PREDICTION ON DIABETES USING MACHINE LEARNING

Mini Project Report

Submitted in partial fulfillment of the requirements for the award of the Degree

of Bachelor of Technology (B. TECH)

in

COMPUTER SCIENCE AND ENGINEERING

By

CH KAVYA	18AG1A0511
M INDU	18AG1A0534
P RATHNAMALA	18AG1A0544

Under the Esteemed Guidance of
Ms. D. Laxmi Rohitha
(Assistant Professor)
Department of Computer Science and Engineering

Department of Computer Science and Engineering

ACEENGINEERINGCOLLEGE

An Autonomous Institution

(NBA ACCREDITED B. TECH COURSES: EEE, ECE, MECH, CIVIL&CSE, ACCORDED NAAC 'A'GRADE)

(Affiliated to Jawaharlal Nehru Technological University, Hyderabad, Telangana)

Ghatkesar, Hyderabad-501 301

February 2022

ACE
Engineering College
An Autonomous Institution
(NBA ACCREDITED B. TECH COURSES: EEE, ECE, MECH, CIVIL&CSE ACCORDED NAAC 'A'GRADE)
Ghatkesar, Hyderabad-501301
(Affiliated to Jawaharlal Nehru Technological University Hyderabad)
Website:www.aceec.ac.inE-mail:info@aceec.ac.in

CERTIFICATE

This is to certify that the Mini project work entitled **"PREDICTION ON DIABETES USING MACHINE LEARNING"** is being submitted by **CH KAVYA (18AG1A0511), M INDU (18AG1A0534), P RATHNAMALA (18AG1A0544)** in partial fulfilment for the award of Degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** to the Jawaharlal Nehru Technological University, Hyderabad during the Academic year 2021-22 is a record of bonafide work carried out by her/him under our guidance and supervision.

The results embodied in this report have not been submitted by the student to any other University or Institution for the award of any degree or diploma.

Internal Guide
Ms.D. Laxmi Rohitha
Assistant Professor

Head of the Department
DR. M.V. Vijaya Saradhi
Professor and Head
Department of CSE

External Examiner

ACKNOWLEDGEMENT

We would like to express my gratitude to all the people behind the screen who have helped me transform an idea into a real time application.

We would like to express my heart-felt gratitude to my parents without whom I would not have been privileged to achieve and fulfil my dreams.

A special thanks to our Secretary, **Prof. Y. V. GOPALA KRISHNA MURTHY,** for having founded such an esteemed institution. I am also grateful to our beloved principal, **Dr. B. L. RAJU** for permitting us to carry out this project.

We profoundly thank **Dr. M. V. Vijaya Saradhi,** Head of the Department of Computer Science and Engineering, who has been an excellent guide and also a great source of inspiration to my work.

We extremely thank **Mr. K PREM KUMAR** Assistant Professor, Project coordinator, who helped us in all the way in fulfilling of all aspects in completion of our Mini-Project.

We are very thankful to my internal guide **Ms. D. Laxmi Rohitha,** Assistant Professor of the Department of Computer Science and Engineering who has been an excellent and given continuous support for the Completion of my project work.

The satisfaction and euphoria that accompany the successful completion of the task would be great, but incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success. In this context, I would like to thank all the other staff members, both teaching and non-teaching, who have extended their timely help and eased my task.

CH KAVYA (18AG1A0511)
M INDU (18AG1A0534)
P RATHNAMALA (18AG1A0544)

DECLARATION

We hereby declare that the project work entitled "**PREDICTION ON DIABETES USING MACHINE LEARNING"** submitted to the **ACE Engineering College**, is a record of an original work done by me under the guidance of **Ms. D. Laxmi Rohitha**, Assistant Professor of the Department of Computer Science and Engineering, **ACE Engineering College**, and this project work is submitted in the partial fulfilment of the requirements for mini project; the results embodied in this thesis have not been submitted to any other university or institute forward of any degree or diploma

CH KAVYA (18AG1A0511)
M INDU (18AG1A0534)
P RATHNAMALA (18AG1A0544)

ABSTRACT

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. In this work we will use Machine Learning Classification and ensemble techniques on a dataset to predict diabetes. Which are K-Nearest Neighbor (KNN), Logistic Regression (LR), Decision Tree (DT), Support Vector Machine (SVM), Gradient Boosting (GB) and Random Forest (RF). The accuracy is different for every model when compared to other models. The Project work gives the accurate or higher accuracy model shows that the model is capable of predicting diabetes effectively. Our Result shows that Random Forest achieved higher accuracy compared to other machine learning techniques.

Keywords: Diabetes, Machine, Learning, Prediction, Dataset, Ensemble.

INDEX

LIST OF FIGURES

CHAPTER-1
INTRODUCTION

1.1. PROBLEM STATEMENT

Diabetes has become a global problem that the manual decisions of doctor's can be alarming. It can be difficult to determine whether a patient having diabetes or not with better accuracy.

Diabetes is a chronic health problem with devastating, yet preventable consequences. It is characterized by high blood glucose levels resulting from defects in insulin production, insulin action, or both.1,2 Globally, rates of type 2 diabetes were 15.1 million in 2000,3 the number of people with diabetes.

1.2. PURPOSE

This project aims to develop a model for predicting diabetes using machine learning algorithm. The main goal is to develop a model that predict whether a patient having diabetes or not.

1.3. INTRODUCTION

Diabetes is a common chronic disease which can pose great threat to human health. Diabetes can be identified when blood glucose is higher than normal level, which is caused by high secretion of insulin or biological effects. Diabetes can cause various damage to our body and can disfunction tissues, kidneys, eyes and blood vessels. Diabetes can be divided into two categories, type 1 diabetes and type 2 diabetes. Patients with type 1 diabetes are normally younger with an age less then 30 years old. The clinical symptoms are increase thirst and frequent urination this type of diabetes cannot be cleared by medications as it requires therapy.

Type 2 diabetes occurs more commonly on middle-aged and old people, which can show hypertension, obesity and other diseases. with our living standards diabetes has increased commonly in people's daily life. So how to analyse diabetes is worth studying. As we get the diagnos is earlier we can control it. Machine learning can make preliminary judgement on diabetes mellitus according to physical examination data, and by reference with doctors.

Recently, many algorithms are used to predict diabetes, including machine learning methods like Random Forest, (KNN) K-Nearest Neighbor, Decision Tree and so on. With this machine learning techniques we are able to predict diabetes by constructing predicting models which are obtained by medical datasets. By extracting such knowledge we are able to predict diabetic patient. We use the best technique to predict based on our attributes of the given datasets in order to get the perfect accuracy to predict diabetes mellitus. 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 crabs 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.

For this purpose we use the Pima Indian Diabetes Dataset, we apply various Machine Learning classification and ensemble Techniques to predict diabetes. Machine Learning Is a method that is used to train computers or machines explicitly. Various Machine Learning Techniques provide efficient result to collect Knowledge by building various classification and ensemble models from collected dataset.

Such collected data can be useful to predict diabetes. Various techniques of Machine Learning can capable to do prediction, however it’s tough to choose best technique. Thus for this purpose we apply popular classification and ensemble methods on dataset for prediction.

1.4.AIM AND OBJECTIVE

The aim of this project is to develop a system which can perform early prediction of diabetes for a patient with a higher accuracy by combining the results of different machine learning techniques. This project aims to predict diabetes via three different supervised machine learning methods including: SVM, Logistic regression, Linear regression. This project also aims to propose an effective technique for earlier detection of the diabetes disease.

1.4. MOTIVATION

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so.

Machine learning algorithms use historical data as input to predict new output values. The extensive spread of faux news can have a significant negative impact on individuals and society.

There has been drastic increase in rate of people suffering from diabetes since a decade. Current human lifestyle is the main reason behind growth in diabetes. In current medical diagnosis method, there can be three different types of errors- 1. The false-negative type in which a patient in reality is already a diabetic patient but test results tell that the person is not having diabetes.

2. The false-positive type. In this type, patient in reality is not a diabetic patient but test reports say that he/she is a diabetic patient. 3. The third type is unclassifiable type in which a system cannot diagnose a given case. This happens due to insufficient knowledge extraction from past data, a given patient may get predicted in an unclassified type.

However, in reality, the patient must predict either to be in diabetic category or non-diabetic category. Such errors in diagnosis may lead to unnecessary treatments or no treatments at all when required. In order to avoid or reduce severity of such impact, there is a need to create a system using machine learning algorithm and data mining techniques which will provide accurate results and reduce human efforts.

1.6. SCOPE

Machine Learning deals with the development, analysis and study of algorithms that can automatically detect patterns from data and use it to predict future data or perform decision making. Machine learning does its functionality by creating models out of it. Machine Learning has become widespread and has its applications in the field of bio informatics, computer vision, robot locomotion, computational finance, search engine etc.

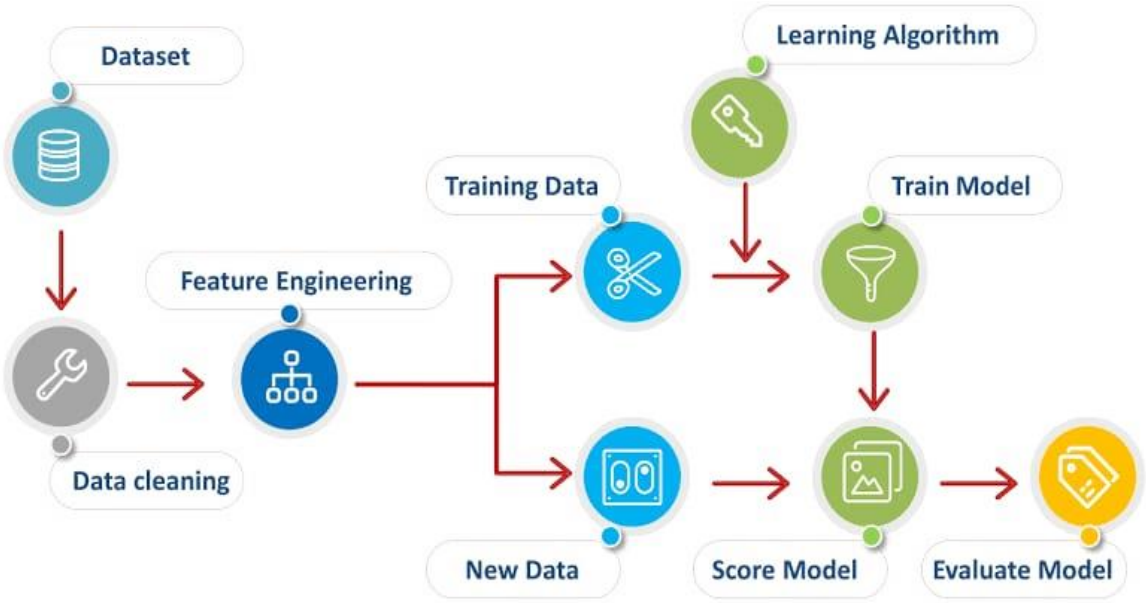


Fig 1.1. Machine Learning Structure

Fig. 1.1 Example: The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

CHAPTER-2

LITERATURE SURVEY

2.1. SYSTEM REVIEW

We went through different sites and applications and looked for the fundamental data. Based on these data, we made an audit that helped us get new thoughts and make different arrangements for our task to find an appropriate algorithm for the classification.

2.2. TECHNOLOGY USED

PYTHON- Python is an interpreted, high-level, general purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

COLAB - Colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members.

2.3. EARLIER MODEL SURVEY

K. Vijiya Kumar 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.

Tejas N. Joshi et al. 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 et al. proposed diabetes disease prediction using data mining assemble Intelligent Diabetes Disease Prediction System that gives analysis of diabetes malady utilizing diabetes patient's database. In this system, they propose the use of algorithms like Bayesian and KNN (K-Nearest Neighbor) to apply on diabetes patient's database and analyze them by taking various attributes of diabetes for prediction of diabetes disease. Muhammad Azeem Sarwar et al. 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.

In Amour Diwani's et al.'s study, all the patient's data are trained and tested using 10 cross-validations with Naïve Bayes and decision tree. Then the performance was evaluated, investigated, and compared with other classification algorithms. The results predicted that the best algorithm is Naïve Bayes with an accuracy of 65.3021%.

In Zou et al.'s study, they applied Random Forest, Decision Tree, ANN for classification algorithm on PIDD after the feature reduction using Principal Component Analysis and Minimum Redundancy. They found that Pima Indian's best accuracy is 77.21% obtained from the random forest with feature reduction method.

CHAPTER-3
SYSTEM ANALYSIS

3.1. EXISTING SYSTEM

Existing many research handled for diabetes detection. Data mining approach like clustering, classification were studied in existing system. Diabetes prediction using algorithms such as k- Nearest Neighbour (KNN), 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.

Prediction using traditional methods and models involves various risk factors and it consists of various measures of algorithms such as datasets, programs and much more to add on. High-risk and Low-risk patient classification is done on the basis of the tests that are done in group. But these models are only valuable in clinical situations and not in big industry sector. So, to include the disease predictions in various health related industries, we have used the concepts of machine learning and supervised learning methods to build the predictions system.

After doing the research and comparison of all the algorithms and theorems of machine learning we have come to conclusion that all those algorithms such as Decision Tree, KNN, Naïve Bayes, Regression and Random Forest Algorithm all are important in building a disease prediction system which predicts the disease of the patients from which he/she is suffering from and to do this we have used some performance measures like ROC, KAPPA Statistics, RMSE, MEA and various other tools. After using various techniques such as neural networks to make predictions of the diseases and after doing that we come to conclusion that it can predicts up to 90% accuracy rate after doing the experimentation and verifying the results. The information of patient statistics, results, disease history in recorded in EHR, which enables to identify the potential data centric solution, which reduces the cost of medical case studies. Existing system can predict the disease but not the sub type of the disease and it fails to predict the condition of the people, the predictions of disease have been indefinite and non-specific.

The healthcare industry collects an enormous amount of data include hospital records, medical records of patients, and results of medical examinations.

For early disease diagnosis, the disease's prediction is analyzed through a doctor's experience and knowledge, but that can be inaccurate and susceptible.

Hence the manual decisions can be alarming. The hidden pattern of data can be unnoticed, which can impact decision-making; therefore, patients become deprived of the appropriate treatment.

ISSUES IN EXISTING SYSTEM:

Using machine learning the accuracy of detection is less High false positives

There is no interactive tool for users to predict diabetes.

3.2. PROPOSED MODEL

The proposed system of disease prediction using machine learning is that we have used many techniques and algorithms and all other various tools to build a system which predicts the disease of the patient using the symptoms and by taking those symptoms we are comparing with the system's dataset that is previously available. By taking those datasets and comparing with the patient's disease we will predict the accurate percentage disease of the patient. The dataset and symptoms go to the prediction model of the system where the data is pre-processed for the future references and then the feature selection is done by the user where he will enter the various symptoms.

Then the classification of those data is done with the help of various algorithms and techniques such as Decision Tree, KNN, Naïve Bayes, Random Forest and etc.

Then the data goes in the recommendation model, there it shows the risk analysis that is involved in the system and it also provides the probability estimation of the system such that it shows the various probability like how the system behaves when there are n number of predictions are done and it also does the recommendations for the patients from their final result and also from their symptoms like it can show what to use and what not to use from the given datasets and the final result.

Here we have combined the overall structure and unstructured form of data for the overall risk analysis that is required for doing the prediction of the disease.

Using the structured analysis, we can identify the chronic types of disease in a particular region and particular community.

In unstructured analysis we select the features automatically with the help of algorithms and techniques.

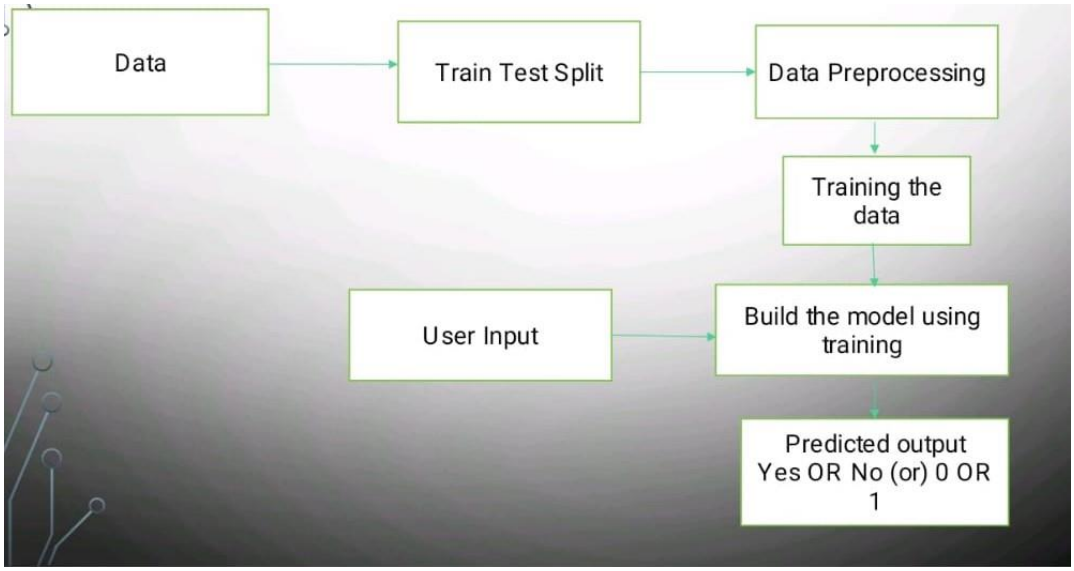


Fig 3.1 work flow of the proposed system

This system takes symptoms from the user and predicts the disease accordingly based on the symptoms that it takes and also from the previous datasets, it also helps in continuous evaluation of viral diseases, heart rate, blood pressure, sugar level and much more which is in the system and along with other external symptoms its predicts the appropriate and accurate We aim to develop a model that can predict the diabetes. The model is built by implementing the Machine Learning with Linear and Logistic Regression algorithm. The developed model show that whether a patient having diabetes or not.

3.3. PROBLEM DEFINITION

Now a day's in Health Industry there are various problems related to machines or devices which will give wrong or unaccepted results, so to avoid those results and get the correct and desired results we are building a program or project which will give the accurate predictions based on information provided by the user and also based on the datasets that are available in that machine. The health industry in information yet and knowledge poor and this industry is very vast industry which has lot of work to be done. So, with the help of all those algorithms, techniques and methodologies we have done this project which will help the peoples who are in the need. So the problem here is that many people
goes to hospitals or clinic to know how is their health and how much they are improving in the given days, but they have to travel to get to know there answers and sometimes the patients may or may not get the results based on various factors such as doctor might be on leave or some whether problem so he might not have come to the hospital and many more reasons will be there so to avoid all those reasons and confusion we are making a project which will help all those person's and all the patients who are in need to know the condition of their health, and at sometimes if the person has been observing few symptoms and he/she is not sure about the disease he/she is encountered with so this will lead to various diseases in future. So, to avoid that and get to know the disease in early stages of the symptoms this disease prediction will help a lot to the various
people's ranging from children to teenagers to adults and also the senior citizens.

3.4. SYSTEM REQUIREMENT SPECIFICATION

3.4.1. FEASIBILITY

A feasibility report is a testimony that attempts to create some sort of action. Feasibility reports are created to persuade/help the decision makers to choose between available options. Remember that your option is not the only one, the decision makers will probably have many to choose from.

A feasibility report also determines whether or not the investigated task can be done with the amount of resources available OR how many resources will be necessary in order to complete the task. A feasibility may be useful in a lot of different situations such as event planning, modeling etc.

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

1. Economic Feasibility
2. Technical Feasibility
3. Operational Feasibility

1.ECONOMIC FEASIBILITY

In the economically feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It requires basic software and hard ware resources. There is nominal expenditure and economic feasibility for certain.

2.TECHNICAL FEASIBILITY

The technical issue usually raised during the feasibility stage of the investigation includes the following:

Does the necessary technology exist to do what is suggested?

Do the proposed equipment have the technical capacity to hold the data required to use the new system?

Can the system be upgraded if developed?

Are there technical guarantees of accuracy, reliability and ease of access?

The current system developed is technically feasible. It uses latest python technology using machine learning techniques. This system uses MS Excel database to hold the data required to use the new system. It uses python and ML algorithms which compiles and process the data very fast compared to other technologies.

3.OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned out into information system that will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following:

Will the system be used and work properly if it is being developed and implemented?
Will there be any resistance from the user that will undermine the possible application benefits?

There is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer.

3.4.2. FUNCTIONAL REQUIREMENTS

A Functional requirement defines a function of a system or its component. A function is described as a set of inputs, the behaviour, and outputs. Functional requirements maybe calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

Behavioural requirements describing all cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

Functional Requirements concerns with the specific functions delivered by the system. So, Functional requirements are statements of the services that the system must provide.

The functional requirements of the system should be both complete and consistent Completeness means that all the services required by the user should be defined. Consistency means that requirements should not have any contradictory definitions. The requirements are usually described in a fairly abstract way. However, functional system requirements describe the system function in details, its inputs and outputs, exceptions and so on.

3.4.3. NON-FUNCTIONAL REQUIREMENTS

Non-functional Requirements refer to the constraints or restrictions on the system. They may relate to emergent system properties such as reliability, response time and store occupancy or the selection of language, platform, implementation techniques and tools.

• The non-functional requirements can be built on the basis of needs of the user, budget constraints, organization policies and etc.

1. Performance requirement: All data entered shall be up to mark and no flaws shall be there for the performance to be 100%.

2. Platform constraints: The main target is to generate an intelligent system to predict the adult height.

3.Accuracy and Precision: Requirements are accuracy and precision of the data.

4. Modifiability: Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).

5. Portability: Since mobile phone is handy so it is portable and can be carried and used whenever required.

6. Reliability: Requirements about how often the software fails. The definition of a failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement. Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error Prediction, and a strategy for correction.

7. Security: One or more requirements about protection of your system and its data.

8. Usability: Requirements about how difficult it will be to learn and operate the system. The requirements are often expressed in learning time or similar metrics.

ACCESSIBILITY:

Accessibility is a general term used to describe the degree to which a product, device, service, or environment is accessible by as many people as possible. In our project people who have registered with the cloud can access the cloud to store and retrieve their data with the help of a secret key sent to their email ids. User interface is simple and efficient and easy to use.

MAINTAINABILITY:

In software engineering, maintainability is the ease with which a software product can be modified in order to include new functionalities can be added in the project based on the user requirements just by adding the appropriate files to existing project using .net and programming languages. Since the programming is very simple, it is easier to find and correct the defects and to make the changes in the project.

SCALABILITY:

System is capable of handling increase total throughput under an increased load when resources (typically hardware) are added. System can work normally under situations such as low bandwidth and large number of users.

PORTABILITY:

Portability is one of the key concepts of high-level programming. Portability is the software code base feature to be able to reuse the existing code instead of creating new code when moving software from an environment to another. Project can be executed under different operation conditions provided it meet its minimum configurations. Only system files and dependent assemblies would have to be configured in such case.

VALIDATION:

It is the process of checking that a software system meets specifications and that it fulfils its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle.

Software validation checks that the software product satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements, not as specification artefacts or as needs of those who will operate the software only; but, as the needs of all stakeholders.

3.4.4 SYSTEM REQUIREMENTS

3.4.4.1. SOFTWARE REQUIREMENTS

- OPERATING SYSTEM : Windows 10 or Higher Versions
- PLATFORM : Google's collab
- PROGRAMMING LANGUAGE : Python
- DATABASE : MS EXCEL

3.4.4.2. HARDWARE REQUIREMENTS

- Processor : Intel core i5 and above
- Hard Disk : 512 or above
- RAM : 8GB or above
- Internet : 4 Mbps or above(Wireless)

CHAPTER -4
SYSTEM DESIGN

4.1. INTRODUCTION

Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement has been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software. The importance can be stated with a single word "Quality".

Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system.

System Design also called top-level design aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of the system design all the major data structures, file formats, output formats, and the major modules in the system and their specifications are decided. During, Detailed Design, the internal logic of each of the modules specified in system design is decided. During this phase, the details of the data of a module is usually specified in a high-level design description language, which is independent of the target language in which the software will eventually be implemented.

In system design the focus is on identifying the modules, whereas during, detailed, design the focus is on designing the logic for each of the modules. In other works, in system design the attention is on what components are needed, while in detailed design how the components can be implemented in software is the issue. Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blueprint for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is minimal clearly specified.

4.2. PYTHON

Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by metaprogramming and metaobjects. Many other paradigms are supported via extensions, including design by contract and logic programming. Python uses dynamic typing and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Python's developers strive to avoid premature optimization, and reject patches to noncritical parts of CPython that would offer marginal increases in speed at the cost of clarity.

When speed is important, a Python programmer can move time-critical functions to extension modules written in languages such as C, or use PyPy, a just-in-time compiler. Cython is also available, which translates a Python script into C and makes direct C-level API calls into the Python interpreter.

An important goal of Python's developers is keeping it fun to use. Python's design offers some support for functional programming in the Lisp tradition. It has filter, map, and reduce functions, list comprehensions, dictionaries, sets, and generator expressions. The standard library has two modules (itertools and function tools) that implement functional tools borrowed from Haskell and Standard ML.

BENEFITS OF PYTHON

- Presence of Third-Party Modules
- Extensive Support Libraries
- Open Source and Community Development
- Learning Ease and Support Available
- User-friendly Data Structures
- Productivity and Speed
- Highly Extensible and Easily Readable Language.

4.3. LIBRARIES USED

Sklearn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.
This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Scikit-learn library is focused on modeling the data. Some of the most popular groups of models provided by Sklearn are as follows −

Supervised Learning algorithms − Almost all the popular supervised learning algorithms, like Linear Regression, Support Vector Machine (SVM), Decision Tree etc., are the part of scikit-learn.

Unsupervised Learning algorithms − On the other hand, it also has all the popular unsupervised learning algorithms from clustering, factor analysis, PCA (Principal Component Analysis) to unsupervised neural networks.

Clustering − This model is used for grouping unlabeled data.

Cross Validation − It is used to check the accuracy of supervised models on unseen data.

Dimensionality Reduction − It is used for reducing the number of attributes in data which can be further used for summarisation, visualisation and feature selection.

Ensemble methods − As name suggest, it is used for combining the predictions of multiple supervised models.

Feature extraction − It is used to extract the features from data to define the attributes in image and text data.

Feature selection − It is used to identify useful attributes to create supervised models.

Open Source − It is open-source library and also commercially usable under DBSD license.

numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.
Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

pandas

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

matplotlib

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

seaborn

Seaborn is a library mostly used for statistical plotting in Python. It is built on top of Matplotlib and provides beautiful default styles and color palettes to make statistical plots more attractive.

4.4 SYSTEM ARCHITECTURE DESIGN

Architecture is nothing but an abstract description of entities in a system. It defines the relations between them and involves a series of decision-making processes. The architecture is a vision and a structure. A system architecture diagram is the distribution of the functional correspondences. These are formal elements, the embodiment of concepts and information. Architecture defines the relations between elements, amongst features, and the surrounding elements. Creating an Architecture diagram is not easy. The examples aim to make things easy for people to understand.

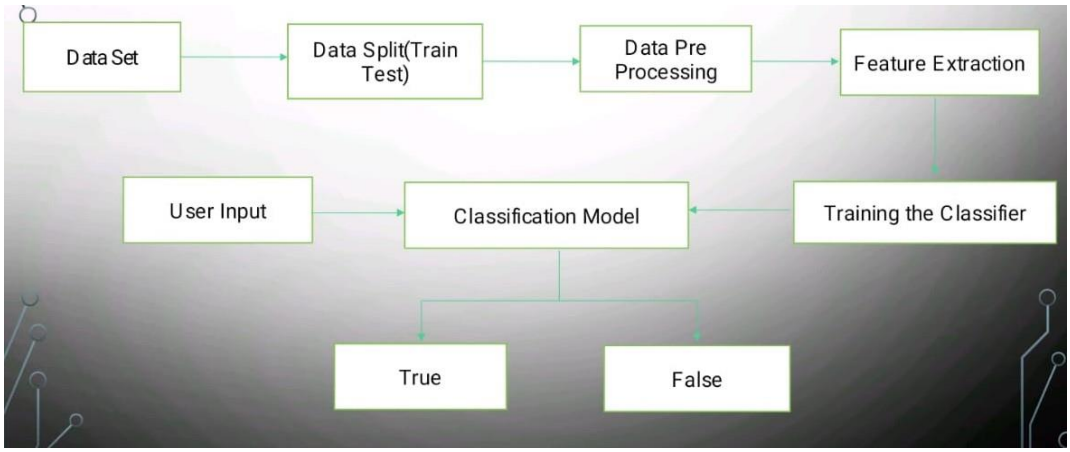


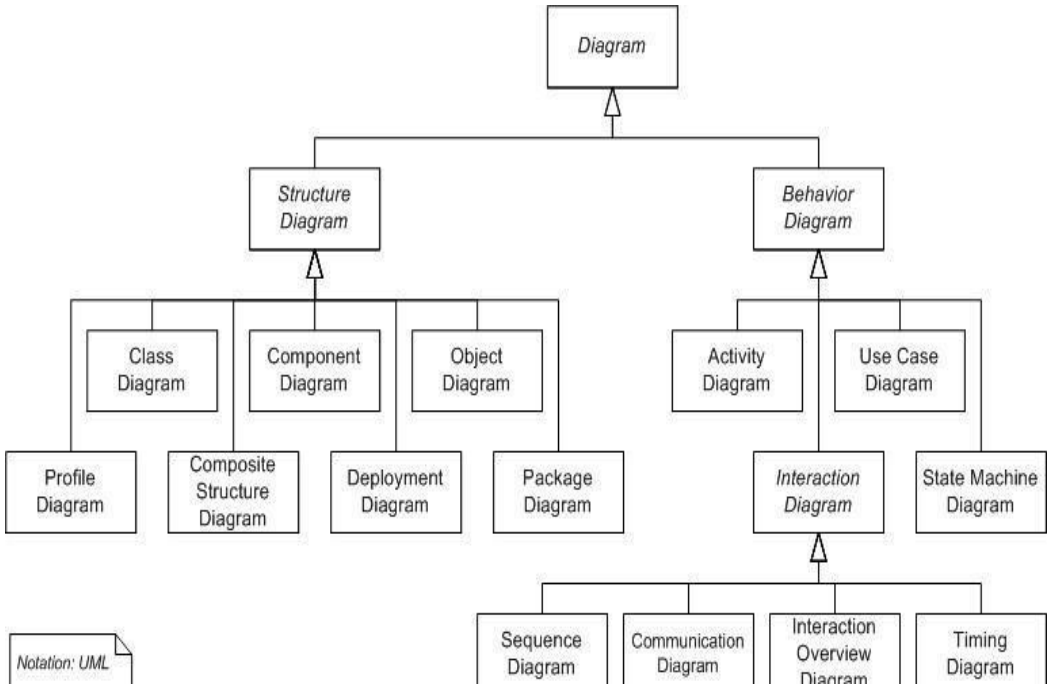
Fig.4.1. System Architecture

4.5 Data Flow Diagrams:

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of data, and delays in the system. The DFD is also known as a data flow graph or a bubble chart.

UML Diagrams Overview:

UML combines best techniques from data modelling (entity relationship diagrams), business model (workflows), object modelling, and component modelling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies.



4.2. Data Flow Diagram

UML has synthesized the notations of the Booch method, the Object-modelling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them in to a single, common and widely usable modelling language. UML aims to be a standard modeling language which can model concurrent and distributed system.

4.5.1 UML DIAGRAMS ON PREDICTING THE DIABETES USING MACHINE LEARNING ALGORITHMS

4.5.1.1 USECASE DIAGRAM

In Use Case diagram, system takes the initiative to drive classify the text, observe extract features, track features and make classification.

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

Purpose of Use Case Diagrams

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.

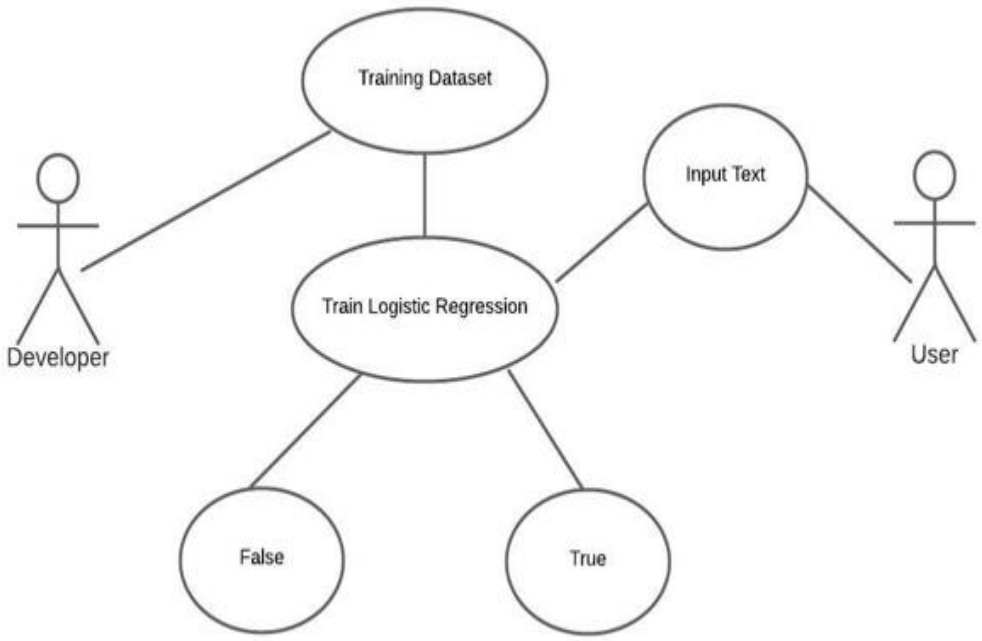


Fig.4.3. Use Case Diagram

4.5.1.2 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It isa construct of a Message Sequence Chart. A sequence diagram shows, as parallel vertical lines ("lifelines"), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical mannar.

The sequence diagram represents the flow of messages in the system and is also termed as an event diagram. It helps in envisioning several dynamic scenarios. It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time.

In UML, the lifeline is represented by a vertical bar, whereas the message flow is represented by a vertical dotted line that extends across the bottom of the page. It incorporates the iterations as well as branching

Purpose of a Sequence Diagram

To model high-level interaction among active objects within a system. To model interaction among objects inside a collaboration realizing a use case. It either models generic interactions or some certain instances of interaction.

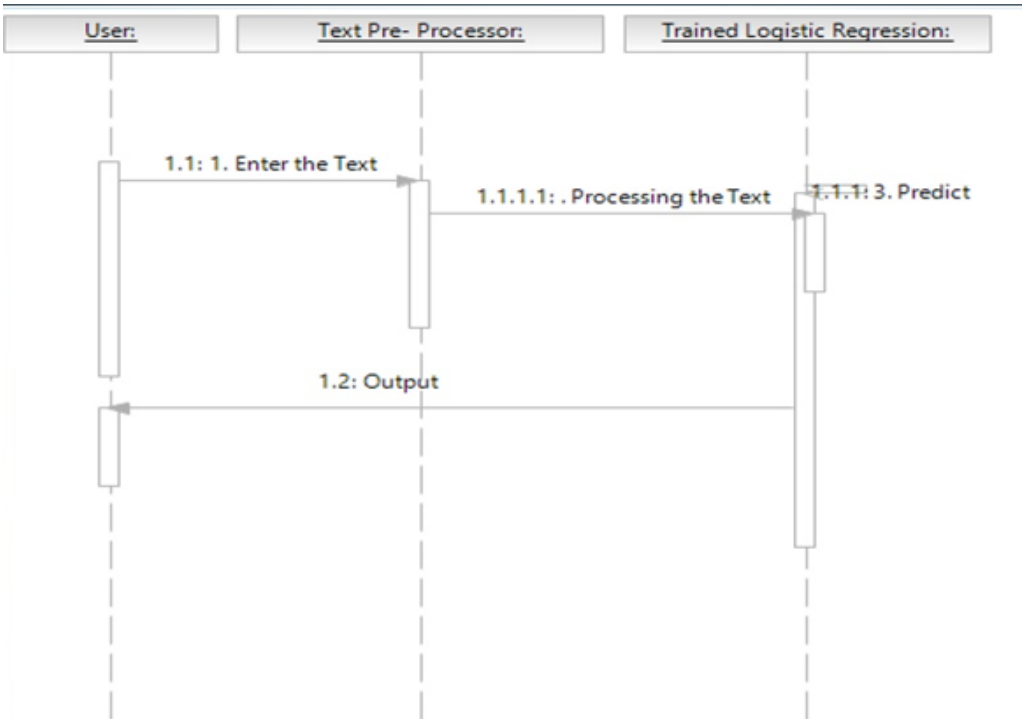


Fig.4.4. Sequence Diagram

4.5.1.3 CLASS DIAGRAM

Disease prediction using machine learning consist of class diagram that all the other application that consists the basic class diagram, here the class diagram is the basic entity that is required in order to carry on with the project. Class diagram consist information about all the classes that is used and all the related datasets, and all the other necessary attributes and their relationships with other entities, all these information is necessary in order to use the concept of the prediction.

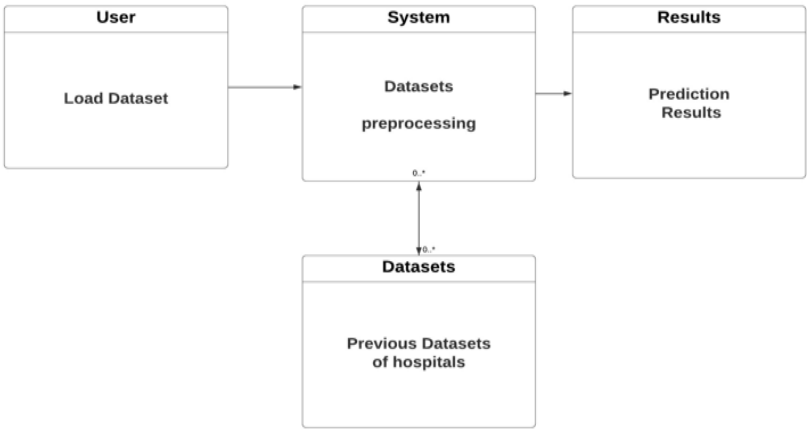


Fig.4.5. Class Diagram

4.5.1.4. STATE CHART DIAGRAM

A State chart diagram describes the behaviour of a single object in response to a series of events in a system. Sometimes it's also known as a Harel state chart or a state machine diagram. This UML diagram models the dynamic flow of control from state to state of a particular object within a system. It is similar to activity diagram but here there are only few rules like how it starts and how it end all are denoted with the help of the symbol given below,

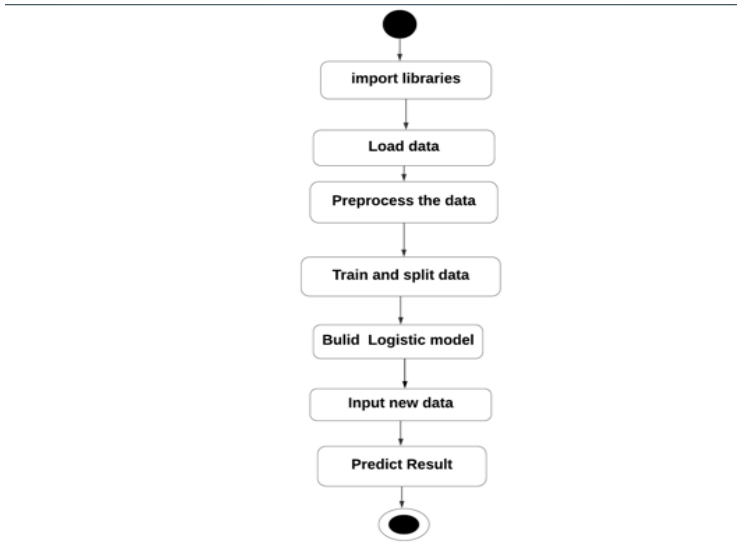


Fig.4.6. State chart Diagram

4.6. ALGORITHMS

4.6.1Linear Regression:

Linear Regression and Logistic Regression are the two famous Machine
Learning Algorithms which come under supervised learning technique. Since both the algorithms are of supervised in nature hence these algorithms use labeled data set to make the predictions. But the main difference between them is how they are being used.

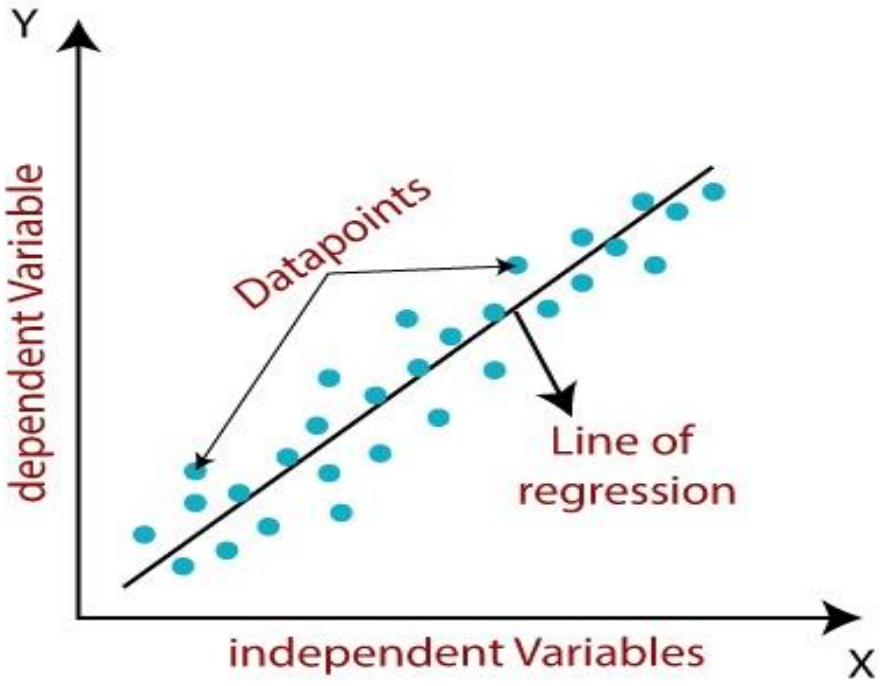


Fig.4.7. Linear Regression

The Linear Regression is used for solving Regression problems whereas Logistic Regression is used for solving the Classification problems.

Linear regression is one of the easiest and most popular Machine Learning algorithms.

It is a statistical method that is used for predictive analysis.

Linear regression makes predictions for continuous/real or numeric variables Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression.

Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

The linear regression model provides a sloped straight line representing the relationship between the variables.

4.6.2 LOGISTIC REGRESSION

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

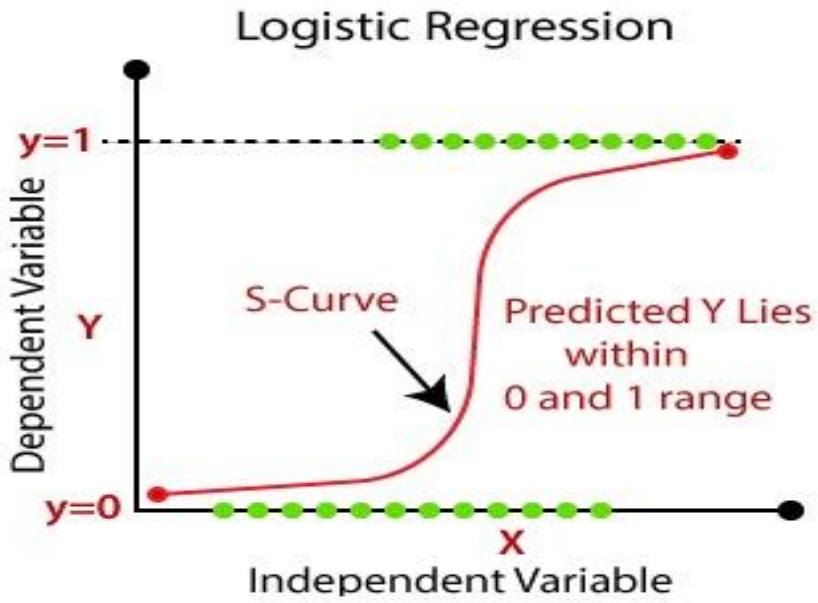


Fig 4.8. Logistic Regression

Steps in Logistic Regression: To implement the Logistic Regression using Python, we will use the same steps as we have done in previous topics of Regression.

Below are the steps:

- Data Pre-processing step
- Fitting Logistic Regression to the Training set
- Predicting the test result
- Test accuracy of the result
- Visualizing the test set result

CHAPTER-5
TECHNOLOGIES

5.1 Machine Learning:

Machine learning is an application of artificial intelligence (AI) that provides system the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves. The process of learning begins with observations or data, such as examples, direct experience, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

Supervised machine learning algorithms can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.

In contrast, **unsupervised machine learning algorithms** are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data.

Semi-supervised machine learning algorithms fall somewhere in between supervised and unsupervised learning, since they use both labeled and unlabeled data for training – typically a small amount of labeled data and a large amount of unlabeled data. The systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised

learning is chosen when the acquired labeled data requires skilled and relevant resources in order to train it / learn from it. Otherwise, acquiring unlabeled data generally doesn't require additional resources.

Reinforcement machine learning algorithms isa learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance.

Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal. Machine learning enables analysis of massive quantities of data. While it generally delivers faster, more accurate results in order to identify profitable opportunities or dangerous risks, it may also require additional time and resources to train it properly. Combining machine learning with AI and cognitive technologies can make it even more effective in processing large volumes of information.

5.2 Python Programming:

Python is a powerful multi-purpose programming language created by Guido Van Rossum.
It has simple easy-to-use syntax, making it the perfect language for someone trying to learn computer programming for the first time. This is a comprehensive guide on how to get started in Python, why you should learn it and how you can learn it.

5.2.1 NumPy

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- A powerful N-dimensional array object
- Sophisticated (broadcasting)functions
- Tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined.
- This allows NumPy to seamlessly and speedily integrate with a wide variety of

databases.

CHAPTER-6
SAMPLE CODE

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
data=pd.read_csv('diabetes.csv')
data.columns
data.shape
data.tail()
data.info()
data.describe()
from numpy import nan
data['Glucose']=data['Glucose'].replace(0,np.nan)
data['BloodPressure']=data['BloodPressure'].replace(0,np.nan)
data['SkinThickness']=data['SkinThickness'].replace(0,np.nan)
data['Insulin']=data['Insulin'].replace(0,np.nan)
data['BMI']=data['BMI'].replace(0,np.nan)
data.head()
print(data.isnull().sum())
data.median()
# missing values with their respective columns median
data.fillna(data.median(), inplace=True)
# Checking if the missing values have been imputed
print(data.isnull().sum())
sns.pairplot(data)
x=data[['Pregnancies','Glucose','BloodPressure','SkinThickness','Insulin','BMI','DiabetesPedi
greeFunction','Age']]
y=data[['Outcome']]
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x_train,y_train)
print(model.coef_)
print(model.intercept_)
predictions=model.predict(x_test)
plt.scatter(y_test,predictions)
plt.figure(figsize=(5,5))
```

```
plt.hist(y_test-predictions)
plt.show()
plt.scatter(y_test,predictions,c='g')
plt.scatter(y_test,y_test-predictions,c='r')
plt.show()
from sklearn import metrics
print(metrics.mean_absolute_error(y_test, predictions))
print(metrics.mean_squared_error(y_test, predictions))
print(np.sqrt(metrics.mean_squared_error(y_test,predictions)))
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
data=pd.read_csv('diabetes.csv')
data.columns
y_data=data['Outcome']
x_data=data.drop('Outcome',axis=1)
x_train,x_test,y_train,y_test=train_test_split(x_data,y_data,test_size=0.3,stratify=y_data,ran
dom_state=5)
model=LogisticRegression()
model.fit(x_train,y_train)
predictions=model.predict(x_train)
train_data_accuracy=accuracy_score(predictions,y_train)
print("accuracy score of the training data:",train_data_accuracy)
```

CHAPTER-7
RESULT SCREENSHOTS

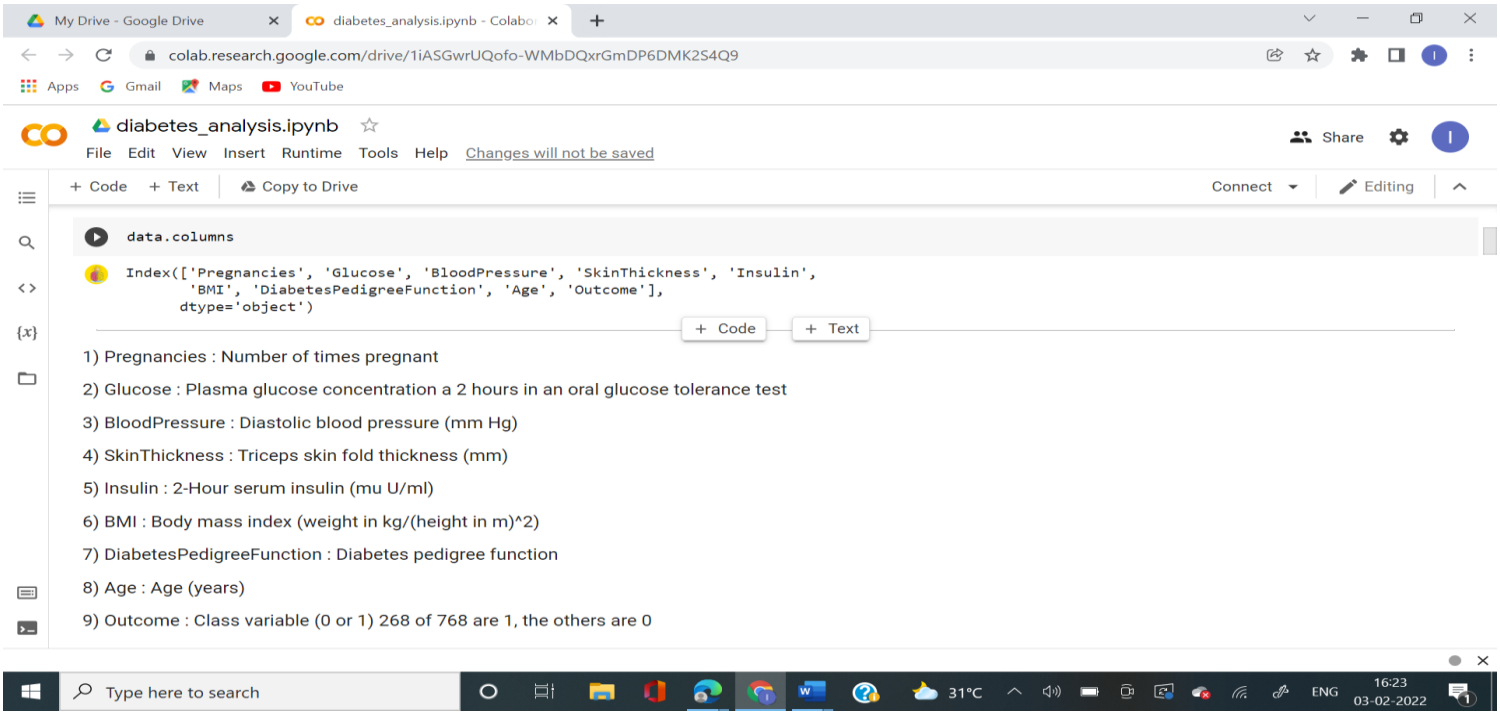
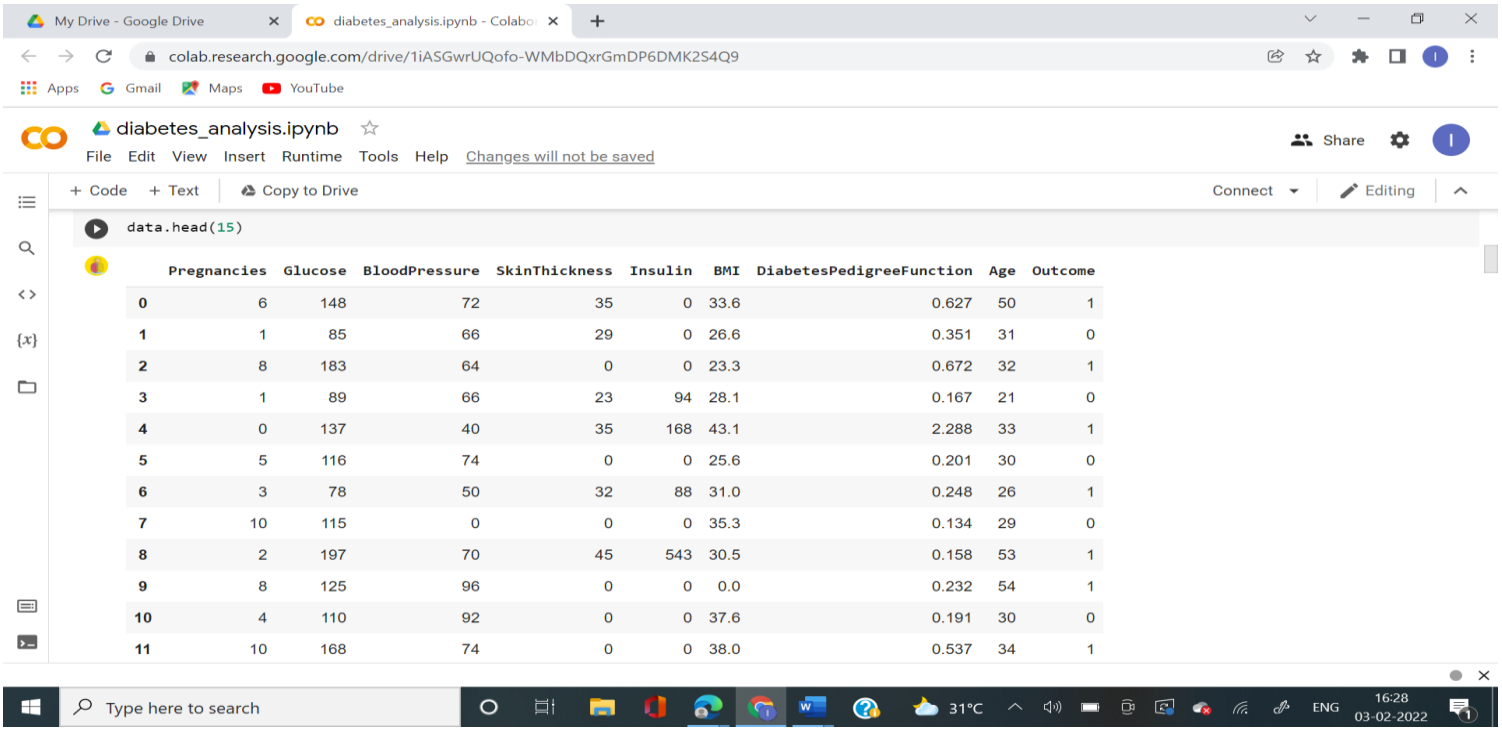
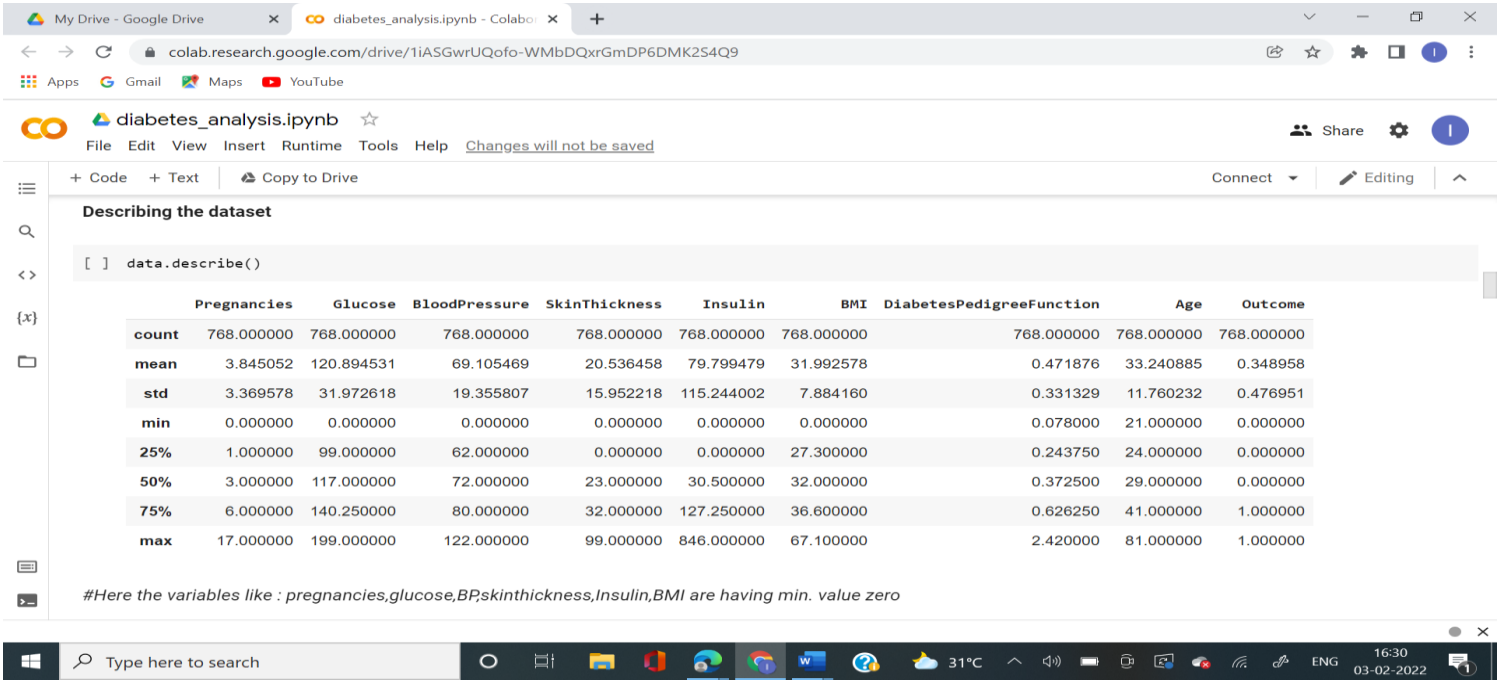


Fig:7.1. Dataset Attributes



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
5	5	116	74	0	0	25.6	0.201	30	0
6	3	78	50	32	88	31.0	0.248	26	1
7	10	115	0	0	0	35.3	0.134	29	0
8	2	197	70	45	543	30.5	0.158	53	1
9	8	125	96	0	0	0.0	0.232	54	1
10	4	110	92	0	0	37.6	0.191	30	0
11	10	168	74	0	0	38.0	0.537	34	1

Fig:7.2. Dataset Attributes and First Fifteen Attributes



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

Fig:7.3. Describing Dataset

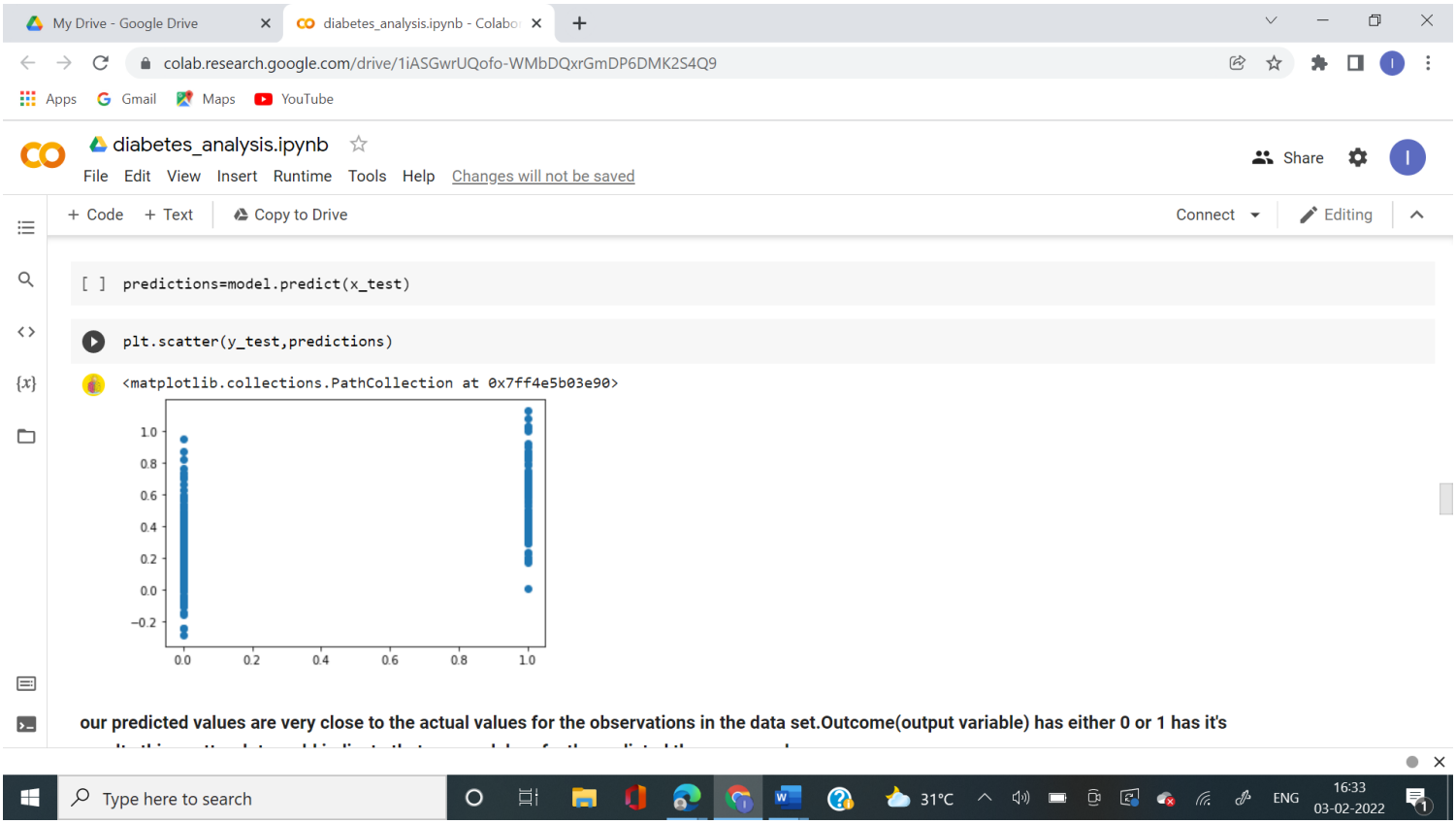


Fig:7.4. Linear Regression Output

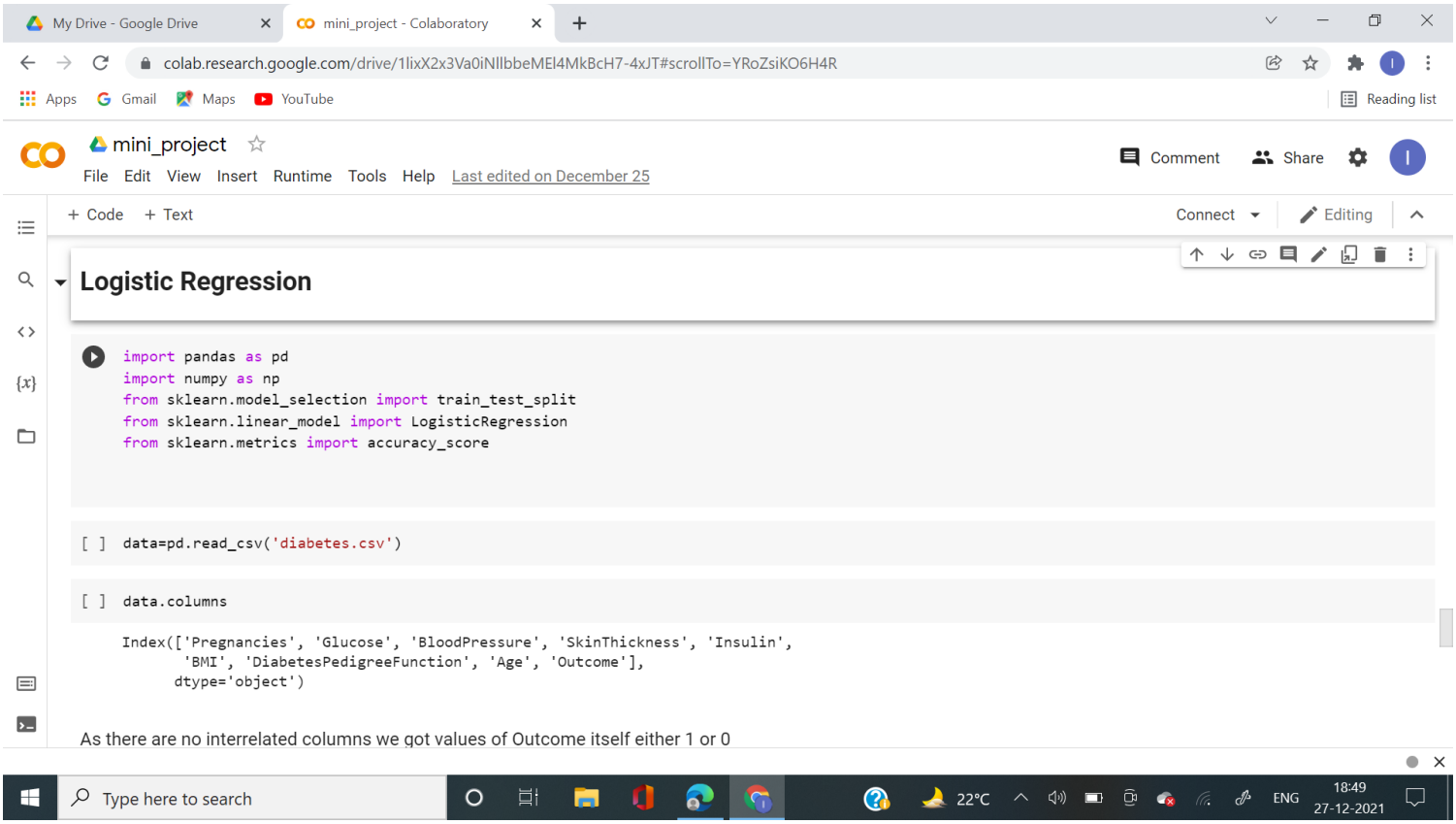


Fig:7.5. Logistic Regression Code

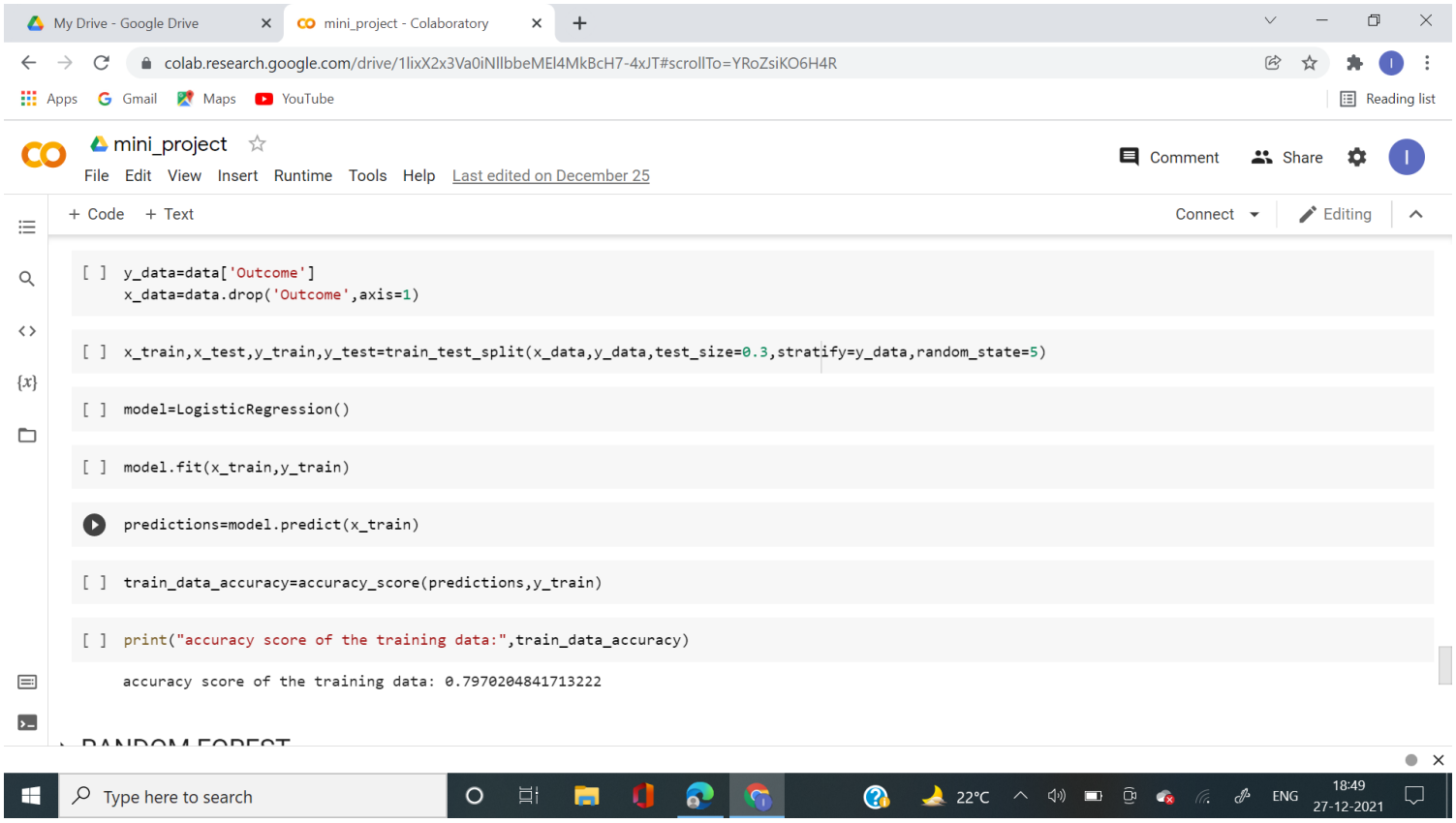


Fig:7.6. Logistic Regression Output

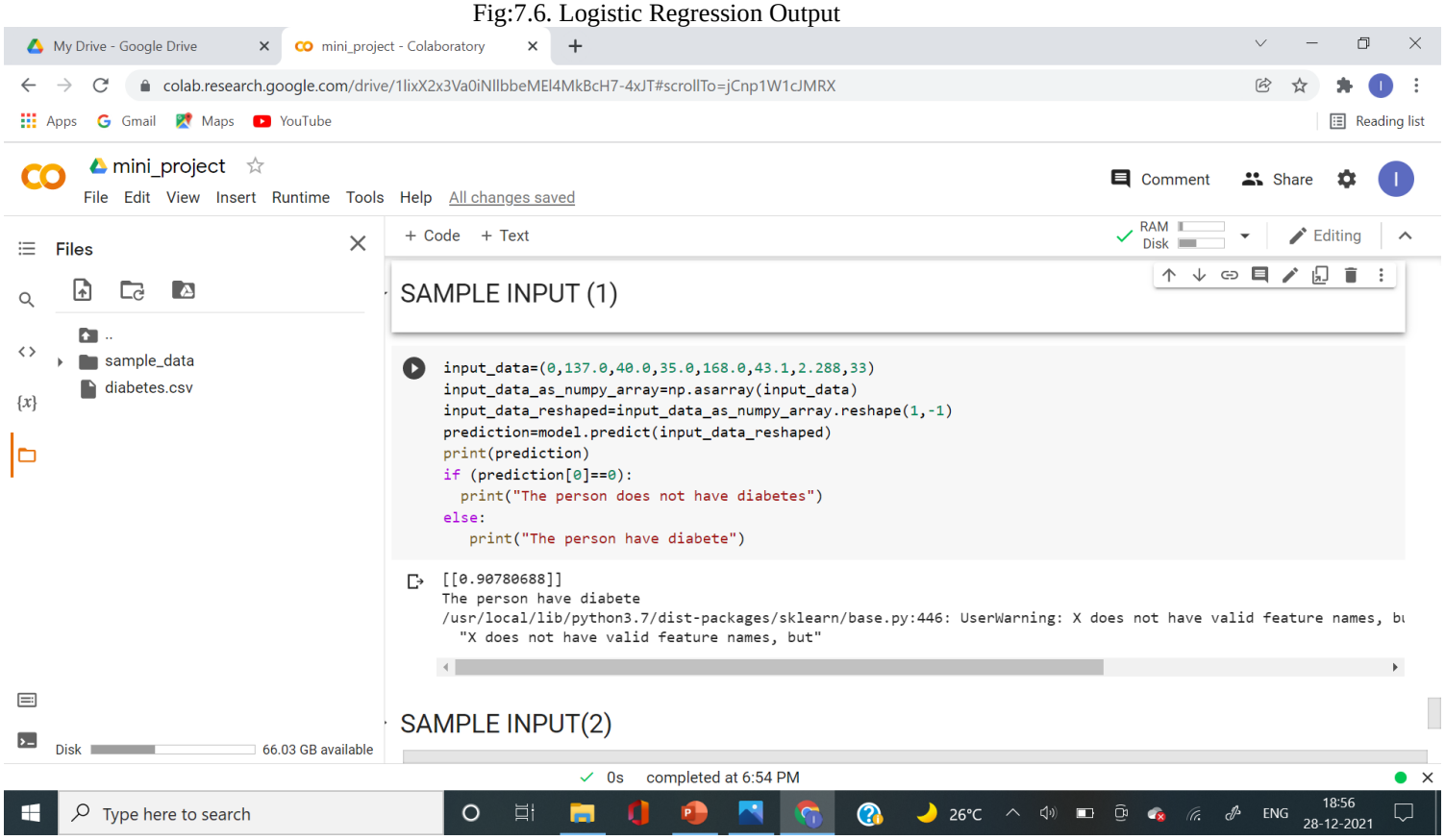


Fig:7.7. Sample Input

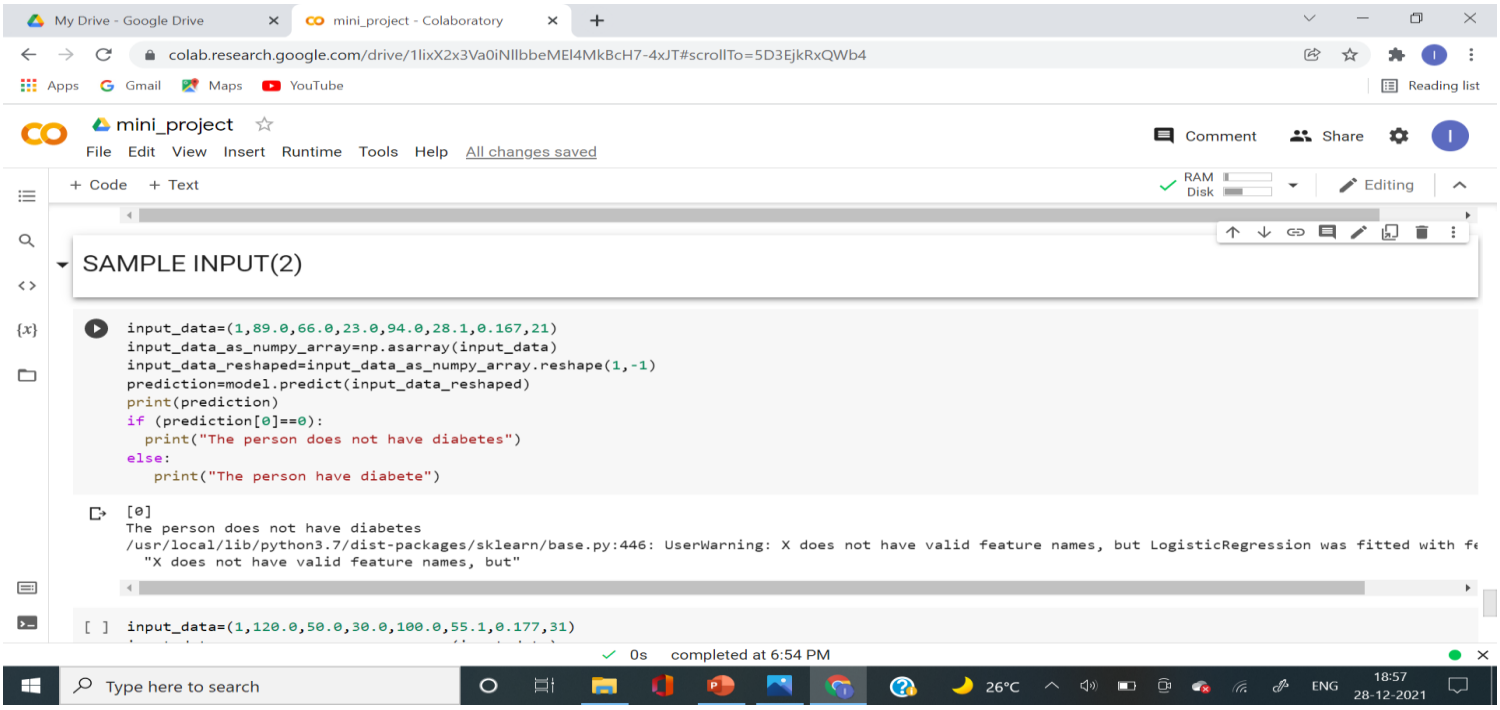


Fig.7.8. Sample Input(2)

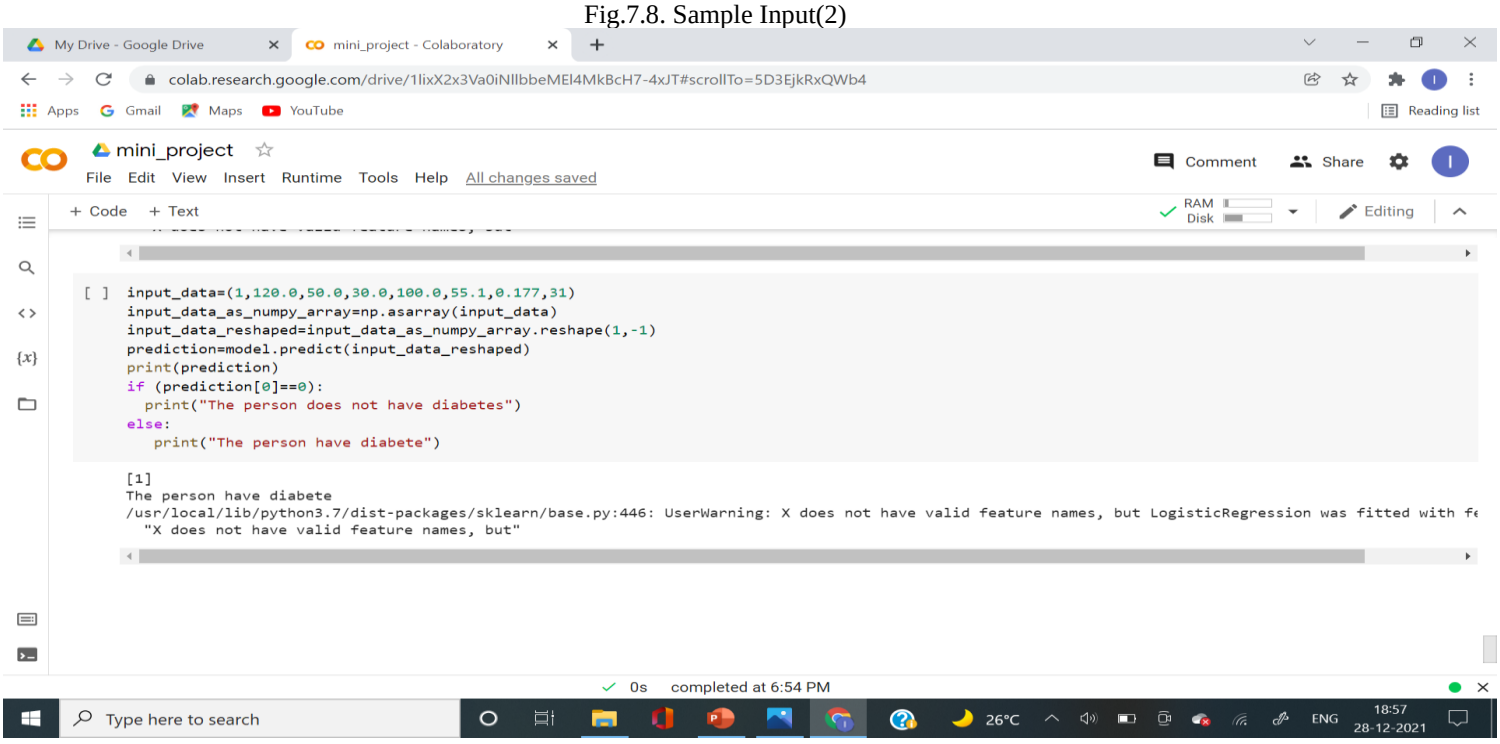


Fig:7.9. Diabetic Prediction

CHAPTER-8
CONCLUSION

This project Disease prediction using machine learning is very much useful in everyone's day to day life and it is mainly more important for the healthcare sector, because they are the one that daily uses these systems to predict the diseases of the patients based on their general information and there symptoms that they are been through.

Now a day's health industry plays major role in curing the diseases of the patients so this is also some kind of help for the health industry to tell the user and also it is useful for the user in case he/she doesn't want to go to the hospital or any other clinics, so just by entering the symptoms and all other useful information the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in just few seconds they can tell the exact and up to some extent the accurate diseases. If health industry adopts this project then the work of the doctors can be reduced and they can easily predict the disease of the patient. The Disease prediction is to provide prediction for the various and generally occurring diseases that when unchecked and sometimes ignored can turns into fatal disease and cause lot of problem to the patient and as well as their family members.

8.1. FUTURE ENHANCEMENT

FUTURE ENHANCEMENT:

Facility for modifying user detail. More interactive user interface. Facilities for Backup creation. Can be done as Web page. Can be done as Mobile Application. More Details and Latest Diseases.

REFERENCES

1. Disease Prediction and Doctor Recommendation System by www.irjet.net
2. Disease Prediction Based on Prior Knowledge by www.hcupus.ahrq.gov/nisoverview.jsp
3. GDPS - General Disease Prediction System by www.irjet.net
4. Disease Prediction Using Machine Learning by International Research Journal of Engineering and Technology (IRJET).
5. Kaveeshwar, S.A., and Cornwall, J., 2014, "The current state of disease mellitus in India". AMJ, 7(1), pp. 45-48.
6. Dean, L., McEntyre, J., 2004, "The Genetic Landscape of Disease [Internet]. Bethesda (MD): National Center for Biotechnology Information (US); Chapter 1, Introduction to Disease. 2004 Jul 7.
7. Machine Learning Methods Used in Disease by www.wikipedia.com
8.https://www.researchgate.net/publication/325116774_disease_prediction_using_machine_learning_techniques
9. https://ieeexplore.ieee.org/document/8819782/disease_prediction
10. Algorithms Details from www.dataspirant.com
11. https://www.youtube.com/disease_prediction
12. https://www.slideshare.com/disease_prediction
[6:19 PM, 2/3/2022] Honey CSE: 13. https:/ en.wikipedia.org /machine_learning_algorithms
14. https://en.wikipedia.org/wiki/Python_(programming_language)
15. https://wiki.python.org/TkInter
16. https://creately.com/lp/uml-diagram-tool/
17. https://app.diagrams.net/