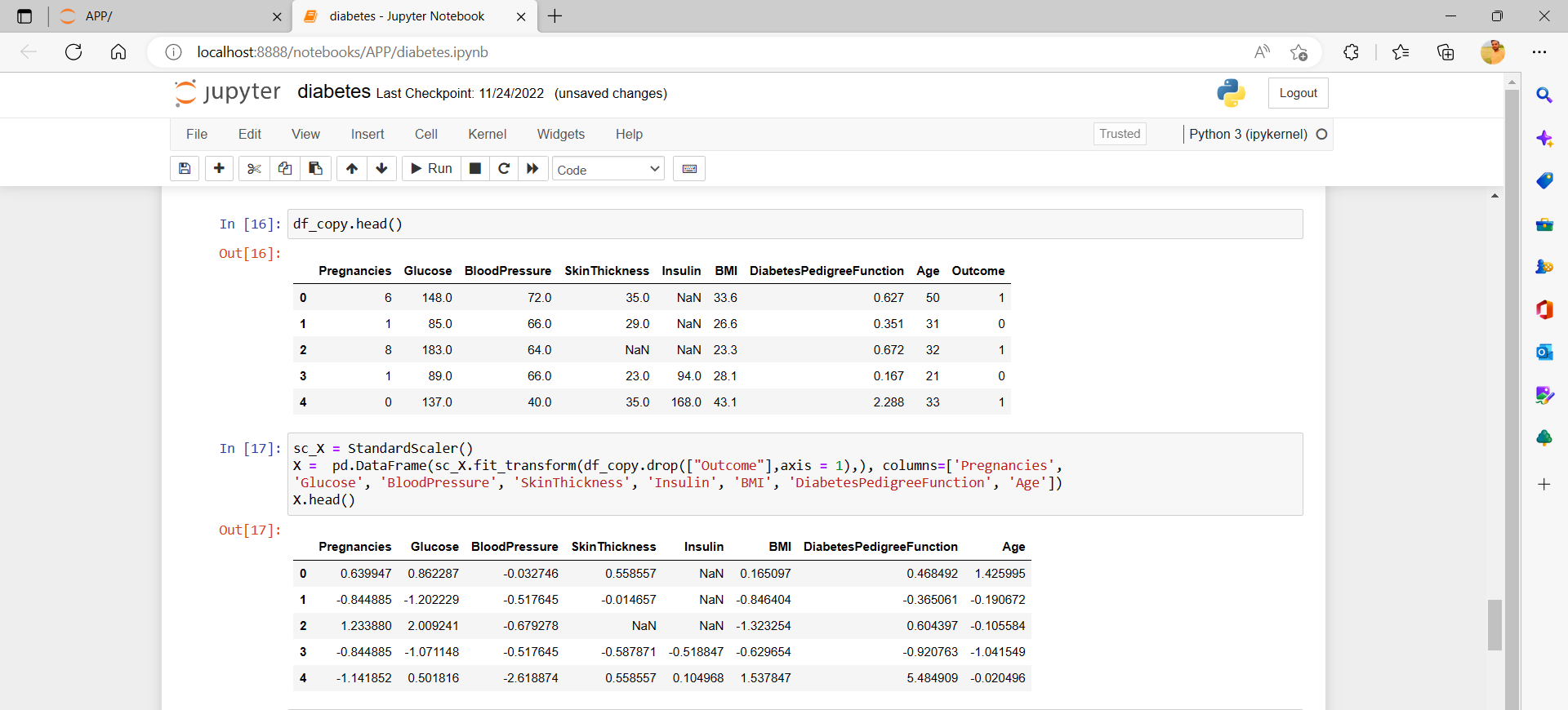


Fig 4.9: declaring of x and y variables

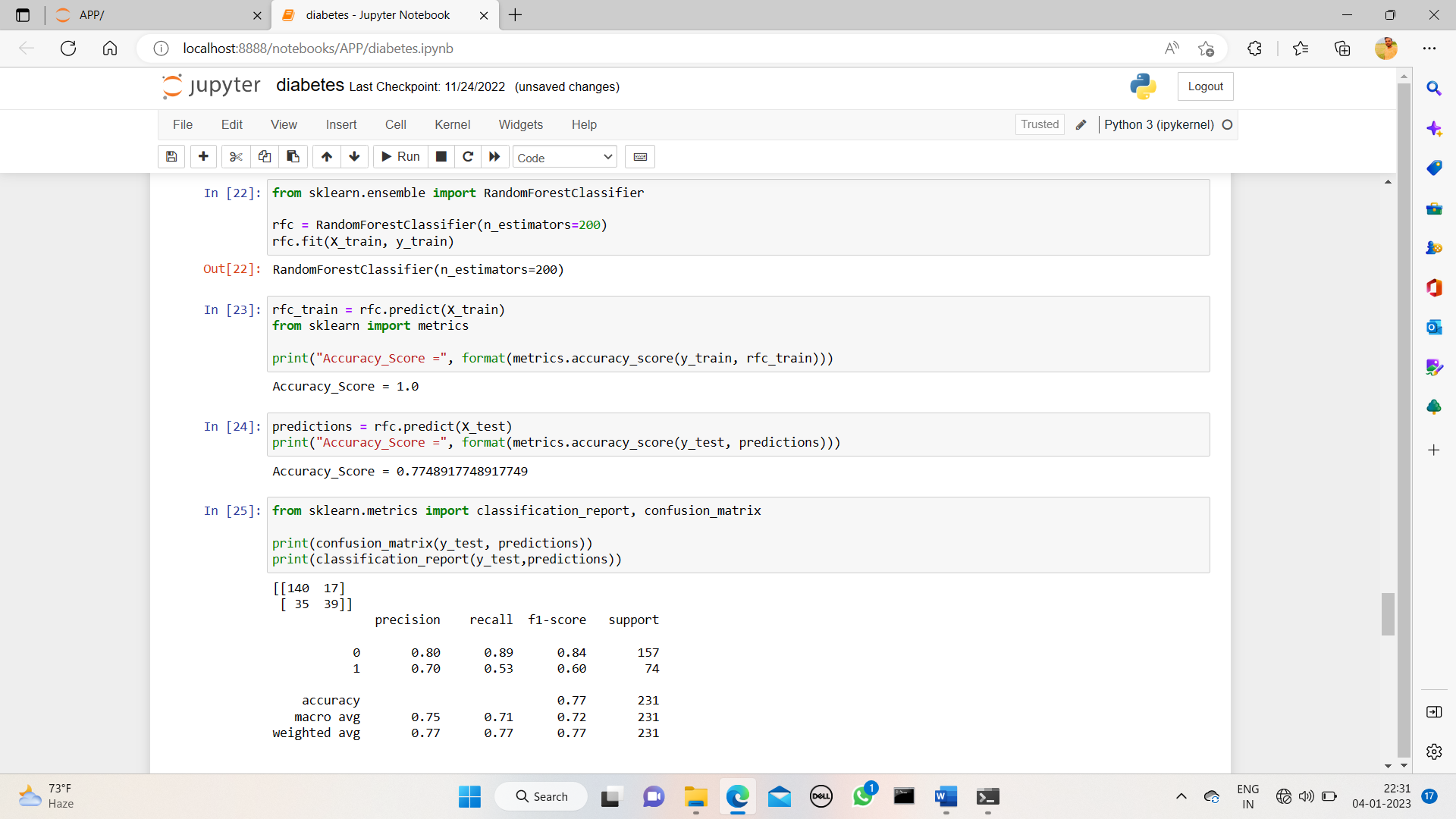


Fig 4.10: importing random forest classifier

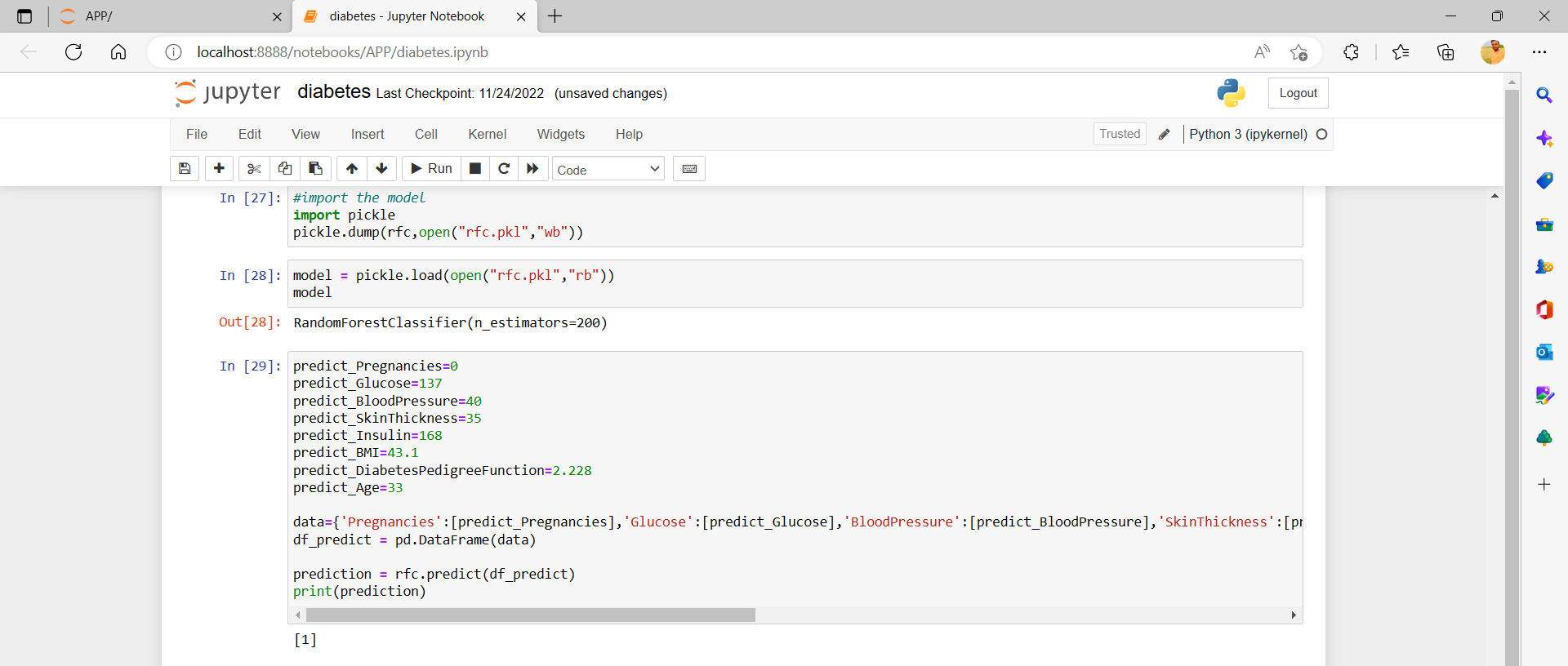


Fig 4.11: import pickle file

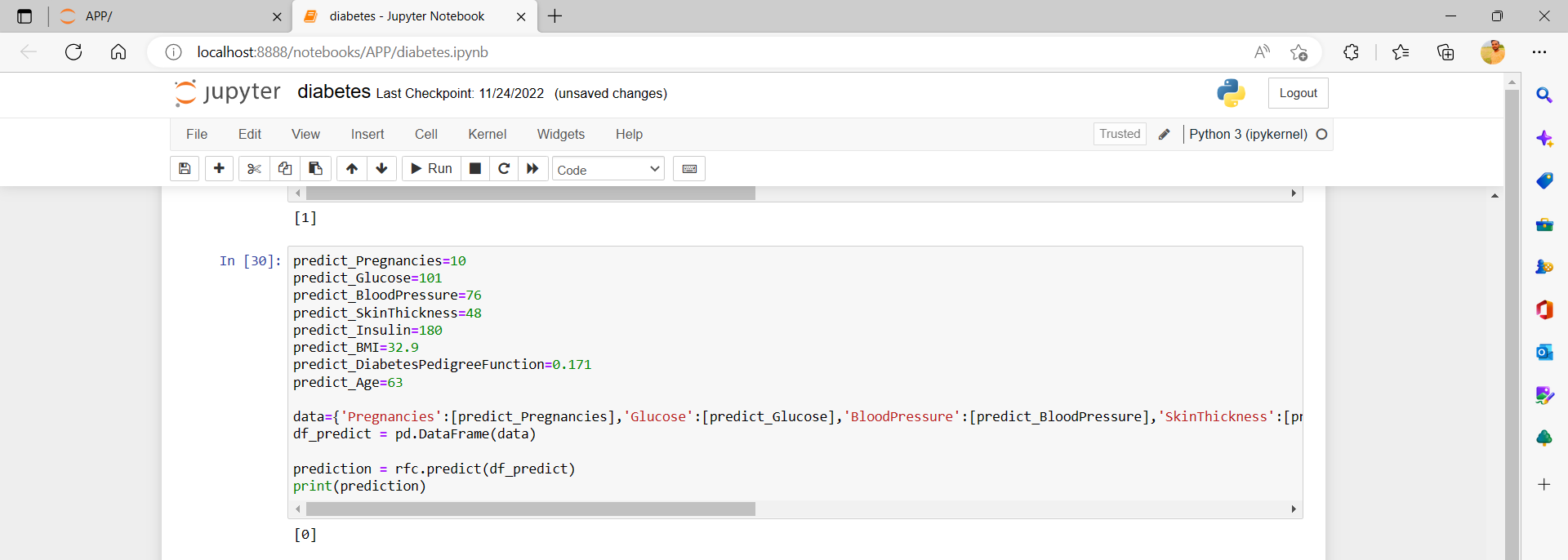
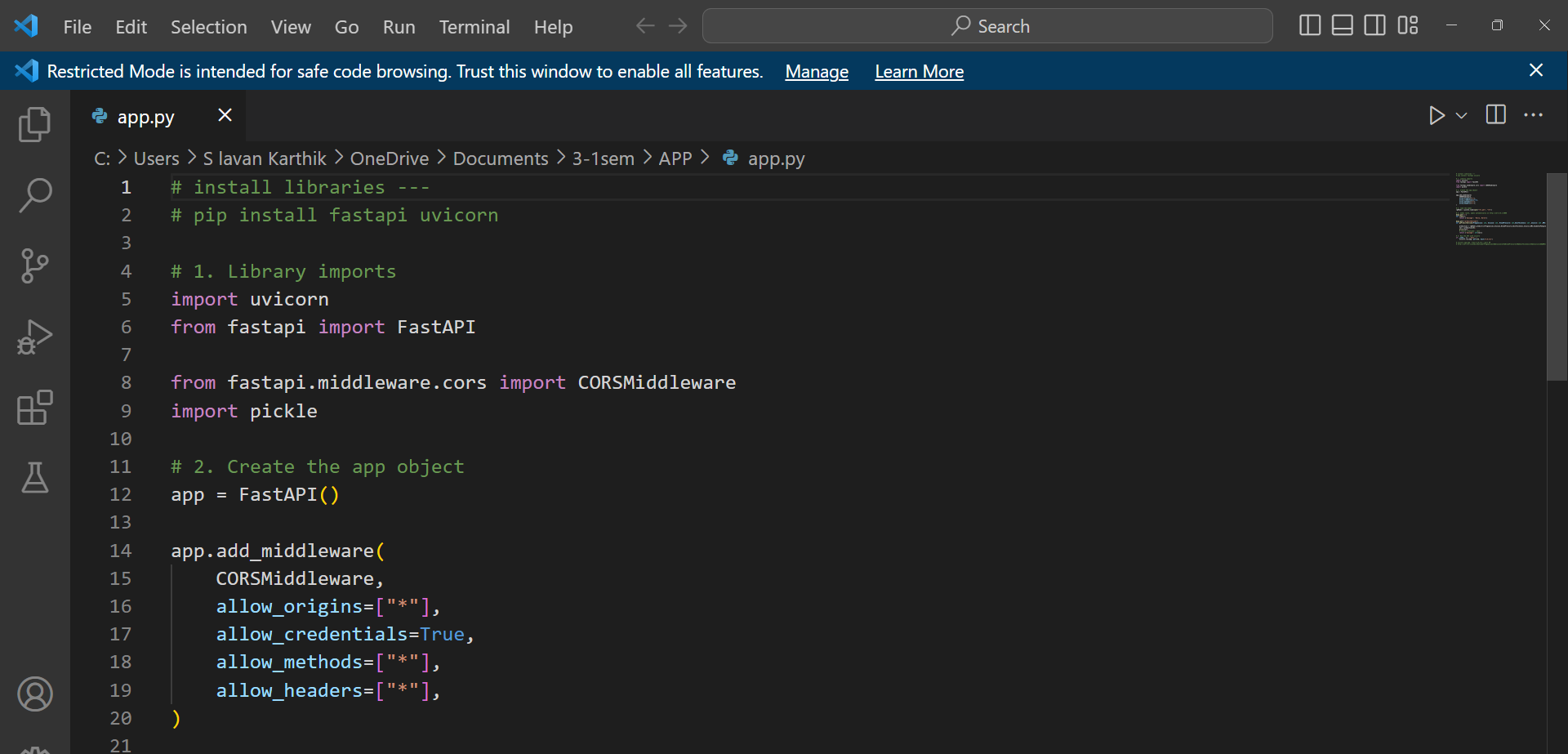


Fig 4.12: predicting diabetes

The above code is the machine learning code that has the algorithm of random forest classification with the pickle file. Then we to create the python code of FastAPI that display the machine learning code in the browser to display the output and the output is come in the webpage.



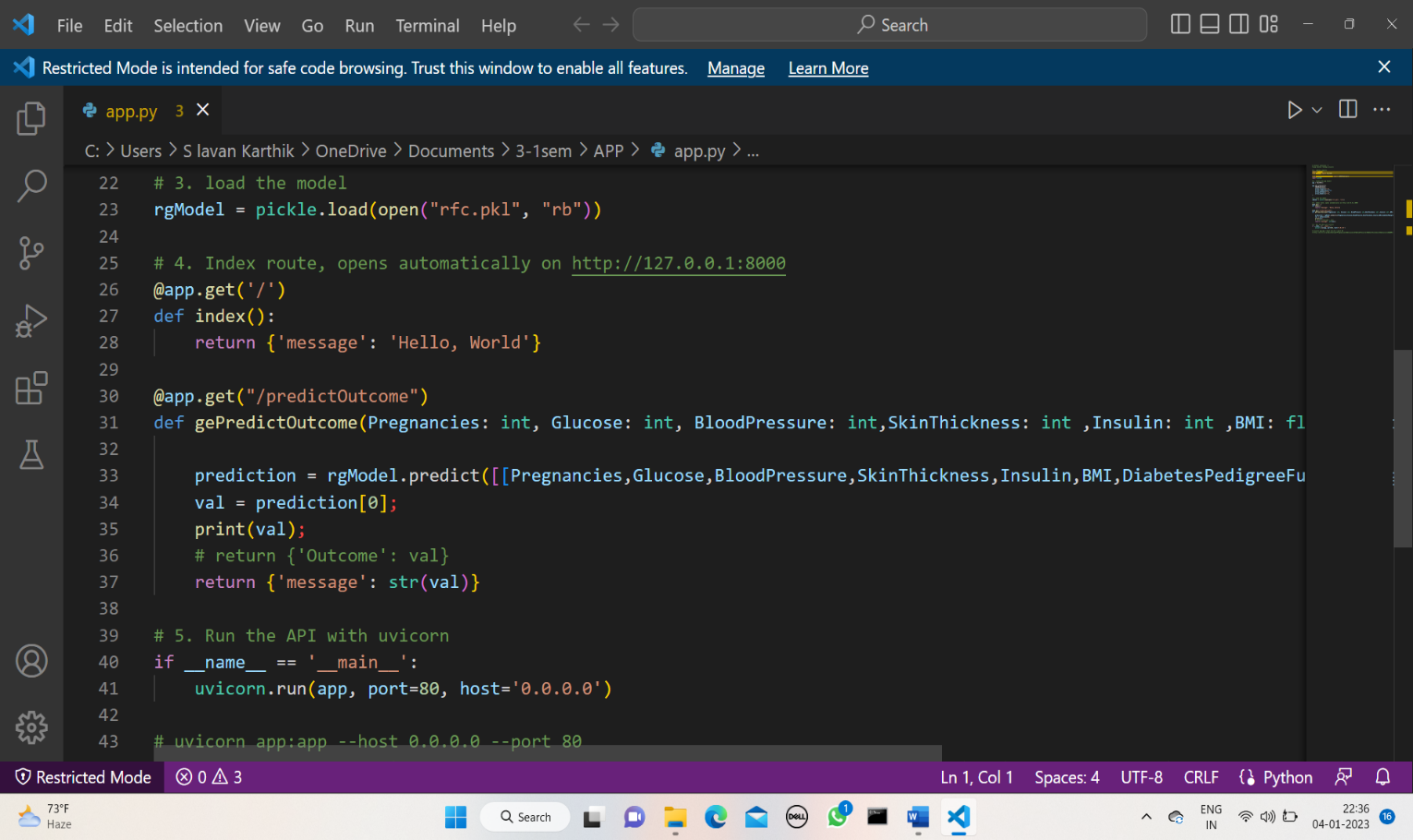


Fig 5: python code

After the python code run the code in the terminal with the command.

**uvicorn app:app --host 0.0.0.0 –port 80** Then run the link in the browser.

Then you got the result in the browser that is output of the machine learning code. Then deploy the code in the AWS management console instance and run the IP address.

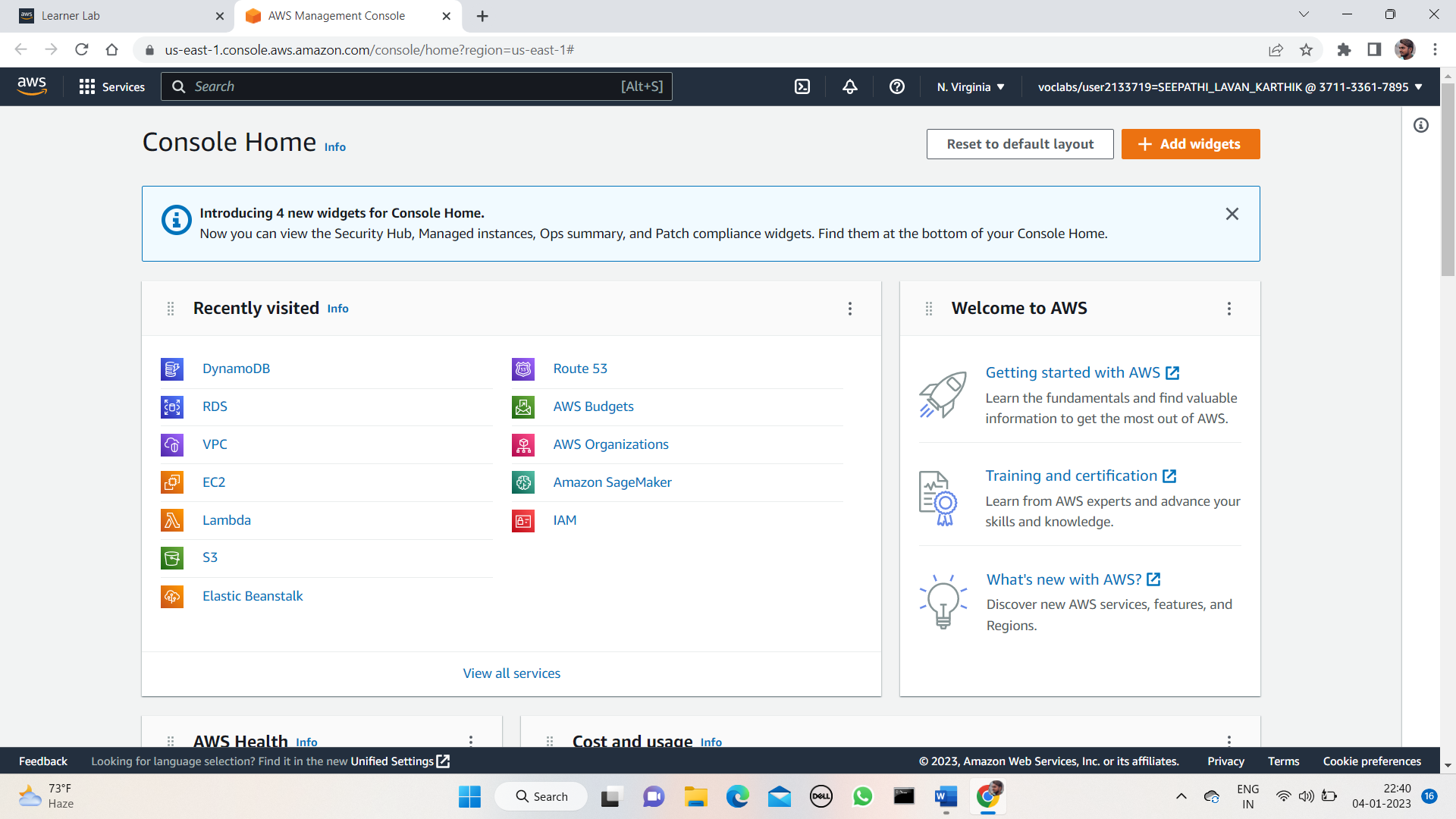


Fig 6: AWS environment

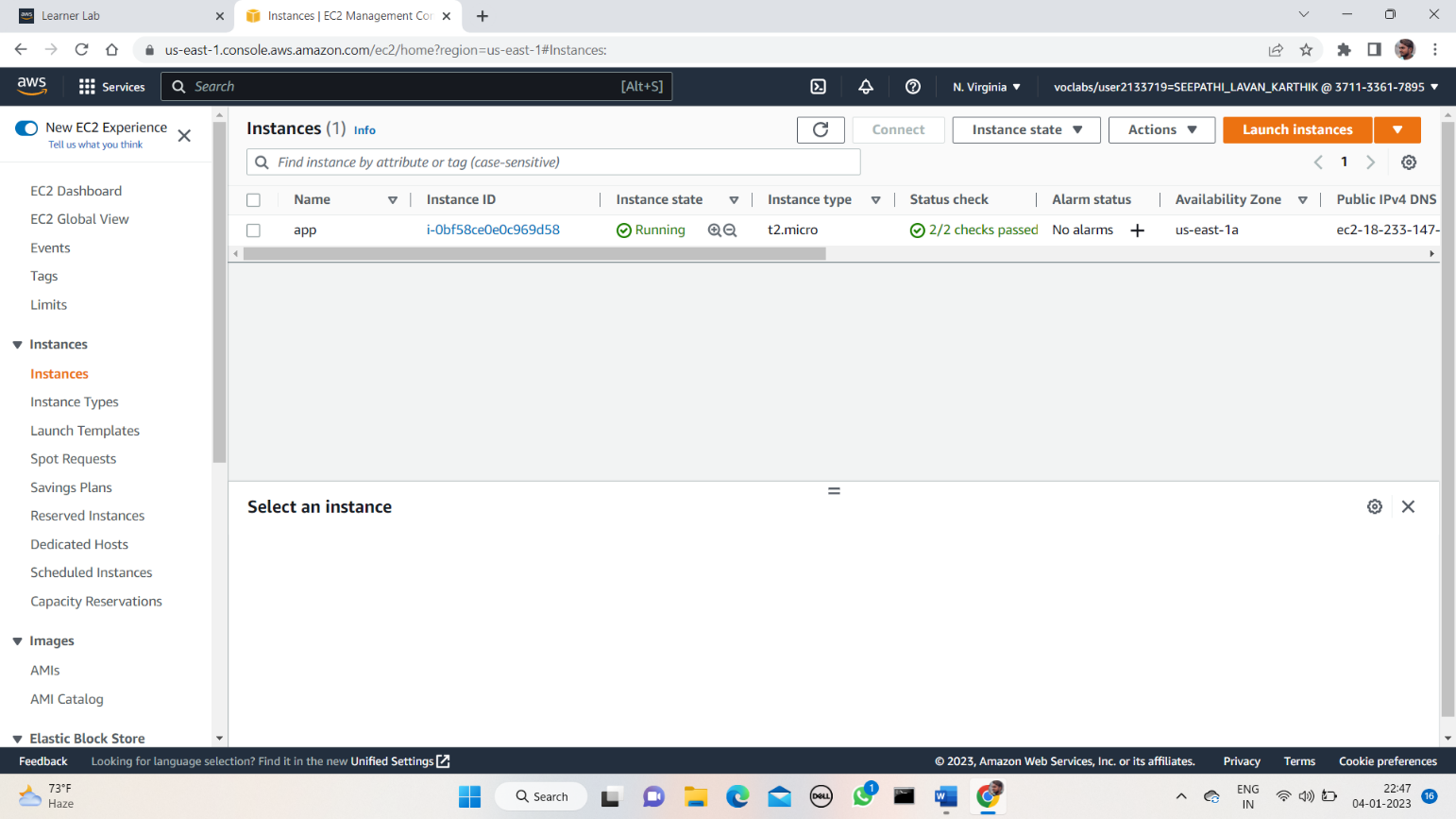


Fig 7: EC2 instance dashboard

**5.OUTPUT SCREENSHOT**

We use the angular for the frontend. To represent the outcome for the machine learning we use the angular application. In the angular we create the service component and we connect the machine learning model to the service by deploying the model in the AWS EC2 instance and that IP address is to be call in the service component in the angular. After completing the connection, we have to run the angular project. To check the project then after you have run the IP address then the angular project should run then after the machine learning model should run in the instances also, we can deploy the angular application in the instance as well as in the s3 bucket of the AWS management console. We see in the instances as well as in the s3 bucket also.

Deploying in the instances:

IP address should run with the port at which it has running.

Graphical user interface, application

Description automatically generated

Fig 8: frontend page

**Case 1:** If the patient has the diabetes, then the outcome would be 1. By giving input values we get the outcome.

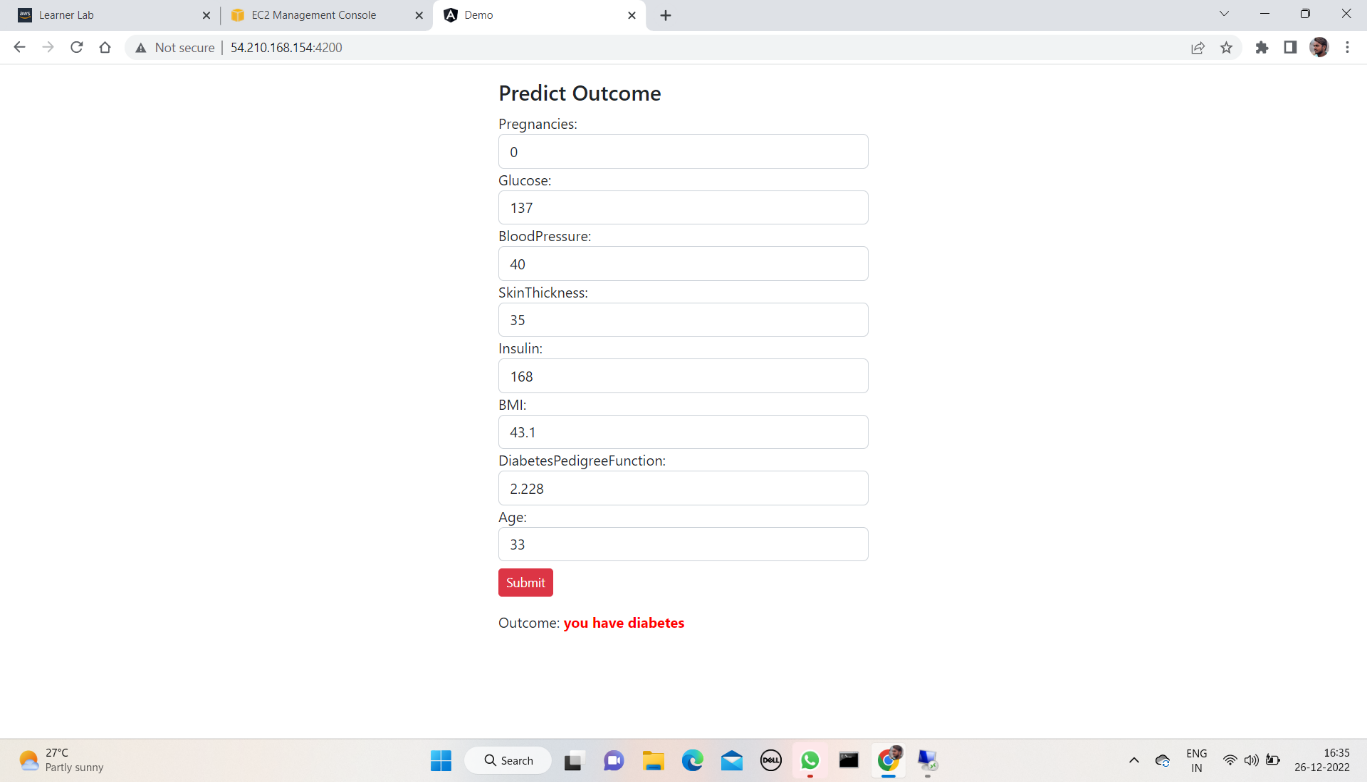
****

Fig 9: **Outcome:** you have diabetes.

**Case 2:** If the patient doesn’t have the diabetes, then the outcome will be. By the given input we can get the outcome.

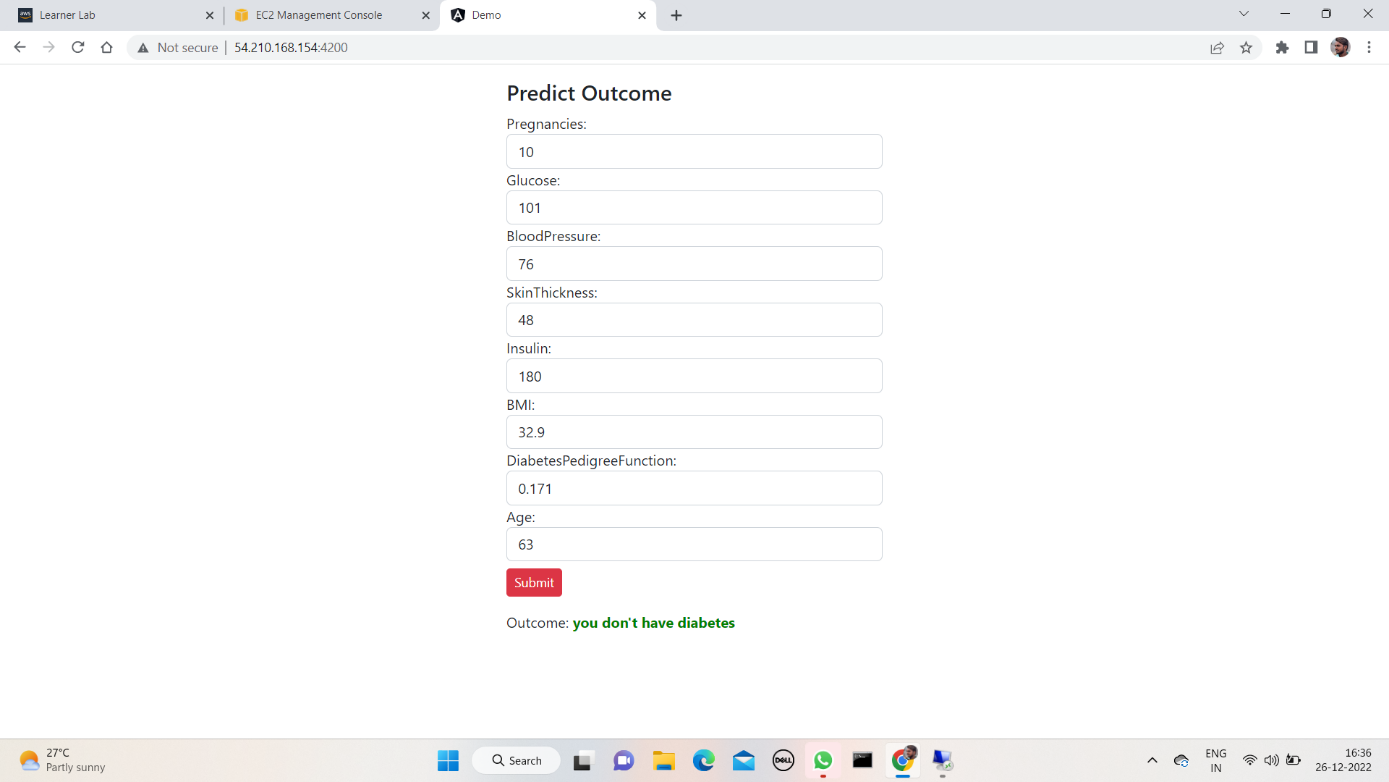
****

Fig 10: **Outcome:** you don’t have diabetes.

# 6.CONCLUSION AND FUTURE ENHANCEMENT

One of the important real-world medical problems is the detection of diabetes at its early stage. In this study, systematic efforts are made in designing a system which results in the prediction of diabetes. During this work, five machine learning classification algorithms are studied and evaluated on various measures. Experiments are performed on john Diabetes Database. Experimental results determine the adequacy of the designed system with an achieved accuracy of 99% using Decision Tree algorithm. In future, the designed system with the used machine learning classification algorithms can be used to predict or diagnose other diseases. The work can be extended and improved for the automation of diabetes analysis including some other machine learning algorithms.

**7.REFERENCES**

1. Alumax, A.A., Ahamad, M.G., Siddiqui, M.K., 2013. Application of data mining: Diabetes health care in young and old patients. Journal of King Saud University - Computer and Information Sciences 25, 127–136. doi:10.1016/j.jksuci.2012.10.003.
2. Arora, R., Suman, 2012. Comparative Analysis of Classification Algorithms on Different Datasets using WEKA. International Journal of Computer Applications 54, 21–25. doi:10.5120/8626-2492.
3. Belmonte, M.P., G.R., 2014. Design of Classifier for Detection of Diabetes Mellitus Using Genetic Programming. Advances in Intelligent Systems and Computing 1, 763–770. doi:10.1007/978-3- 319-11933-5.
4. <https://www.kaggle.com/johndasilva/diabetes>
5. Rani, A. S., & Jyothi, S. (2016, March). Performance analysis of classification algorithms under different datasets. In Computing for Sustainable Global Development (INDIA.Com), 2016 3rd International Conference on (pp. 1584- 1589). IEEE.