Development of application for bio medical and health data using ML approaches to predict health conditions

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology
In

Computer Science and Engineering/Information Technology
By

Harshit Sinha (191305)

Under the supervision of

(Dr. Tiratha Raj Singh)
&
(Dr.Rajni Mohana)

to

Department of Computer Science & Engineering and Information Technology
Jaypee University of Information Technology Waknaghat,
Solan-173234, Himachal Pradesh

Certificate

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"Development of application for bio medical and health data using ML approaches to predict health condition."** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from July 2022 to May 2023 under the supervision of **(Dr. Tiratha Raj Singh)** (Associate Professor , BT/BI) and co-supervisor **(Dr. Rajni Mohana**) (Associate Professor , CSE).
The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature)

Harshit Sinha, 191305

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature)
Dr. Tiratha Raj Singh
Associate Professor
BT/BI
Dated:

Plagiarism Certificate

As provided by the LRC of JUIT.

Acknowledgement

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing that made it possible to complete the project work successfully.

I am really grateful and wish my profound indebtedness to Supervisor **Dr. Tiratha Raj Singh**, **(Associate Professor)**, Department of BT/BI Jaypee University of Information Technology, Wakhnaghat. Deep Knowledge & keen interest of my supervisor in the field of Machine Learning and Neural Networks carrying out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project. I would like to express my heartiest gratitude to Dr. Tiratha Raj Singh, Department of BT/BI, for his kind help to finish my project.

I would also like to show my gratitude to each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

Harshit Sinha(191305)

Table of Content

List of Abbreviations

S. No	Title	Page No.
1	ML: Machine Learning	12
2	CKD: Chronic Kidney disease	16
3	SVM: Support Vector Machine	32
4	KNN: K- Nearest Neighbor	36

List of Figures

LIST OF GRAPH

List of Table

Abstract

Modern people suffer from various diseases due to their environment and lifestyle. This disease claims the lives of many people. As a result, predicting disease early becomes an important task. Many researchers have developed new techniques to predict disease early, before it's too late. This benefits both medicine and people. However, doctors find it difficult to make accurate predictions based on symptoms. The most difficult challenge is accurately predicting disease. Machine learning will play a key role in overcoming this problem by predicting disease. Medicine generates an enormous amount of data each year. Proper analysis of medical data benefits early patient care due to the increase in data in the medical and healthcare field. Machine learning algorithms use disease data to uncover pattern information hidden in vast amounts of medical data.

We have developed a broad range of disease predictions based on patient symptoms. We use random forest classifiers and convolutional neural network (CNN) machine learning techniques for effective disease prediction. Disease prediction requires collection of disease symptoms. A dataset contains a set of instances with different attributes used to train a predictive model used in a web application for prediction. The main goal of this project is the development and application of effective disease prediction models.

Chapter -1
1.1 Introduction

In ML (Machine Learning), a subset of artificial intelligence, computers are instructed to learn information on their own without the assistance of a person.. Computational statistics are used to build basic machine learning algorithms. Data is fed to a computer and the computer "learns" from it. Data “teaches” computers, revealing their complex patterns and underlying algorithms, bringing fresh data knowledge, new insights, and the potential for new discoveries.

Machine learning is increasingly being used in healthcare to help patients and physicians overcome industrial challenges and establish more integrated systems to streamline workflows. There are many notable examples of the use of machine learning and health principles in science and medicine. Deep learning in radiology automatically identifies complex patterns based on insights a radiologist gains from traditionally evaluating images such as his X-rays, CTs, MRIs, PET scans and radiology reports. It helps you make intelligent decisions. Automated detection and diagnosis systems based on machine learning have been proven to work as well as experienced radiologists. Google's healthcare machine learning application has been trained to detect breast cancer and achieved 89% accuracy. This is on par with or better than radiologists.These are just a handful of the numerous applications of machine learning in healthcare.

My App is a machine learning based web app built on the React framework that can predict various diseases. A number of datasets were used to train this model. The trained model takes as input patient-provided attributes related to a particular disease and accurately predicts the patient's health status. This website application seeks to put into practise a machine learning model that effectively forecasts human sickness from human symptoms. Let's see how to approach this machine learning problem.

- **Gathering the Data :** Preparing the data is the initial step in any machine learning problem. We'll be utilizing a Kaggle dataset for this problem .

- **Cleaning the Data :** The most crucial phase of a machine learning project is cleaning. The quality of our machine learning model is determined on the quality of our data.

- **Model Building :** Once the data has been gathered and cleaned, it can be utilized to train a machine learning model. These cleaned data will be used to train the CNN and Random Forest Classifiers.

- **Inference :** We anticipate the patient's health using the attributes they entered on the app's form after training our five models.

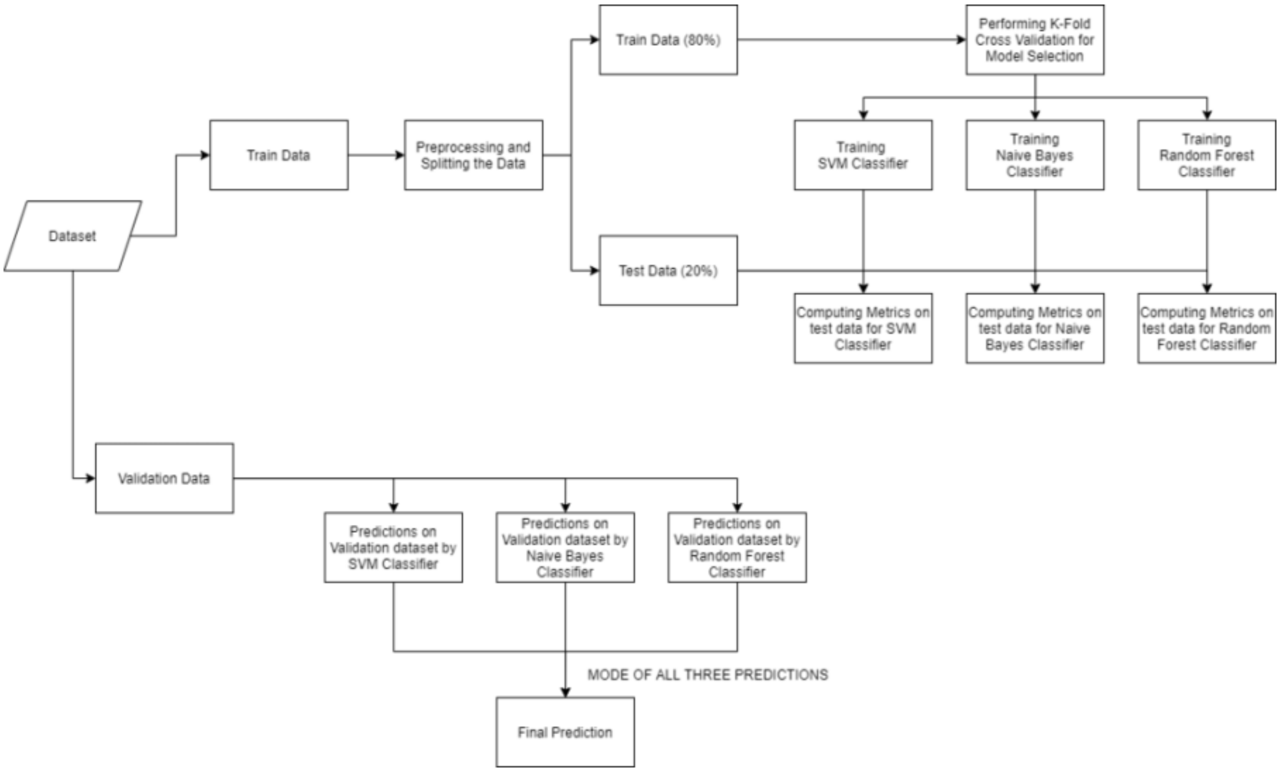

Dataset
Train Data
Preprocessing and Splitting the Data
Train Data (80%)
Test Data (20%)
Performing K-Fold Cross Validation for Model Selection
Training SVM Classifier
Training Naive Bayes Classifier
Training Random Forest Classifier
Computing Metrics on test data for SVM Classifier
Computing Metrics on test data for Naive Bayes Classifier
Computing Metrics on test data for Random Forest Classifier
Validation Data
Predictions on Validation dataset by SVM Classifier
Predictions on Validation dataset by Naive Bayes Classifier
Predictions on Validation dataset by Random Forest Classifier
MODE OF ALL THREE PREDICTIONS
Final Prediction

fig. 1 show how the model works

Our web application uses a total of 5 machine learning models, each of which forecasts a distinct ailment based on specific input variables. The illnesses that are preventable include :

- **Kidney Disease :** Conditions that harm the kidneys and lessen their capacity to keep the body healthy by filtering waste materials from the blood are referred to as kidney diseases. Two-thirds of kidney disease cases are brought on by either diabetes or hypertension.

- **Heart Disease :** Various heart problems are referred to as heart diseases. Symptoms of coronary artery disease may differ between men and women. For example, men are more likely to experience chest pain. In addition to chest discomfort, women are more likely to exhibit other signs and symptoms such as shortness of breath, nausea, and extreme fatigue.

- **Diabetes :** Diabetes, also referred to as diabetes mellitus, is a metabolic condition that causes excessive blood sugar.. The hormone insulin transports sugar from the blood to the cells where it is stored or used as energy.In diabetes, the body either does not produce enough insulin or cannot effectively use the insulin that is produced.

  High blood sugar from untreated diabetes can damage nerves, eyes, kidneys, and other organs.

- **Liver Diseases :** The human body liver is the second largest organ (after the skin). It is located just below the ribcage on the right side and is about the size of a soccer ball. The liver separates nutrients and waste products as they move through the digestive system.

  Some types of liver disease (including non-alcoholic fatty liver disease) cause few symptoms. In other conditions, the most common symptom is jaundice. This is a yellowing of the skin and the whites of the eyes. Jaundice occurs when the liver cannot remove a substance called bilirubin.

- **Malaria :** This illness is brought on by parasites. Through the bite of an infected mosquito, the parasite is transferred to people. Malaria typically causes severe illness, including high fever and chills..

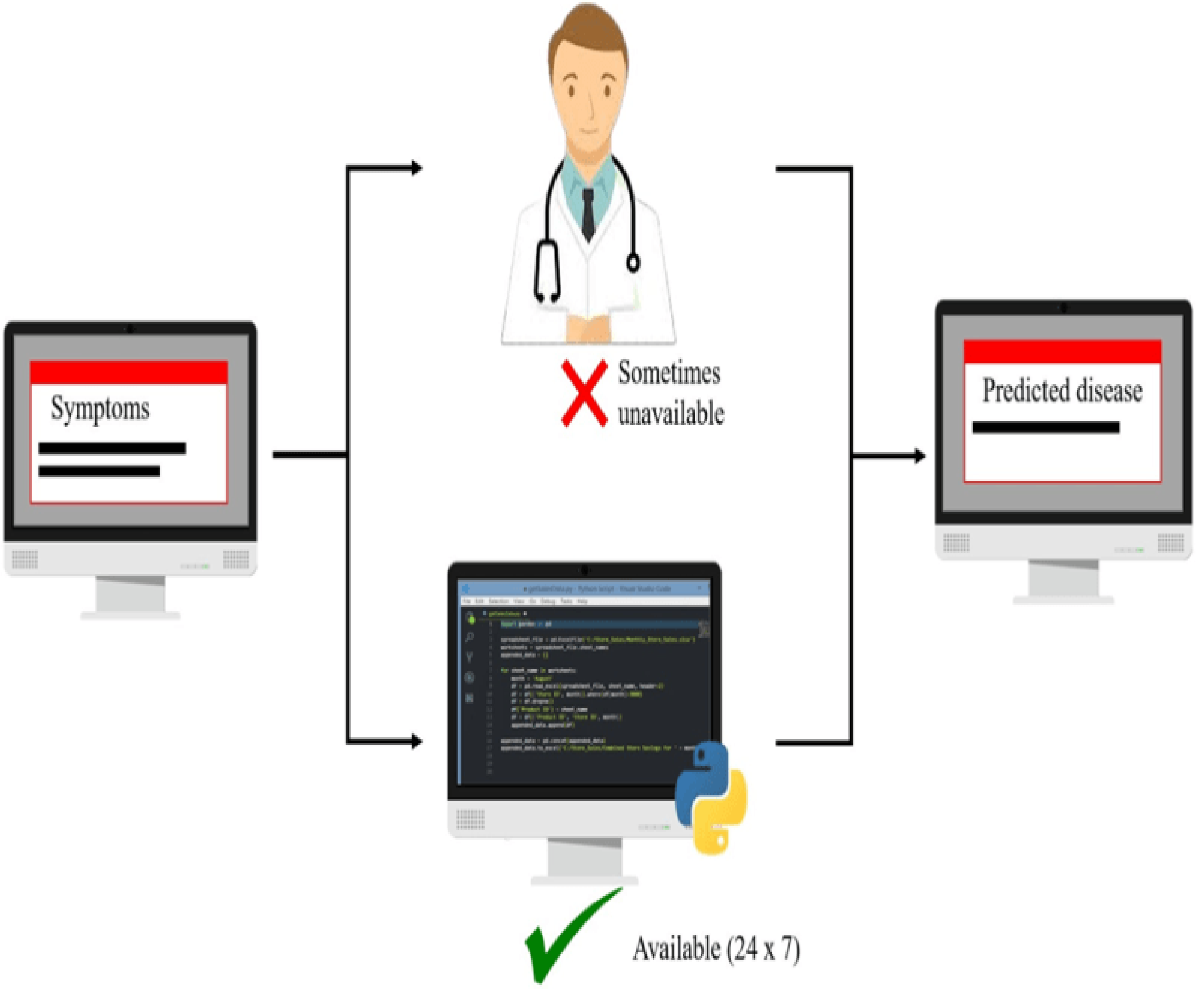


fig.2 Outline of the app how it works

1.2 Problem Statement

In this virtual world, doctors are doing everything possible to save their life, even if they have to risk themselves. There are also remote villages without medical facilities. A virtual doctor is a certified doctor who chooses to see a doctor online via video or phone appointment rather than an in-person appointment, but in an emergency this is not possible. This is because machines can perform tasks more efficiently and humans can do so with consistent accuracy without human error. A disease predictor is a type of virtual physician who can accurately diagnose any patient without the involvement of a person.

Even with diseases like covid-19 and Ebola, a disease predictor that can detect someone's disease without physical contact could be a boon. I have some models of virtual doctors, but they don't take into account all the parameters I need, so they don't have the accuracy I need.

1.3 Objectives

- Aim of this project is to use medical data as a useful resource, identify trends, and ultimately provide a user-friendly interface to access results obtained through training on the data so that the general public can make it easily accessible.

- In the event of an emergency, doctors and other medical personnel are constantly needed. Our prediction algorithm can be useful and employed in the diagnosis of a disease under the COVID-19 scenario, where adequate facilities and resources are inadequate.

fig3. shows the different category

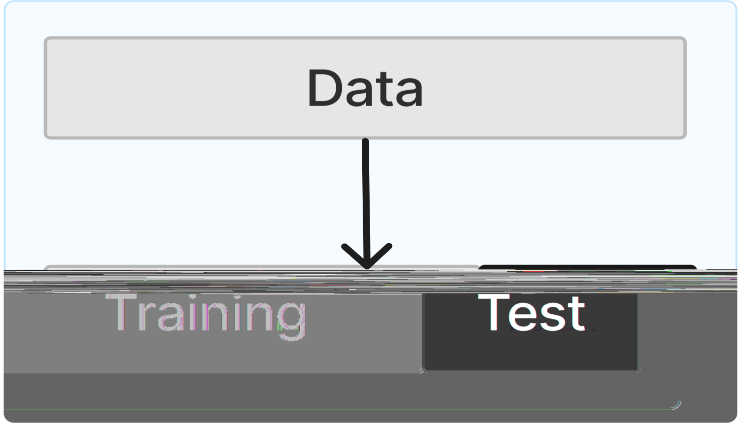
1.4 Methodology

- **Database :** A database is information set up so that it can be easily accessed, managed, and updated. Computer databases typically store collections of records or files containing information such as sales transactions, customer data, financial data, product information, and so on.

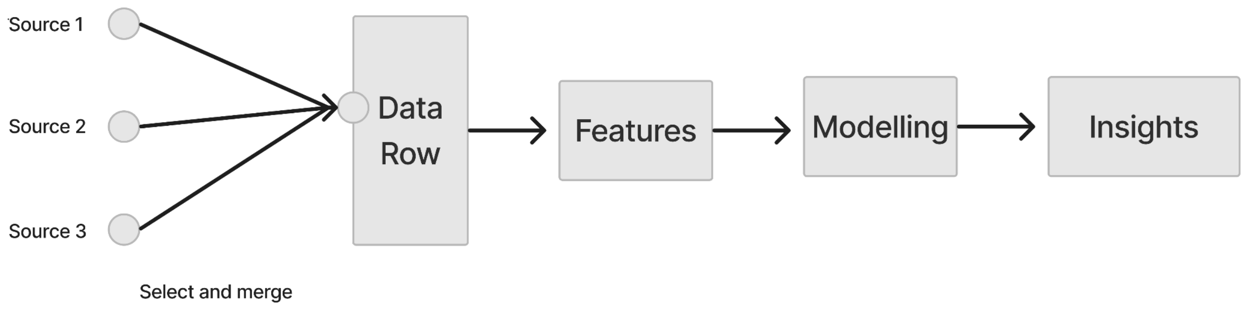
  Databases are used to store, maintain, and access data. Collect information about people, places, and things. This information is collected in one place for observation and analysis. A database can be viewed as an organized collection of information.

- **Pre Processing :** Data preprocessing refers to the procedures we must follow to alter or encode data so that a machine can quickly and readily decode it.

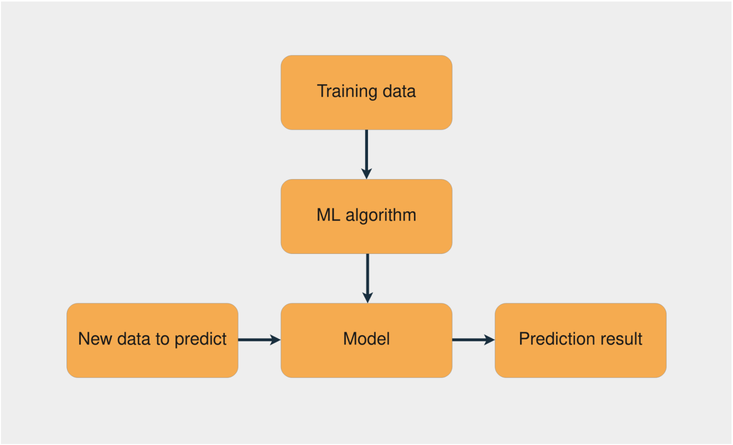
  The algorithm's ability to quickly analyze the properties of the data is essential for a model to be accurate and exact in its predictions.



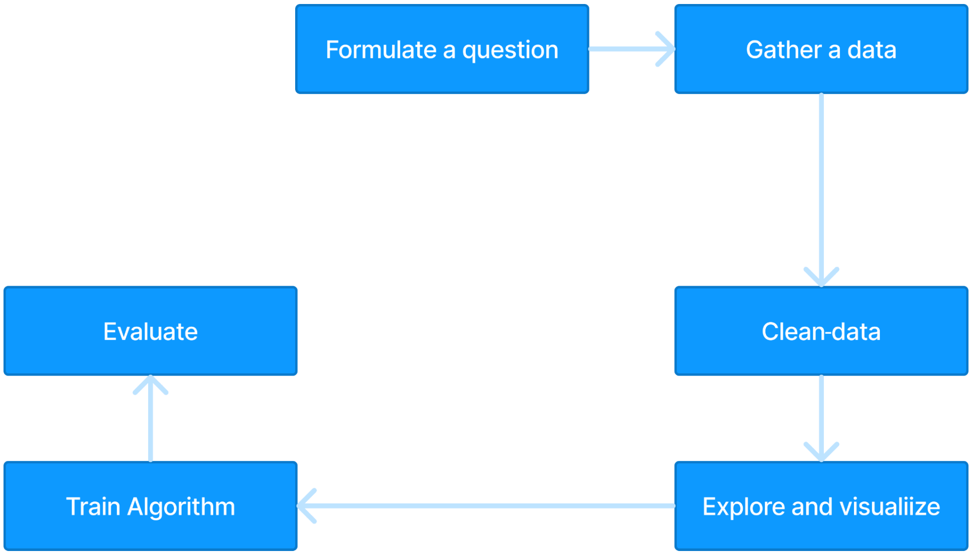
- **Feature Engineering :** It's a machine learning's preprocessing step, which extracts features from unprocessed data. It aids in better communicating a fundamental issue to predictive models, increasing the model's accuracy for unobserved data. The feature engineering method chooses the most practical predictor variables for the model, which is composed of predictor variables and an outcome variable.



- **Model Selection :** Selecting a statistical model based on input data from a list of viable models. In the simplest scenarios, an existing set of data is taken into account. However, the work could also entail planning trials so that the information gathered is useful for the challenge of model choice.

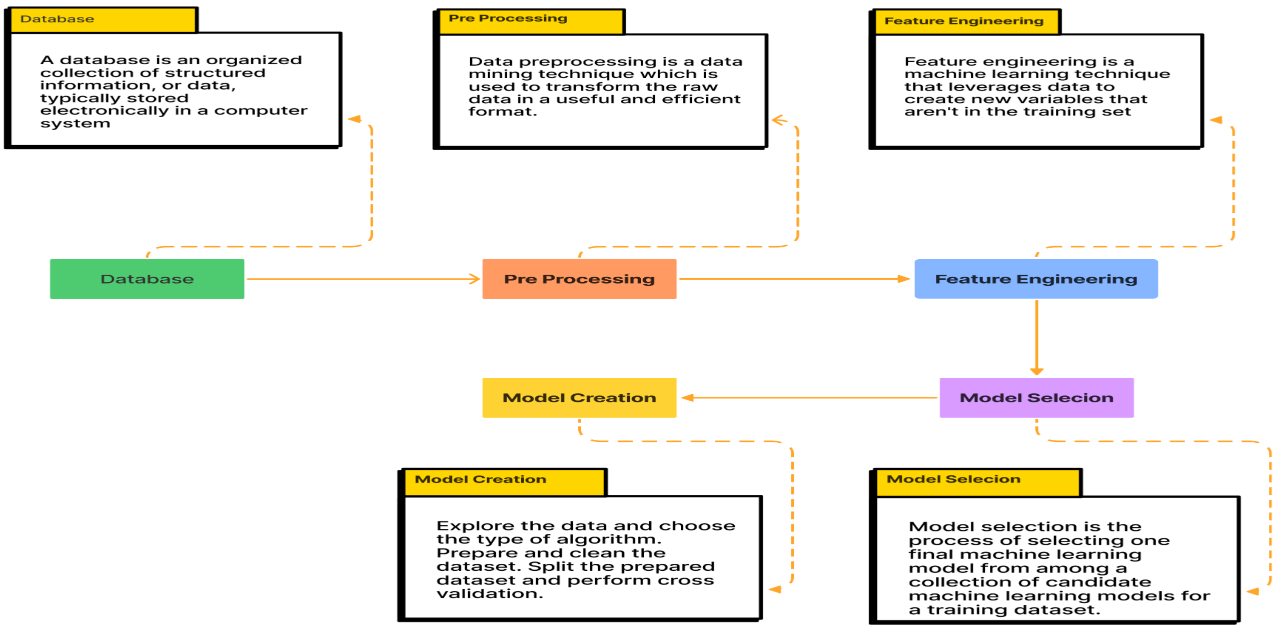


- **Model Creation :** In machine learning, an illustration of model creation A machine learning model is created by using the knowledge it has gained through training data, generalizing it, and then using it to make predictions and accomplish its goal on brand-new data that it has never seen before. The model can't be built without data, and having access to data is insufficient



And here is the figure which shows how they all works :-

fig 4. Methodology how it function step wise



A database is an organized collection of structured information, or data, typically stored electronically in a computer system

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.

Feature engineering is a machine learning technique that leverages data to create new variables that aren't in the training set

Explore the data and choose the type of algorithm. Prepare and clean the dataset. Split the prepared dataset and perform cross validation.

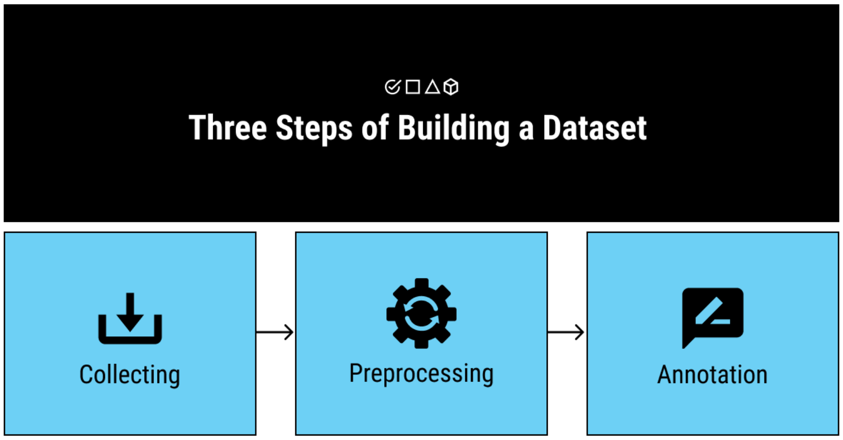
Model selection is the process of selecting one final machine learning model from among a collection of candidate machine learning models for a training dataset.

1.5 Organization

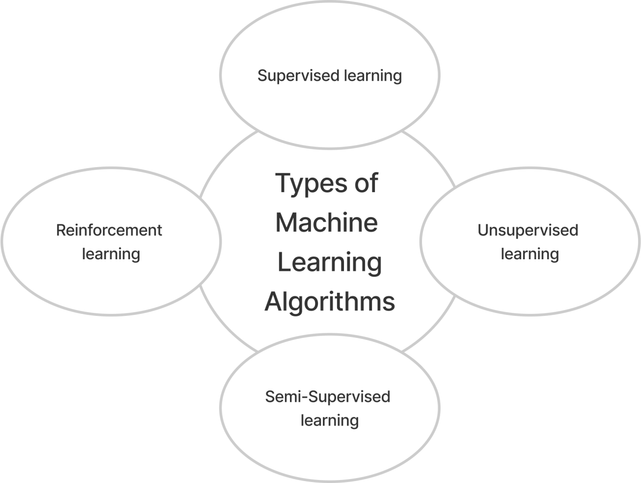
We have organized our project strategy using following steps:

❖ Understanding the term related to the topic :

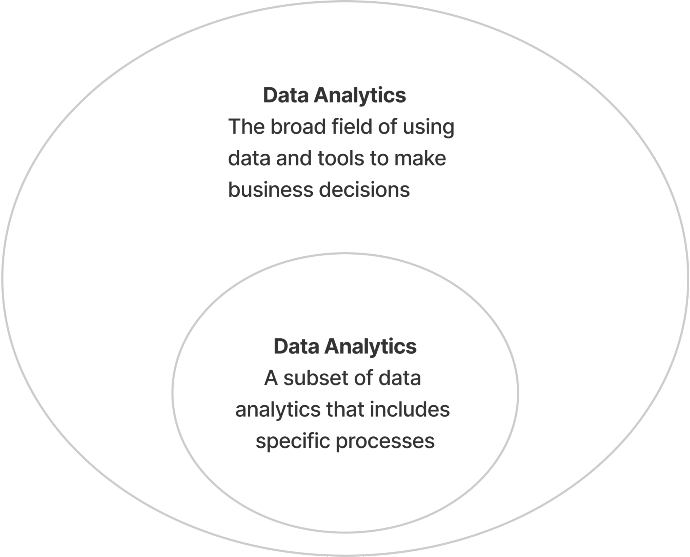
- **Dataset :** A file containing one or more records is referred to as a record. The fundamental informational unit utilized by z/OS programmes is the record. A record is any named collection of records.



- **Algorithms :**Machine learning algorithms enable programmes to foresee outcomes, find hidden patterns in data, and improve performance based on past results. For example, basic linear regression for prediction issues like stock market forecasting and the KNN algorithm for classification jobs are just a few examples of the many algorithms that may be employed in machine learning to complete a variety of tasks for classification problems.

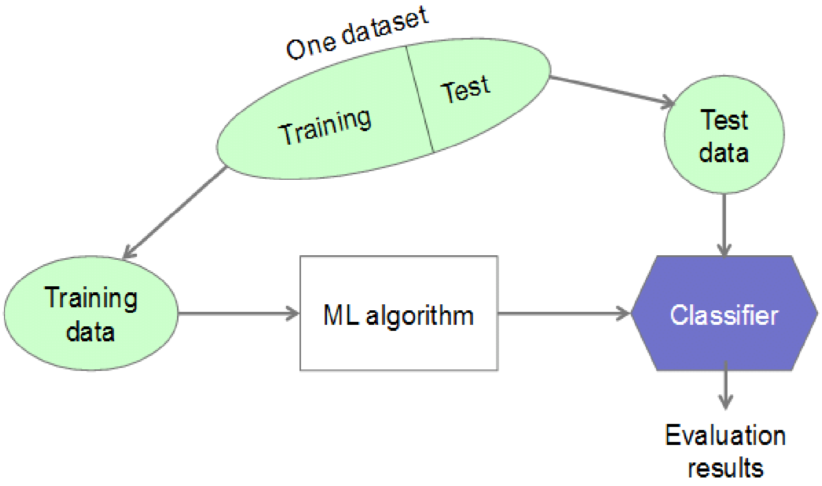


- **Analysis :** Data analysis to find relevant information, support reasoning, and support decision-making is the process of analyzing, cleaning, manipulating, and modeling data. Data analysis is used in several fields of business, science and social sciences and has many dimensions and methods. It contains various techniques and has various names. Data analytics contribute to more scientific decision-making and more efficient business operations in the modern business world.

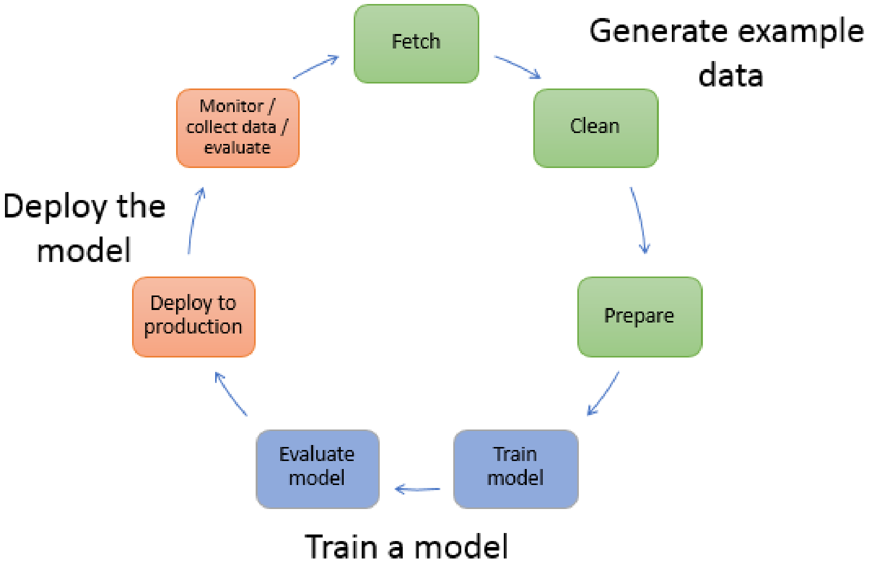

Data Analytics
The broad field of using data and tools to make business decisions
Data Analytics
A subset of data analytics that includes specific processes

- **Training and Testing model :** Simply put, training a model involves learning appropriate values for each of the weight and bias from labeled samples. Risk minimization is the process by which a machine learning algorithm builds a supervised learning model by looking at a large number of examples and trying to identify the model that minimizes the loss.

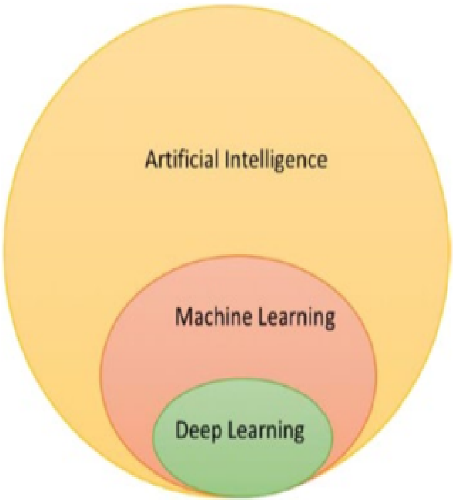
  Model testing in machine learning is the process of assessing a fully trained model's performance on a testing set.

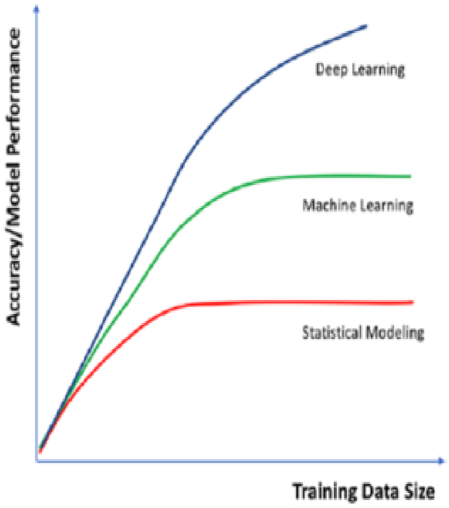


- **Model Creation :** A machine learning model is created by taking the knowledge it has gained from training data, generalizing it, and then using it to make predictions and achieve its goal. You can't develop the model if there isn't enough data, and having access to data is insufficient.



- **Accuracy :** Machine learning models are used by businesses to make defensible business decisions, and improved model results lead to better judgments. The cost of errors can be extremely high, however this cost is reduced by improving model accuracy. There is, of course, a threshold of diminishing returns at which the value of creating a more accurate model will not lead to a comparable profit rise, but frequently it is useful everywhere. For instance, a false-positive cancer test costs the hospital and the patient money a lot. Enhancing model accuracy has advantages that save time, money, and unnecessary worry.





1. **Step - 1 :** Gather data on a variety of disorders that can be predicted in order to forecast health.

2. **Step - 2 :** Perform exploratory data analysis for procured datasets such as data cleaning, removing irrelevant fields etc.

3. **Step - 3 :** Observe the correlations between the data fields in case of csv files and in case of malaria cell images, the images were converted to matrices to train the model.

4. **Step - 4 :** All the datasets were trained and fitted into models.

5. **Step - 5 :** Backend to load and deploy these models was created.

6. **Step - 6 :** Frontend to Post values and get predictions from the Backend was created.

Chapter-2
LITERATURE SURVEY

This project uses various research papers from IEEE, Cisco, Changsha University, Google .

Talking about each one in detail:

Designing Disease Prediction Model Using Machine Learning Approach - D. Dahiwade, G. Patle and E. Meshram by IEEE format :

In this early prediction of disease is an important issue. However, it can be exceedingly challenging for doctors to make a precise forecast based on symptoms. The most challenging challenge is to accurately forecast sickness. Data mining is crucial in disease prediction in order to solve this issue. Every year, data in medicine grows considerably. Accurate medical data analysis helps with early patient treatment as the amount of data in the medical and healthcare fields grows. To uncover hidden patterns in massive amounts of medical data, data mining employs illness data. Based on the symptoms of the patient, we developed a general disease prognosis. Convolutional neural networks (CNN) and K-Nearest Neighbor (KNN) machine learning techniques are used to precisely forecast disease. Disease prediction requires a data set of disease symptoms. This common disease prediction takes into account the person's lifestyle and laboratory information for an accurate prediction. The CNN algorithm outperforms the ANN algorithm in terms of general disease prediction accuracy with an accuracy of 84.5%. Additionally, his KNN is more efficient in terms of time and storage than CNN. After predicting common diseases, the system can indicate whether the risk of common diseases is low or high, and the risks associated with common diseases.

Application of Machine Learning in Disease Prediction - by P. S. Kohli and S. Arora

In this the application of machine learning in the field of medical diagnostics is gradually increasing. Most notably, this will help improve the classification and detection systems used to diagnose diseases, providing data that can help medical professionals detect deadly diseases early and improve patient survival. rate is greatly improved. In this article, we apply different classification algorithms, each with their own advantages, to three separate disease databases (heart, breast cancer, and diabetes) available in the UCI Disease Prediction Repository. Feature selection for each dataset was achieved by backward modeling using p-value tests. The results of this study support the idea of using machine learning for early disease detection.

Prediction of Heart Disease Using Machine Learning by A. Gavhane, G. Kokkula, I. Pandya and K. Devadkar :

With heart attack rates skyrocketing among adolescents, systems need to be put in place to detect and prevent heart attack symptoms early. Because it is impractical for ordinary people to undergo expensive tests such as electrocardiograms frequently, there is a need for a practical and reliable system for predicting possible heart disease. Therefore, we propose to develop an application that can predict heart disease susceptibility based on basic symptoms such as age, gender, and pulse rate. Neural network machine learning algorithms are used in the proposed system as they have been proven to be the most accurate and reliable algorithms.

A Survey on machine learning techniques for the diagnosis of liver disease by - G. Shaheamlung, H. Kaur and M. Kaur :

As the number of people suffering from liver disease is rapidly increasing due to excessive alcohol consumption, inhalation of pollutant gasses, drugs, food contamination, and packaging of food pickles, the medical professional system will help doctors auto-predict. Repeated developments in machine learning techniques will enable early prediction of liver disease, making it easier for people to diagnose deadly diseases early. This is more useful in the health sector, and can also use the medical expert system in remote areas. The liver plays a very important role in life helping to remove toxins from the body.Early prediction is therefore very important for disease diagnosis and recovery. Different types of machine learning, supervised, unsupervised, and semi-supervised reinforcement learning for liver disease diagnosis, such as SVM, ANN, K-mean clustering, neural networks, decision trees, etc. and sensitivity. The motivation for this work is to provide an overview and comparative analysis across machine learning methods for diagnosis and prediction of liver disease in the medical field. These have already been used by various authors to predict liver disease and build on them. About analysis of accuracy, sensitivity, precision, and specificity

Prediction of diseases using random forest classification algorithm by - R.D.H.D.P. Sreevalli, K.P.M. :

Modern people face various diseases due to environmental conditions and lifestyle habits. Predicting disease early is becoming very difficult. Accurately predicting symptoms becomes difficult for physicians. Therefore, accurate prediction of disease becomes the most difficult task. Data mining plays an important role in solving this problem. With increasing data in the medical and healthcare sector, accurate analysis of medical data benefits from early patient care.finds hidden pattern information from disease data in a huge amount of medical data by data mining. We proposed a global disease prediction based on patient symptoms. For disease prediction, we use the Random Forest machine learning algorithm for accurate disease prediction. Disease symptom record necessary for disease prediction .This common disease prediction takes into account a person's lifestyle and laboratory information to make an accurate prediction. The overall disease prediction accuracy using the random forest algorithm is more accurate and efficient. Experimental results have proven that this has higher time and memory requirements as well. After predicting common diseases, the system can show common disease-related risks. General disease risk is lower general disease risk or higher risk.

Classification and Prediction of diabetes disease using machine learning paradigm by - M. Maniruzzaman, M.J. Rahman, B. Ahammed, M.M. Abedin :

A chronic condition known as diabetes is characterized by elevated blood sugar. It may result in a variety of different diseases, including heart attack, kidney failure, and stroke. In 2014, the global prevalence of diabetes was estimated to be 422 million. In 2040, the number will reach 642 million. A machine learning (ML)-based system for anticipating diabetes patients is the major goal of this project.

Author(s)	**Journal/ conference, year**	**Published By**	**Methodology**	**Disadvantage**
Ketut Agung Enriko Muhammad Suryanegara Dinda Agnes Gunawan	2016	ICCMC	KNN	Not Optimal
Chandrasekhar Rao Jetti Rehamatulla Shaik Sadhik Shaik	2021	International Journal of Science and Healthcare Research	Naive bayes	Not Optimal
Shahadat Uddin , Arif Khan, Md Ekramul Hossain and Mohammad Ali Moni	2019	BMC Medical Informatics and Decision Making	Random forest	Optimal
D. Dahiwade, G. Patle and E. Meshram	2019	ICCMC	CNN	Optimal
A. Gavhane, G. Kokkula, I. Pandya and K. Devadkar	2018	ICECA	SVM, ANN	Not OPtimal

M. Maniruzzaman, M.J. Rahman, B. Ahammed, M.M. Abedin	2020	IEEE	Deep learning	Optimal

Chapter-3
SYSTEM DEVELOPMENT

3.1 Date Set Used in the Major Project

The machine learning models were built using a total of five datasets:

- **Chronic Kidney Disease dataset :** Data was collected in India over a two-month period using 25 characteristics (eg, RBC count, WBC count, etc). The categorization, which can be either "ckd" or "not ckd" (ckd stands for chronic kidney disease), is the aim. 400 rows are present.

  The data has to be cleaned since it contains NaNs and needs to have the numeric characteristics made to float. Basically, we were told to delete any row containing even one NaN, which translates to ALL ROWS with Nans, with no threshold.

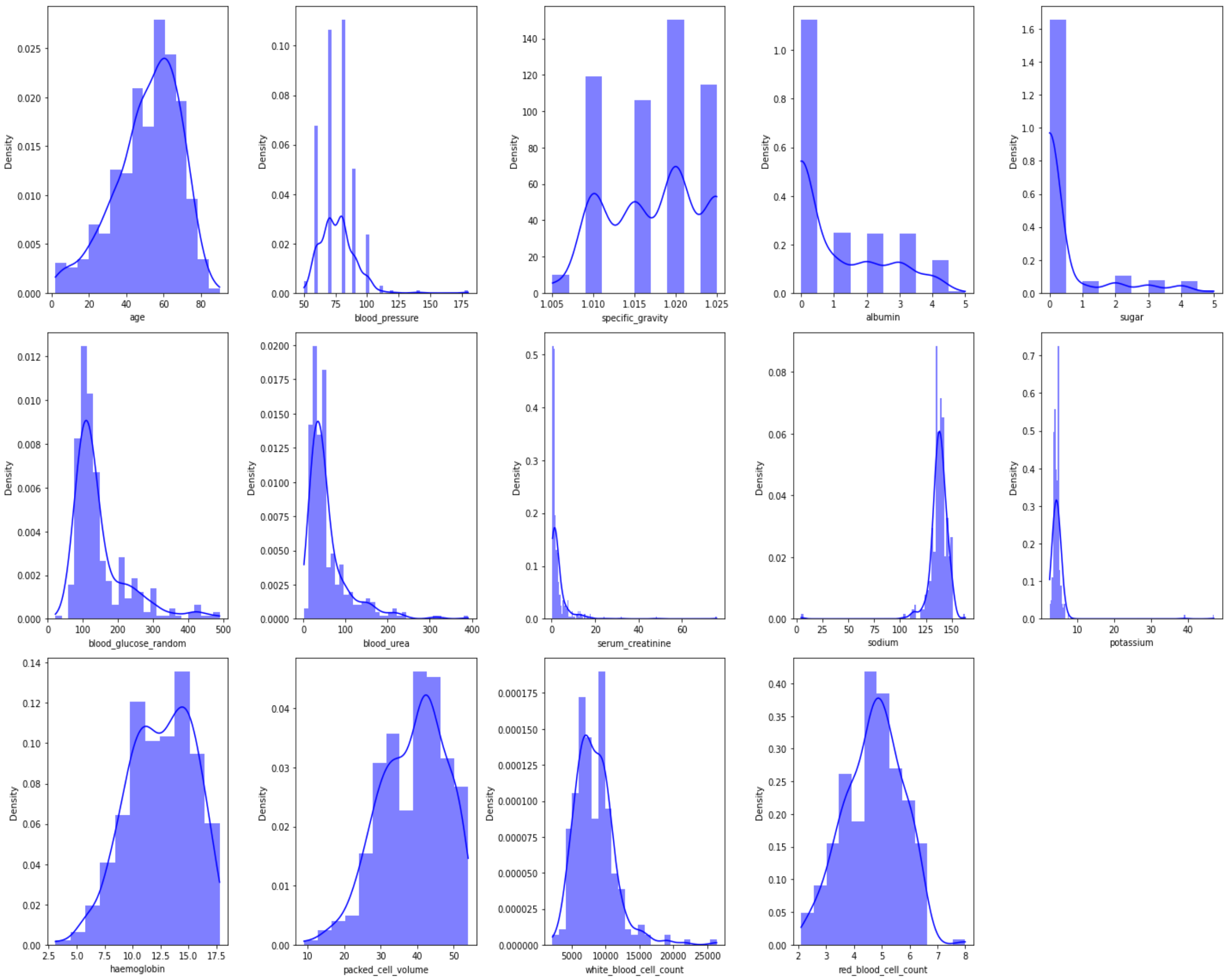


fig5. shows the dataset graph of kidney disease

- **Pima Indians Diabetes Database :** The Institute of Diabetes and Digestive and Kidney Diseases is the original way of this dataset. Based on certain diagnostic metrics present in the dataset, the dataset's goal is to diagnostically check whether a patient has diabetes or not. These cases were chosen from a bigger database under a number of restrictions. Particularly, all patients at this facility are Pima Indian women who are at least 21 years old.

  The datasets include one goal variable, Outcome, together with a number of medical predictor factors. The patient's BMI, insulin level, age, number of previous pregnancies, and other factors are predictor variables.

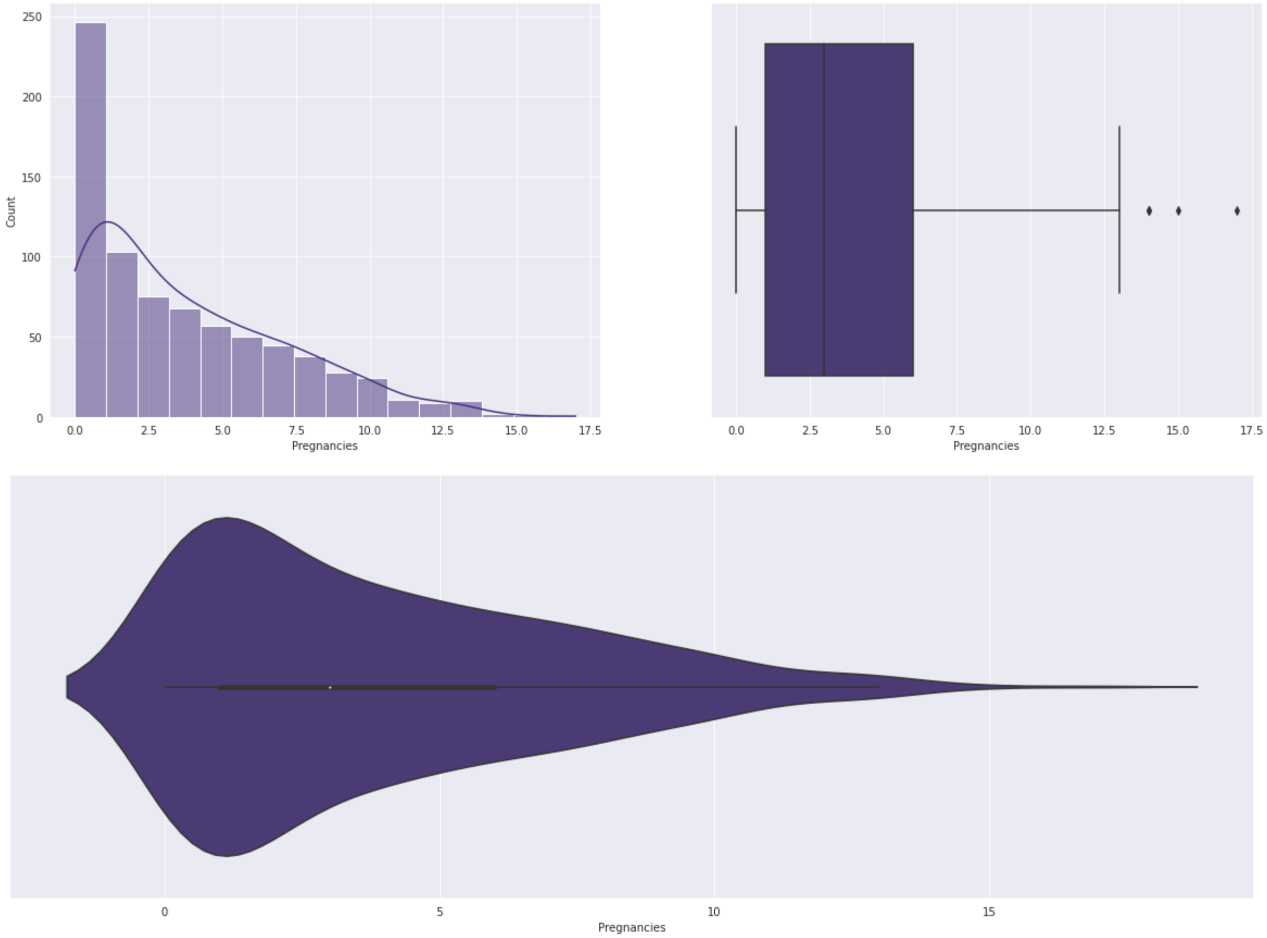


fig6. shows the graph of diabetes database

- **Indian Liver Patient Records :**The number of patients with liver disease has been steadily rising as a result of excessive alcohol use, exposure to hazardous gases, consumption of tainted foods like pickles and pickles, and drug use. In an effort to ease the burden on doctors, this dataset was used to assess prediction algorithms.

The datasets include one goal variable, Outcome, together with a number of medical predictor factors. The patient's BMI, insulin level, age, number of previous pregnancies, and other factors are predictor variables.

Any patient whose age exceeded 89 is listed as being of age "90".

Columns:

- Patient Age
- Patient Gender
- Total Bilirubin
- Direct Bilirubin
- Alkaline Phosphatase
- Alanine Aminotransferase
- Aspartate Aminotransferase
- Total Proteins
- Albumin
- Albumin and Globulin Ratio
- Dataset: Field split the data into two sets (patient with liver disease, or no disease)

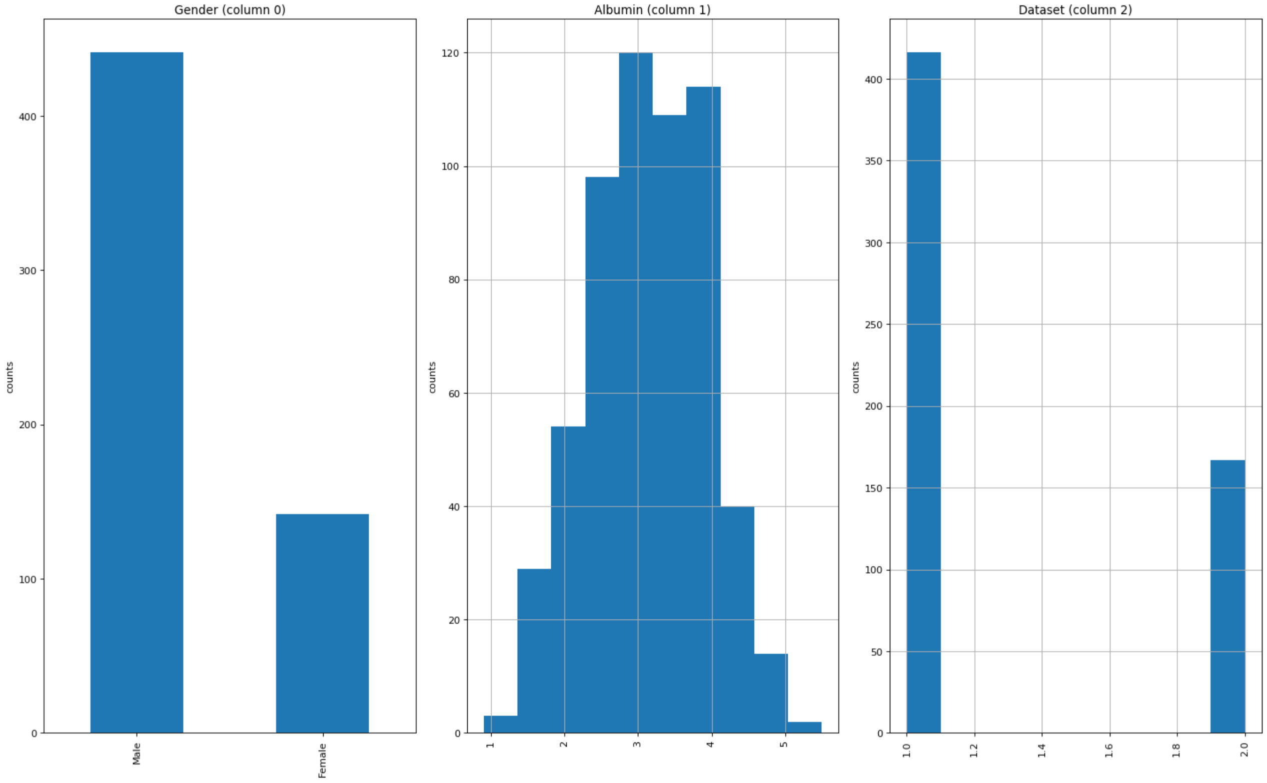
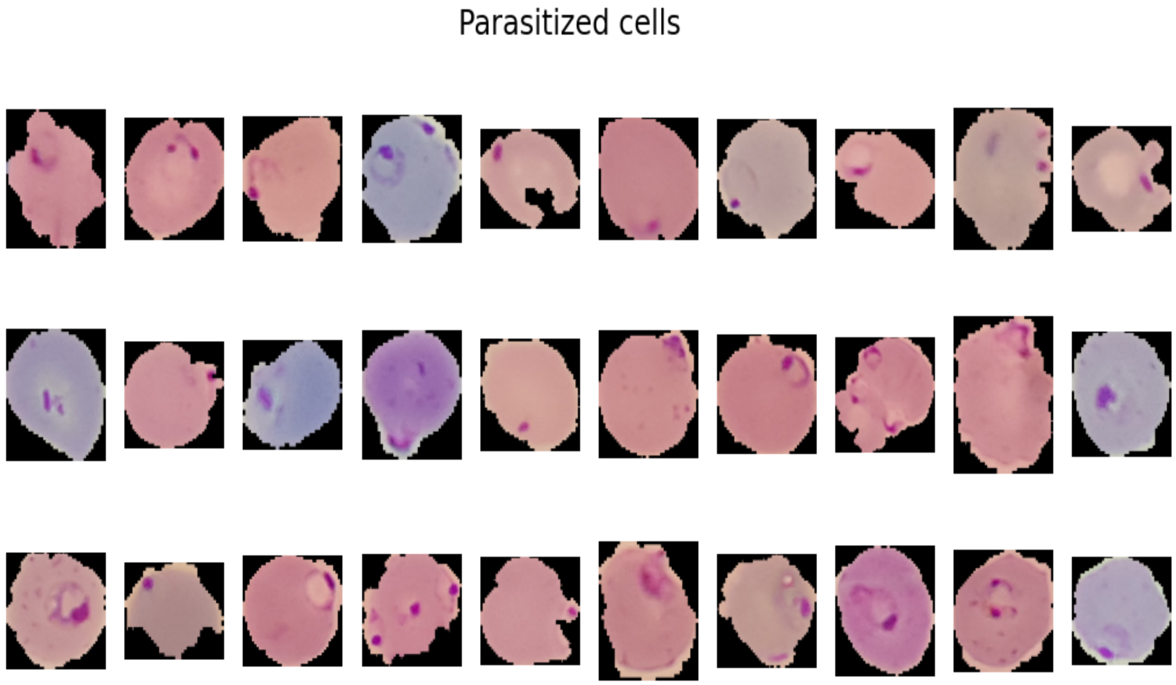


fig7. shows liver graph

- **Malaria Cell Images Dataset:** The dataset includes 27,558 photos spread over 2 files, Infected and Uninfected.


Parasitized cells

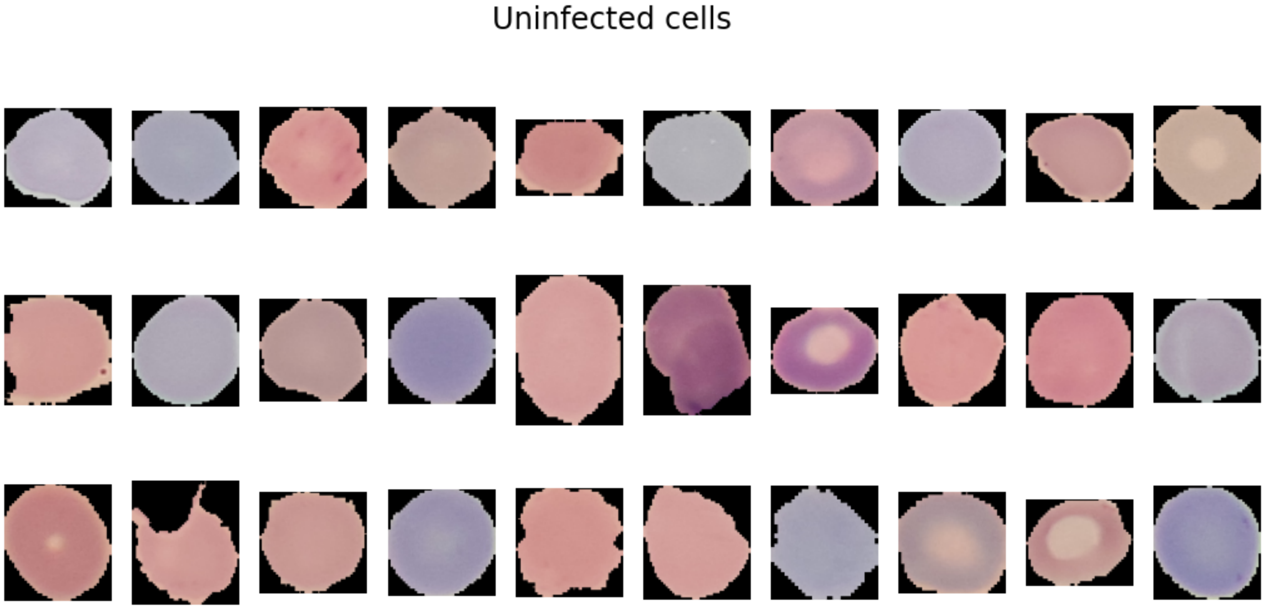

Uninfected cells

fig8. malaria image

- **Heart Attack Prediction:** The heart-disease directory's contents are described in this file. There are four databases about diagnosing heart disease in this directory. Every attribute has a numerical value. The information was gathered from the four places listed below:

1. Foundation for Cleveland Clinic (cleveland.data)
2. Budapest's Hungarian Institute of Cardiology (hungarian.data)
3. Long Beach, California's VA Medical Center (long-beach-va.data)
4. Switzerland's University Hospital, (switzerland.data)

All databases use the same instance format. In the databases, only 14 of the 76 raw attributes are actually used. I went ahead and made two copies of each database, one with all the attributes and the other with just the 14 that were actually used in prior studies.

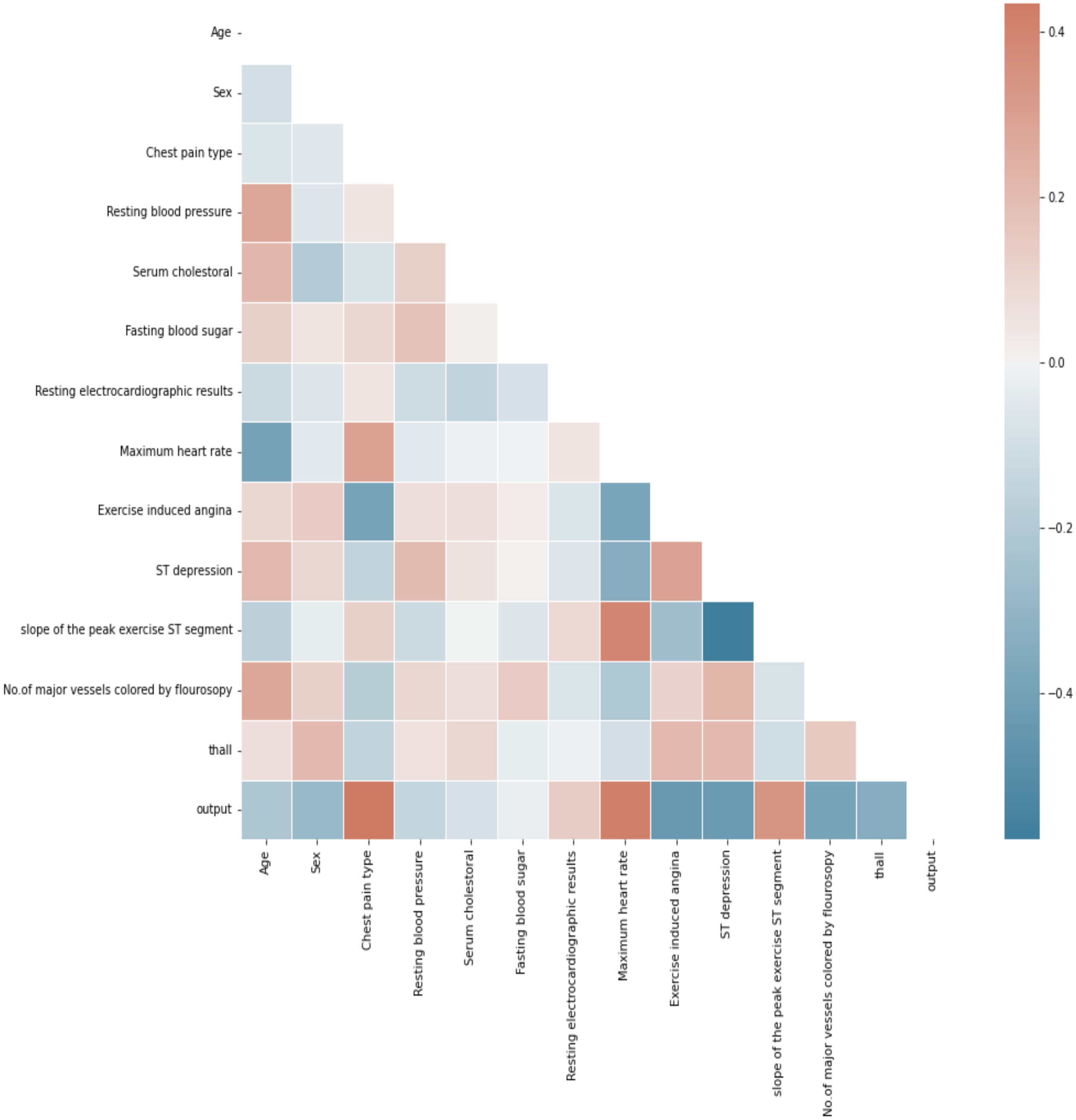


fig9. graph of heart attack

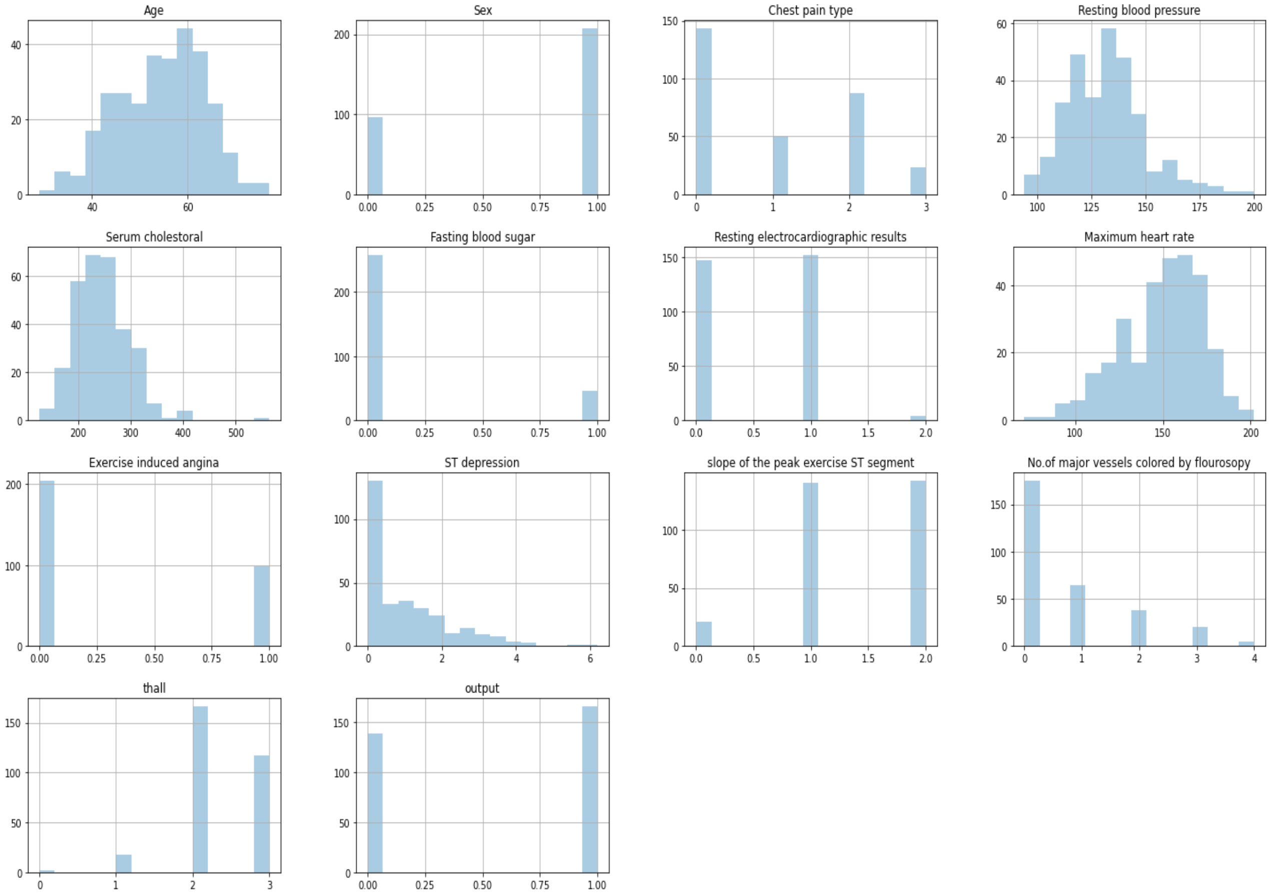


fig.10 graph of heart attack

3.3 Data Set Types

Csv : Comma Separated Values, or CSV, is the abbreviation for files. These database fields were exported into a single-line format with commas separating each database entry. CSV files resemble ordinary text files in their format. This makes it possible for users who run different database programmes to exchange database files.

PNG files : Computer vision algorithms heavily rely on datasets, which are collections of images, to mimic these cognitive functions. In computer vision, a dataset is a carefully managed collection of digital images that programmers use to test, train, and assess the effectiveness of their algorithms.

3.4 Attributes and fields of the data set

Each of the datasets contains a different set of attributes and fields-

- **Chronic Kidney disease dataset -** age, id, rbc, pc, sg etc.
- **Heart Attack Prediction -** age, sex, chol, slope, cp etc.
- **Indian Liver Patient Record -** age, sex, bilirubin, protein etc.
- **Pima Indian Diabetes Dataset**- age, BMI, pregnancies, glucose, bp etc.
- **Malaria Cell Images Dataset**- parasitized and uninfected images

3.5 Design of Problem Statement

The ultimate goal of this project is to develop models with realistic clinical potential by pushing the boundaries of an automatic disease detection mechanism that has the potential to revolutionize medical science by early diagnosis and less severity.

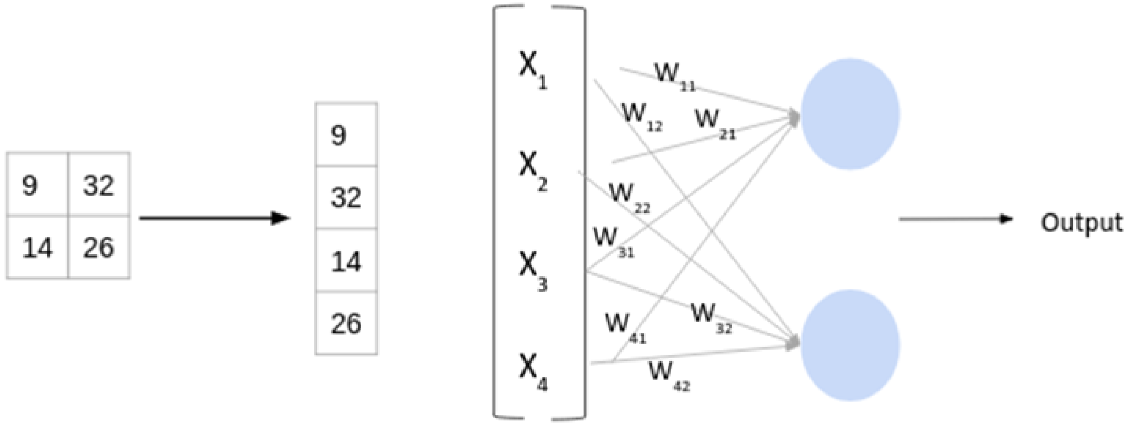
3.6 Algorithm / Pseudo code of the Project Problem

- **CNN Algorithm**

  CNN, commonly referred to as convolutional neural networks, is a very significant and fundamental algorithm utilized today.

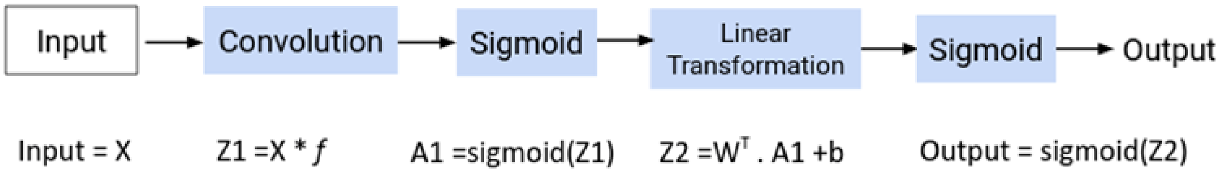
A Deep Learning system called a Convolutional Neural Network (ConvNet/CNN) is able to take an input image, assign pertinent weights and biases to various aspects and objects in the image, and then distinguish between them.

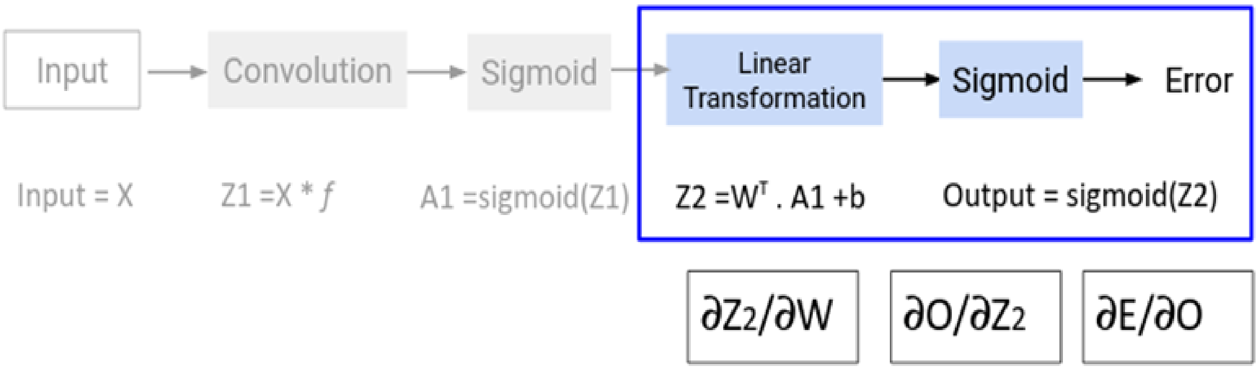
A ConvNet requires substantially less pre-processing compared to other classification techniques. ConvNets can learn these filters and properties with enough training, unlike simple techniques that require hand-engineering of filters.



$Z = WT.X + b$

Here, b (also known as bias) is a constant and X is the input while W is the weight. W will be a matrix of (randomly initialized) numbers in this instance.





Backward Propagation (Fully Connected layer)

- **Random Forest Classifier**

A component of the supervised learning approach is Random Forest. It can be used to solve classification and regression-related ML problems. It is based on the concept of ensemble learning, which is a technique for combining different classifiers.

As its name suggests, Random Forest is a classifier that averages several decision trees applied to various subsets of the input dataset to improve the predictive accuracy of the dataset. rather than 18

Using a single decision tree as its foundation, the random forest predicts the outcome by taking the prediction from each tree and basing it on the majority of predictions.

Higher accuracy and overfitting are prevented by the larger number of trees in the forest.

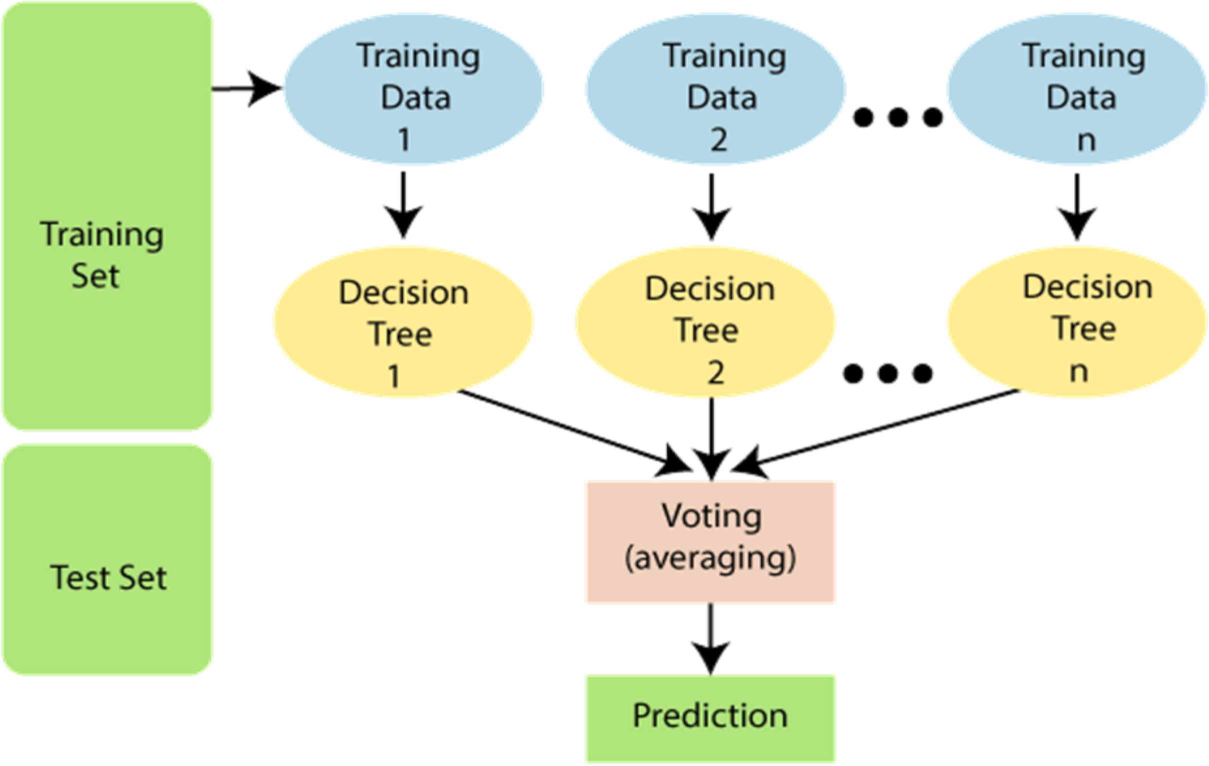
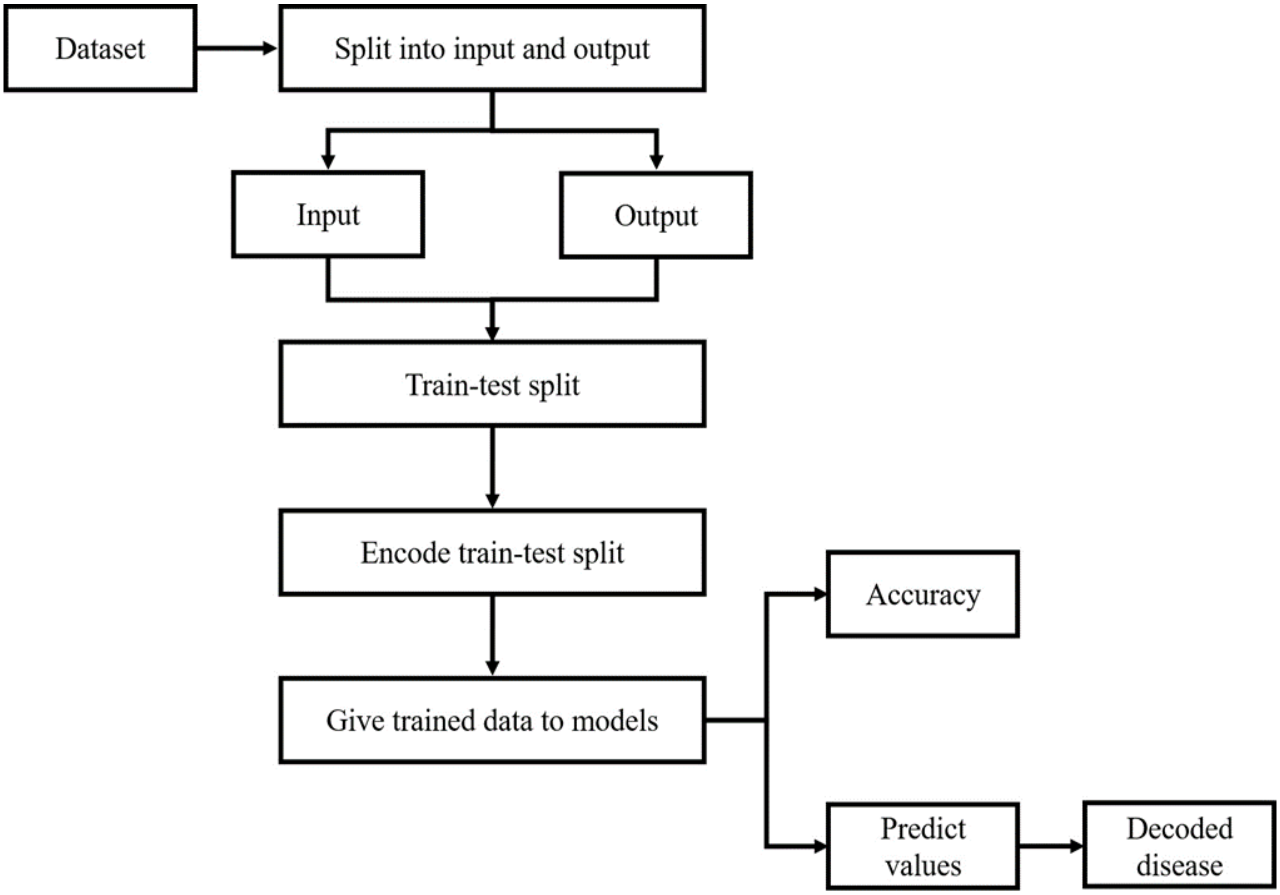


fig11. shows random forest classifier works

3.7 Flow of the Major Project Problem

- ➔ **Data collection**
- ➔ **Dataset processing**
- ➔ **Image processing**
- ➔ **Feature selection**
- ➔ **Data sorting and clearing**
- ➔ **Training csv data using Random Forest Classifier**
- ➔ **Training cell images data on CNN**
- ➔ **Training cell images on vgg19**
- ➔ **Observing the accuracy**
- ➔ **Creating a backend API on FastApi**
- ➔ **Creating a Frontend on ReactJs**
- ➔ **Deploying an Hosting of the App**



Flow diagram of project fig12. workflow of project

Chapter-4
RESULT/PERFORMANCE ANALYSIS

In machine learning, machines are presented with large amounts of data in order to learn and make predictions, find patterns, or classify data. There are three types of machine learning: supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning

In 2022, enterprise information technology leaders will continue to employ supervised learning the most [2]. In order to get the best possible match between the output and the desired result, this type of machine learning entails providing historical input and output data to a machine learning algorithm, processing between each input/output pair, and enabling the system to shift the model. Being able to produce Neural networks, decision trees, linear regression, and support vector machines are typical supervised learning techniques.

This type of machine learning is called "supervised" learning because it provides algorithmic information to assist in "supervised" learning. The system uses the remaining information you supply as an input function, and the output you send it is identified as data. If you want to understand more about the relationship between loan defaults and borrower information, for instance, you can offer the engine your 500 examples of clients who have defaulted on loans and your other 500 cases of clients who have not. To determine what data to search for, the engine "watches" the tagged data. Through supervised learning, a variety of commercial objectives, such as sales forecasting, inventory optimisation, and fraud detection, can be achieved.. An example use case is:

- Prediction of real estate prices
- Classification of whether banking transactions are fraudulent
- Finding disease risk factors
- Determining whether loan applicants are low-risk or high-risk
- Predicting the failure of industrial equipment's mechanical parts

Unsupervised learning

Unlike supervised learning, which relies on people to help the machine learn, unsupervised learning does not employ the same labeled training set and data. The algorithm instead looks for less evident patterns in the data. When you need to identify patterns and use data to inform decisions, this type of machine learning is particularly helpful. Unsupervised learning methods that are frequently used include hidden Markov models, k-means clustering, hierarchical clustering, and Gaussian mixture models.

Through the use of this type of machine learning, predictive models are frequently created. Additionally, correlations and clustering are frequently employed to find the rules that connect clusters. A model that organizes items based on specific attributes is created through clustering. There are numerous application scenarios, including :

- Group customers based on buying habits
- Group inventory levels based on manufacturing and/or sales metrics
- Identify customer data relationships (e.g., customers who buy a particular handbag may may be interested in shoes in the style of

Reinforcement learning

The machine learning technique that most closely matches human learning is reinforcement learning. When an algorithm or agent interacts with the environment and receives rewards—whether positive or negative—it learns. Common algorithms include deep adversarial networks, Q-learning, and time difference.

Thinking back to the example of a customer with a bank loan, we can use reinforcement learning to explore customer data. Algorithms will benefit if you mark it as high risk and switch to default mode. If you do not violate your obligations, you will receive a negative reward from the algorithm. By improving awareness of the problem and surroundings, both practices ultimately benefit machine learning.

According to Gartner, most machine learning systems do not have reinforcement learning capabilities because most companies lack the necessary computing power. Situations that are entirely reproducible, immovable, or entail a tonne of relevant data can all benefit from reinforcement learning. Since this type of machine learning necessitates less maintenance than supervised learning, it is said to be easier to use when working with unlabeled datasets. This kind of machine learning is still used in real world applications. Some usage examples are:

- Teaching vehicles how to park and drive themselves
- Dynamically adjust traffic signals to reduce congestion
- Use raw video as input to train robots how to follow rules so they can copy observed behavior

Us The project involves supervised learning using KNN. , random forest algorithms and deep learning

KNN (K-Nearest Neighbors)

According to the KNN algorithm, related objects are close by. To put it another way, related items are situated close to one another.

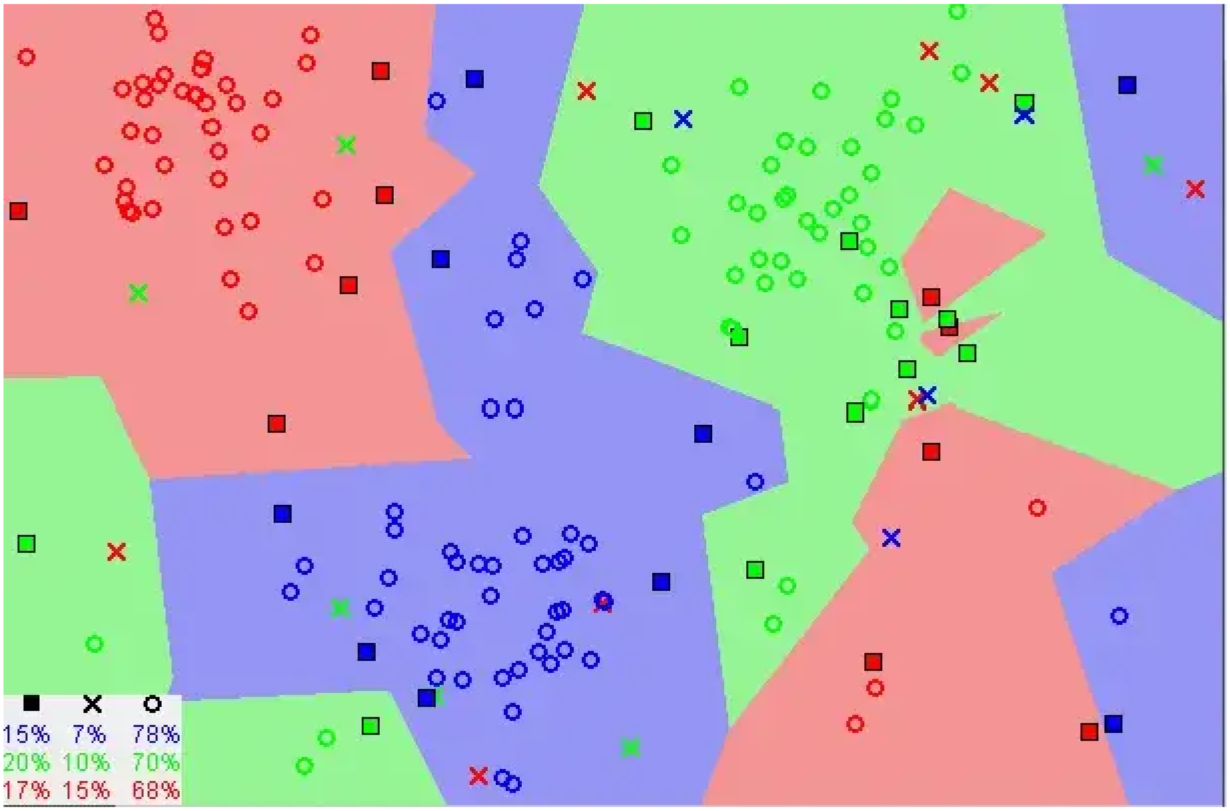


fig13. working of KNN model

In most cases, the comparable data points in the graph above are close to each other.In order for the KNN algorithm to be effective, this presumption must be sufficiently true. KNN employs some of the math we may have learned as children to represent the concept of similarity (also known as distance, closeness, or proximity). (calculating the distance between points on a graph).

Before proceeding, it is important to know how to determine the distance between points in a graph. If you're not familiar with how this calculation is done, or just need a reminder, read "Distance Between Two Points" carefully.

There are other ways to calculate distance, and depending on the problem at hand, one of them may be better for you. However, a popular and well-known variant is direct distance (also known as Euclidean distance).

KNN Algorithm

1. Enter the information up to

2. Set K to the number of neighbors you choose.

3. For each data example, determine the distance between them

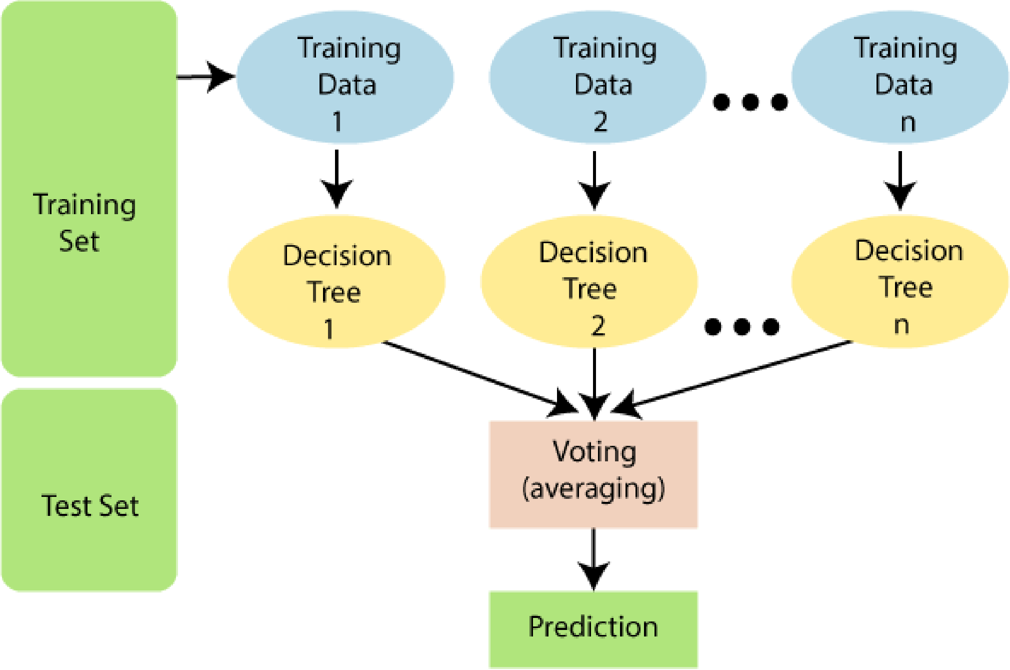
3.1 query example and the current example of data.

3.2 Add the distance and index of the examples to the sorted collection.

4. Sort the distances in ascending order from smallest to largest in the sorted set of indices and distances.

Random Forest Algorithm

Random Forest is a component of the supervised learning algorithm used in machine learning techniques.It can be applied to solve classification and regression issues in ML. The idea of ensemble learning, which combines various classifiers to solve complicated issues and enhance model functioning, serves as its conceptual underpinning. A random forest, as its name suggests, is a classifier that increases the predicted accuracy of a dataset by averaging numerous decision trees over various informational subsets. Suitable Trees Instead of relying on individual predictions, Random Forest predicts each tree and the outcome using the majority vote of the predictions. The more trees there are in the forest, the better the accuracy and the ability to avoid overfitting problems.



Deep learning

Deep learning can be seen as a subset of machine learning, as is well known. a discipline dedicated to the study of computer algorithms for learning and self-development. Deep learning, however, makes use of artificial neural networks that were made to closely imitate how people think and learn , machine learning makes use of straightforward ideas. Neural networks' complexity was previously constrained by their processing speed. Complicated neural networks which are larger in size may now be created thanks to developments in big data analytics, which will enable computers to monitor, learn, and react to complex events more quickly than people. Speech recognition, language translation, and image categorization are all supported by deep learning. Problems with pattern recognition can be resolved using it, and it can be used without human intervention. Similar to the human brain, a neural network is a layer of nodes made up of neurons. Nodes inside different layers are linked to layers close by. A network is categorized as deeper based on how many layers it has. In the human brain, one neuron gets many impulses from other neurons. Signals are transferred between nodes in artificial neural networks, and nodes are given the proper weights. The following node hierarchy is more influenced by nodes with higher weights. The output is created by combining the weighted inputs in the final layer.

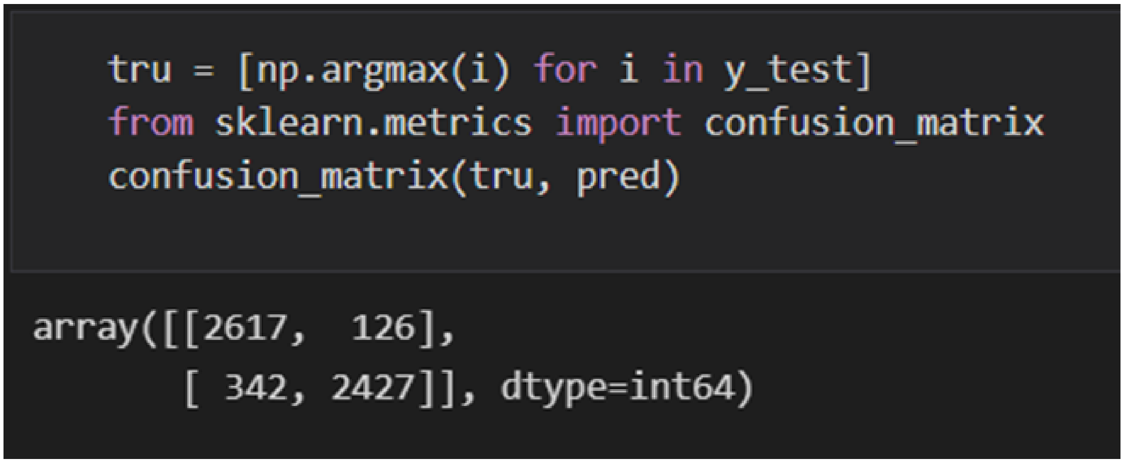
Deep learning systems require powerful hardware as they process large amounts of data and involve many complex mathematical calculations.

Chapter-5
CONCLUSIONS

1.1 Conclusions

Four of our datasets—the heart dataset, kidney dataset, liver dataset, and diabetes dataset—were trained using the Random Forest Classifier, and the accuracy results were as follows:

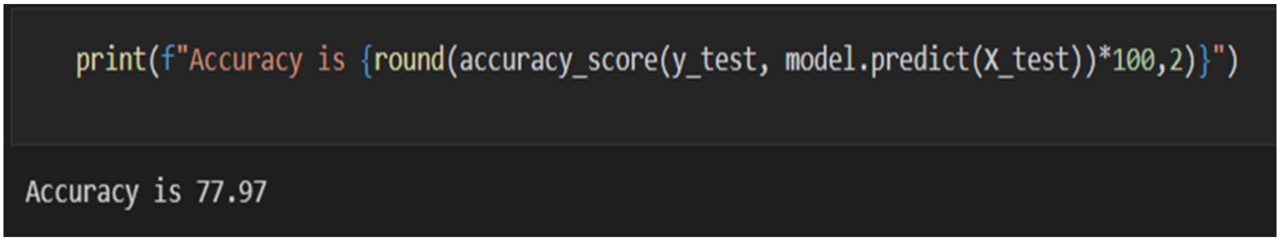
- Model for Predicting Heart Disease: 81.97%
- Model for Predicting Kidney Disease: 98 %
- Model for Predicting Liver Disease: 77.97%
- Model for Predicting Diabetes Disease: 98.25 %
- We used CNN to train the malaria cell image dataset using VGG19, and the accuracy was 91.51%.



```
tru = [np.argmax(i) for i in y_test]
from sklearn.metrics import confusion_matrix
confusion_matrix(tru, pred)
```

```
array([[2617,  126],
       [ 342, 2427]], dtype=int64)
```

Fig 14. Confusion matrix for Malaria dataset



```
print(f"Accuracy is {round(accuracy_score(y_test, model.predict(X_test))*100,2)}")
```

```
Accuracy is 77.97
```

Fig 15 Accuracy score for Liver dataset

We could simply manage the medical resources needed for the treatment once the sickness was predicted. This strategy would aid in reducing the expense involved in treating the sickness and would also speed up the healing process.

In an emergency, doctors and other medical personnel are constantly needed. Our prediction system can be useful in the current COVID-19 situation where sufficient resources are lacking and can be used to diagnose a disease.

1.1 Future Scope

To increase accuracy, create better machines, and add additional tests for other diseases, we want to work on training these algorithms on very big datasets in the future. Additionally, we want to include a database management system that will enable users to save and update the supplied login parameters.

1.3 Applications of the Project

There are several real-world uses for them, some of which include:

1. It may be used as a home health consultant, saving time and money spent on doctor visits.

2. It can assist those in rural and underdeveloped areas who are unable to obtain good consultation facilities due to cost or lack of access.

3. It can allow older people the opportunity to do tests at home and receive findings right away, saving them the trouble of having to visit a doctor.

4. The web app might be developed further to assist physicians in getting a data-influenced illness analysis, which would provide a safer approach to medical diagnosis and lead to improved diagnosis.

References

- D. Dahiwade, G. Patle and E. Meshram, "Designing Disease Prediction Model Using Machine Learning Approach," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019, pp. 1211-1215, doi: 10.1109/ICCMC.2019.8819782.

- P. S. Kohli and S. Arora, "Application of Machine Learning in Disease Prediction," 2018 4th International Conference on Computing Communication and Automation (ICCCA), 2018, pp. 1-4, doi: 10.1109/CCAA.2018.8777449.

- A. Gavhane, G. Kokkula, I. Pandya and K. Devadkar, "Prediction of Heart Disease Using Machine Learning," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 1275-1278, doi: 10.1109/ICECA.2018.8474922.

- R.D.H.D.P. Sreevalli, K.P.M. Asia, Prediction of diseases using random forest classification algorithm.

- D.R. Langbehn, R.R. Brinkman, D. Falush, J.S. Paulsen, M. Hayden, an International Huntington's Disease Collaborative Group, A new model for prediction of the age of onset and penetrance for Huntington's disease based on cag length, Clinical genetics 65(4), 267 (2004).

- M. Maniruzzaman, M.J. Rahman, B. Ahammed, M.M. Abedin, Classification and prediction of diabetes disease using machine learning paradigm, Health Information Science and Systems 8(1), 7 (2020).

- M. Chen, Y. Hao, K. Hwang, L. Wang, L. Wang, Disease prediction by machine learning over big data from healthcare communities, Ieee Access 5, 8869 (2017)

Appendices

Technologies / language used :

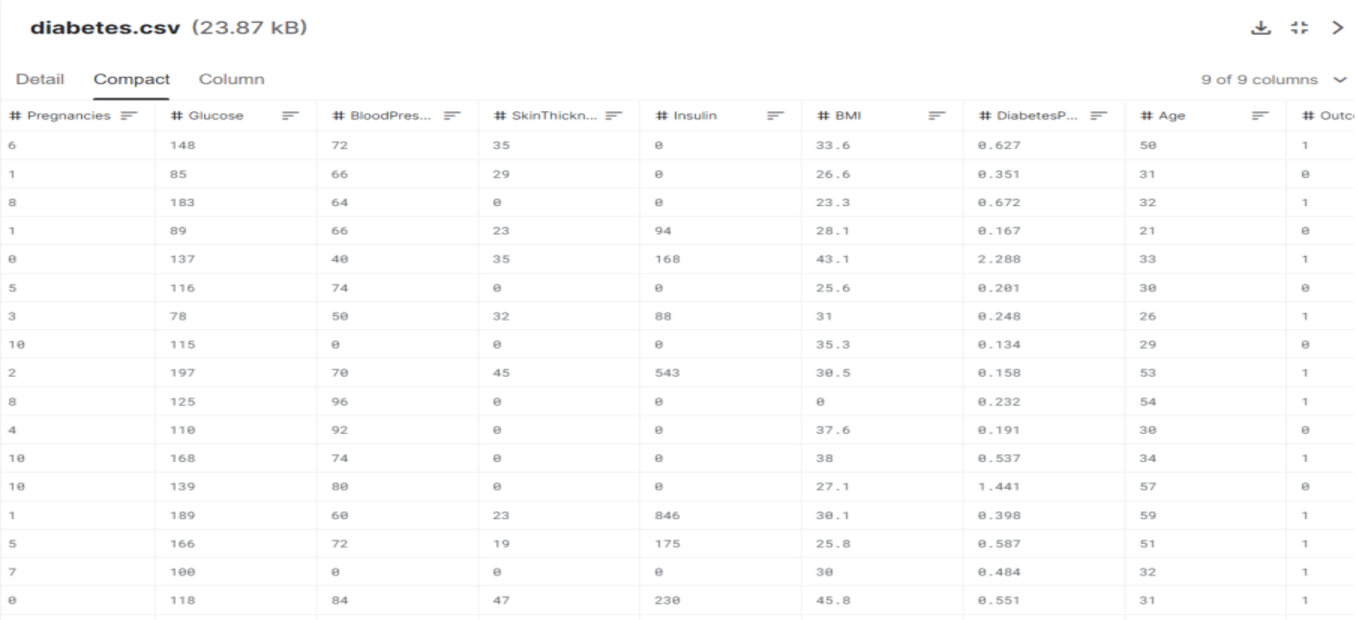
- **Numpy :** A key Python package for scientific computing is called NumPy, which stands for Numerical Python. It provides useful multi-dimensional array objects in addition to a variety of ways to employ these array objects.

- **Pandas :** Pandas is mostly used for tabular data manipulation in data frames together with data analysis.

- **Sklearn :** It provides a variety of efficient statistical and machine learning modelling techniques, including clustering, regression, and classification.

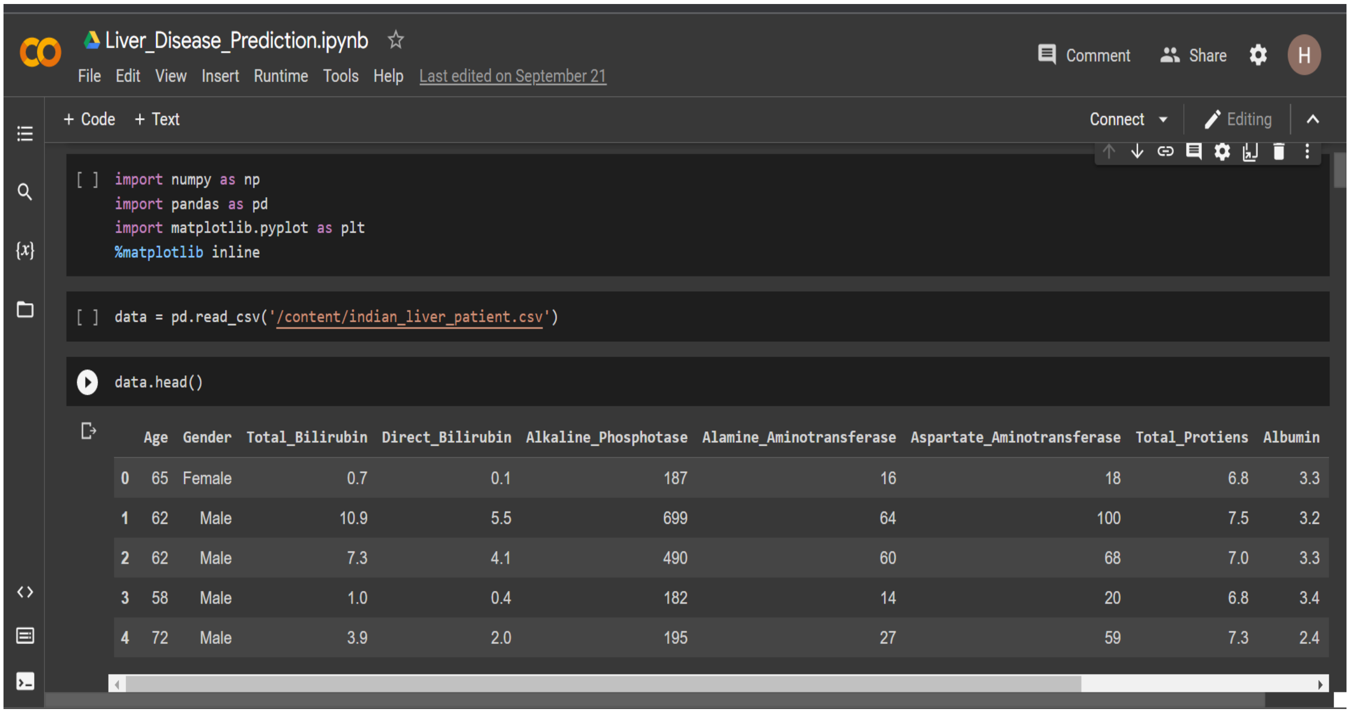
- **Seaborn :** You can create statistical visualizations using Python's Seaborn package. Tightly integrated with Pandas data structures and built on top of Matplotlib.

Source Code :



diabetes.csv (23.87 kB)

# Pregnancies	# Glucose	# BloodPres...	# SkinThickn...	# Insulin	# BMI	# DiabetesP...	# Age	# Outc
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0
3	78	50	32	88	31	0.248	26	1
10	115	0	0	0	35.3	0.134	29	0
2	197	70	45	543	30.5	0.158	53	1
8	125	96	0	0	0	0.232	54	1
4	110	92	0	0	37.6	0.191	30	0
10	168	74	0	0	38	0.537	34	1
10	139	80	0	0	27.1	1.441	57	0
1	189	60	23	846	30.1	0.398	59	1
5	166	72	19	175	25.8	0.587	51	1
7	100	0	0	0	30	0.484	32	1
0	118	84	47	230	45.8	0.551	31	1

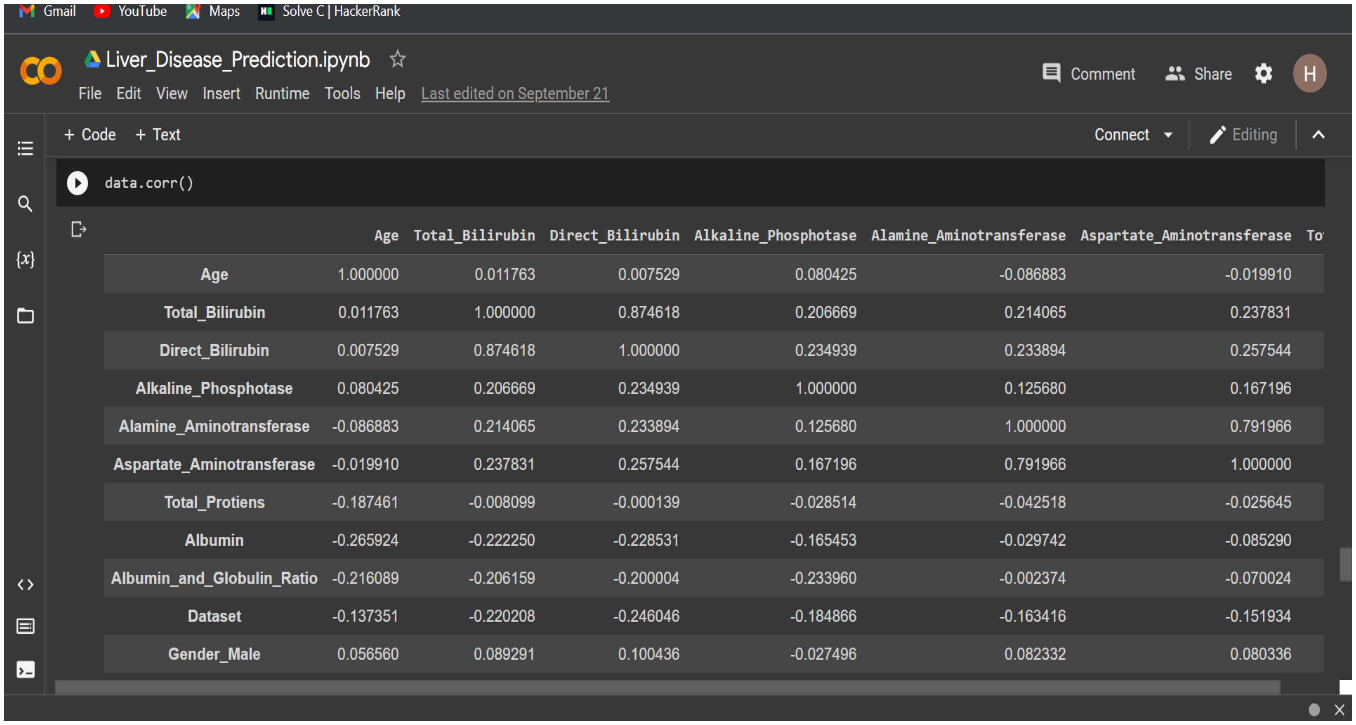

Liver_Disease_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 21
Comment Share
+ Code + Text
Connect Editing

```
[ ] import numpy as np
    import pandas as pd
    import matplotlib.pyplot as plt
    %matplotlib inline
```

```
[ ] data = pd.read_csv('/content/indian_liver_patient.csv')
```

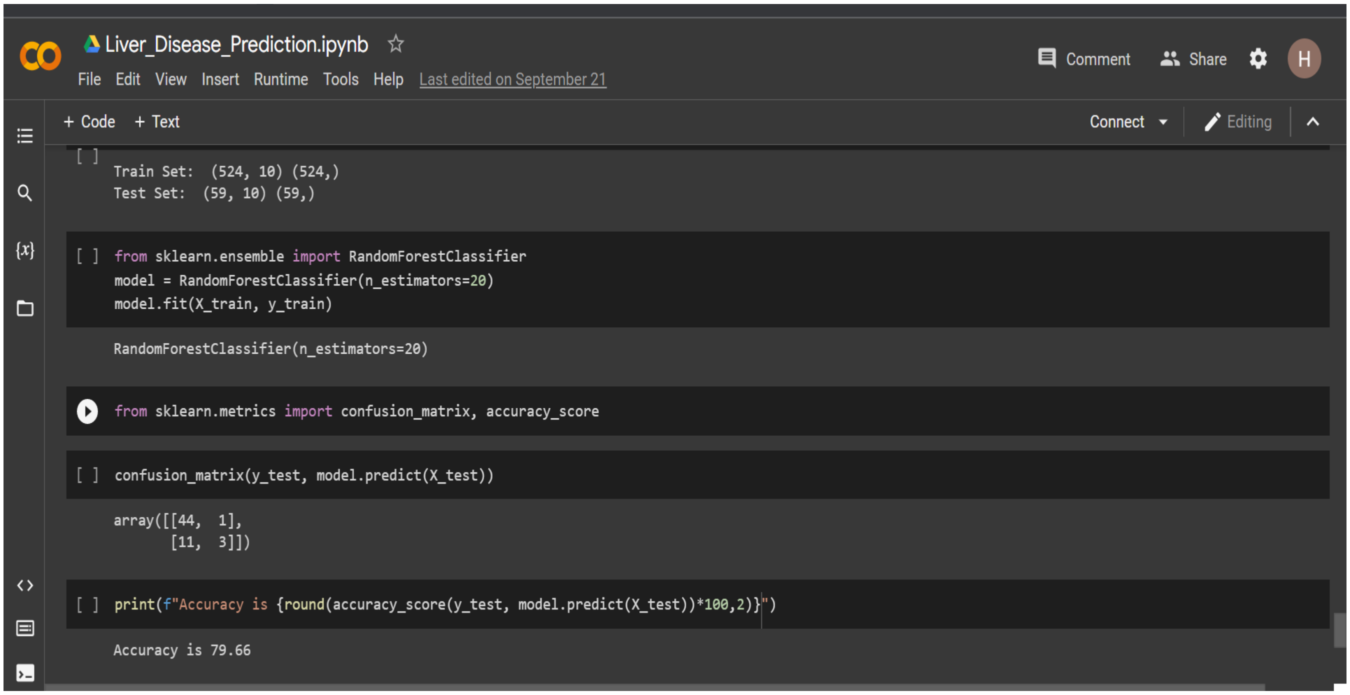
```
data.head()
```

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin
0	65	Female	0.7	0.1	187	16	18	6.8	3.3
1	62	Male	10.9	5.5	699	64	100	7.5	3.2
2	62	Male	7.3	4.1	490	60	68	7.0	3.3
3	58	Male	1.0	0.4	182	14	20	6.8	3.4
4	72	Male	3.9	2.0	195	27	59	7.3	2.4


Gmail YouTube Maps Solve C | HackerRank
Liver_Disease_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 21
Comment Share
+ Code + Text
Connect Editing

```
data.corr()
```

	Age	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	To
Age	1.000000	0.011763	0.007529	0.080425	-0.086883	-0.019910	
Total_Bilirubin	0.011763	1.000000	0.874618	0.206669	0.214065	0.237831	
Direct_Bilirubin	0.007529	0.874618	1.000000	0.234939	0.233894	0.257544	
Alkaline_Phosphotase	0.080425	0.206669	0.234939	1.000000	0.125680	0.167196	
Alamine_Aminotransferase	-0.086883	0.214065	0.233894	0.125680	1.000000	0.791966	
Aspartate_Aminotransferase	-0.019910	0.237831	0.257544	0.167196	0.791966	1.000000	
Total_Protiens	-0.187461	-0.008099	-0.000139	-0.028514	-0.042518	-0.025645	
Albumin	-0.265924	-0.222250	-0.228531	-0.165453	-0.029742	-0.085290	
Albumin_and_Globulin_Ratio	-0.216089	-0.206159	-0.200004	-0.233960	-0.002374	-0.070024	
Dataset	-0.137351	-0.220208	-0.246046	-0.184866	-0.163416	-0.151934	
Gender_Male	0.056560	0.089291	0.100436	-0.027496	0.082332	0.080336	


CO
Liver_Disease_Prediction.ipynb ☆
File Edit View Insert Runtime Tools Help Last edited on September 21
Comment
Share
H
+ Code + Text
Connect
Editing

```
Train Set:  (524, 10) (524,)
Test Set:  (59, 10) (59,)
```

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=20)
model.fit(X_train, y_train)
```

```
RandomForestClassifier(n_estimators=20)
```

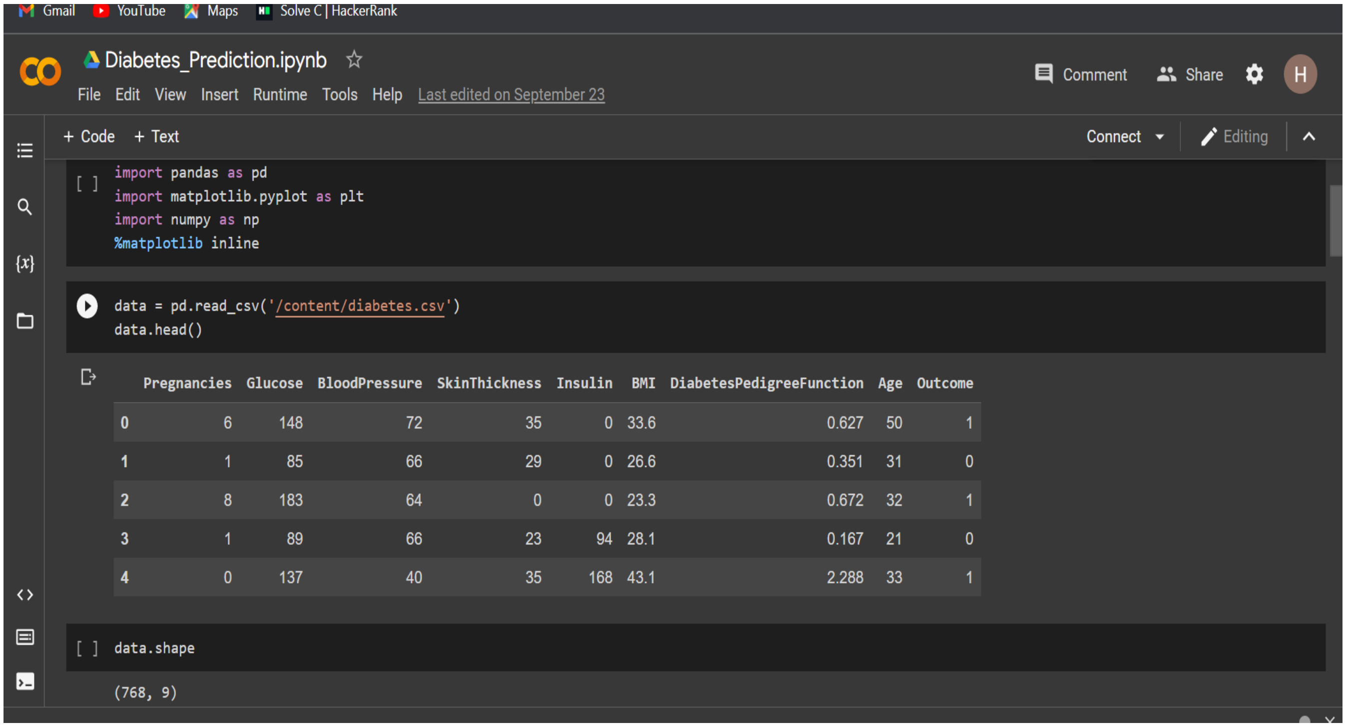
```
from sklearn.metrics import confusion_matrix, accuracy_score
```

```
confusion_matrix(y_test, model.predict(X_test))
```

```
array([[44,  1],
       [11,  3]])
```

```
print(f"Accuracy is {round(accuracy_score(y_test, model.predict(X_test))*100,2)}")
```

```
Accuracy is 79.66
```


Gmail
YouTube
Maps
Solve C | HackerRank
CO
Diabetes_Prediction.ipynb ☆
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment
Share
H
+ Code + Text
Connect
Editing

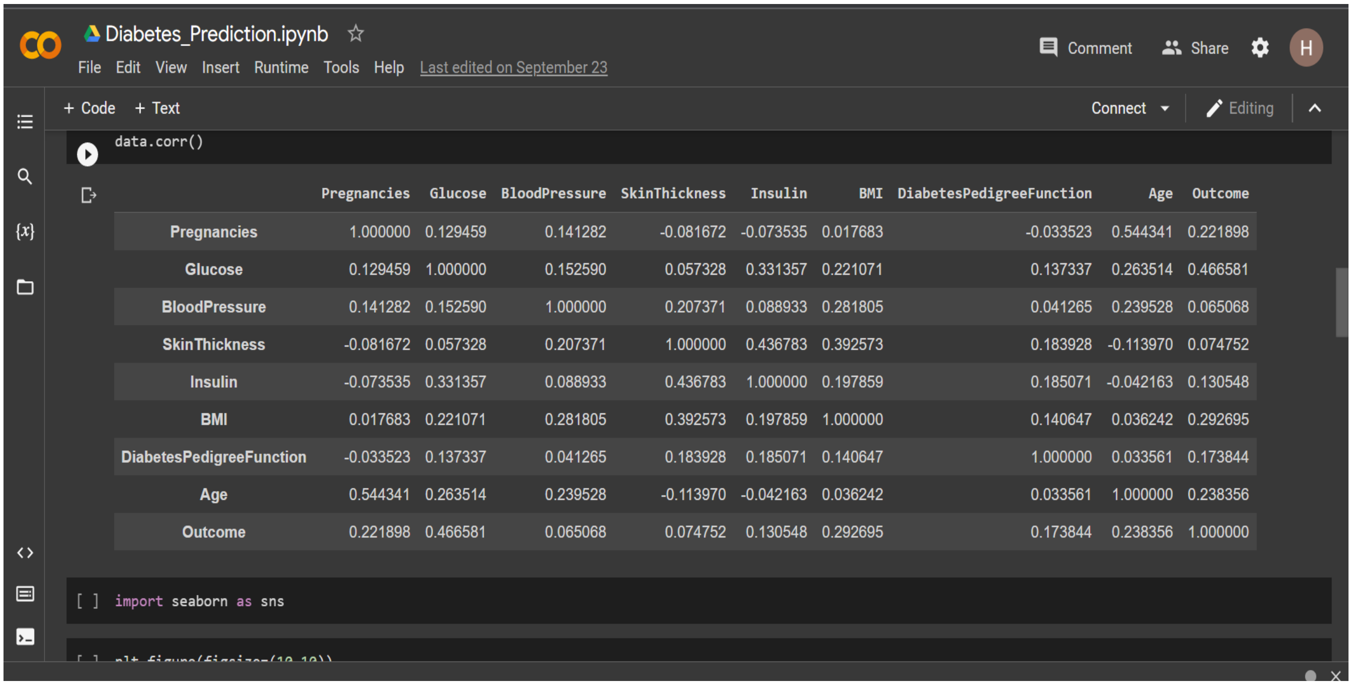
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

```
data = pd.read_csv('/content/diabetes.csv')
data.head()
```

	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

```
data.shape
```

```
(768, 9)
```

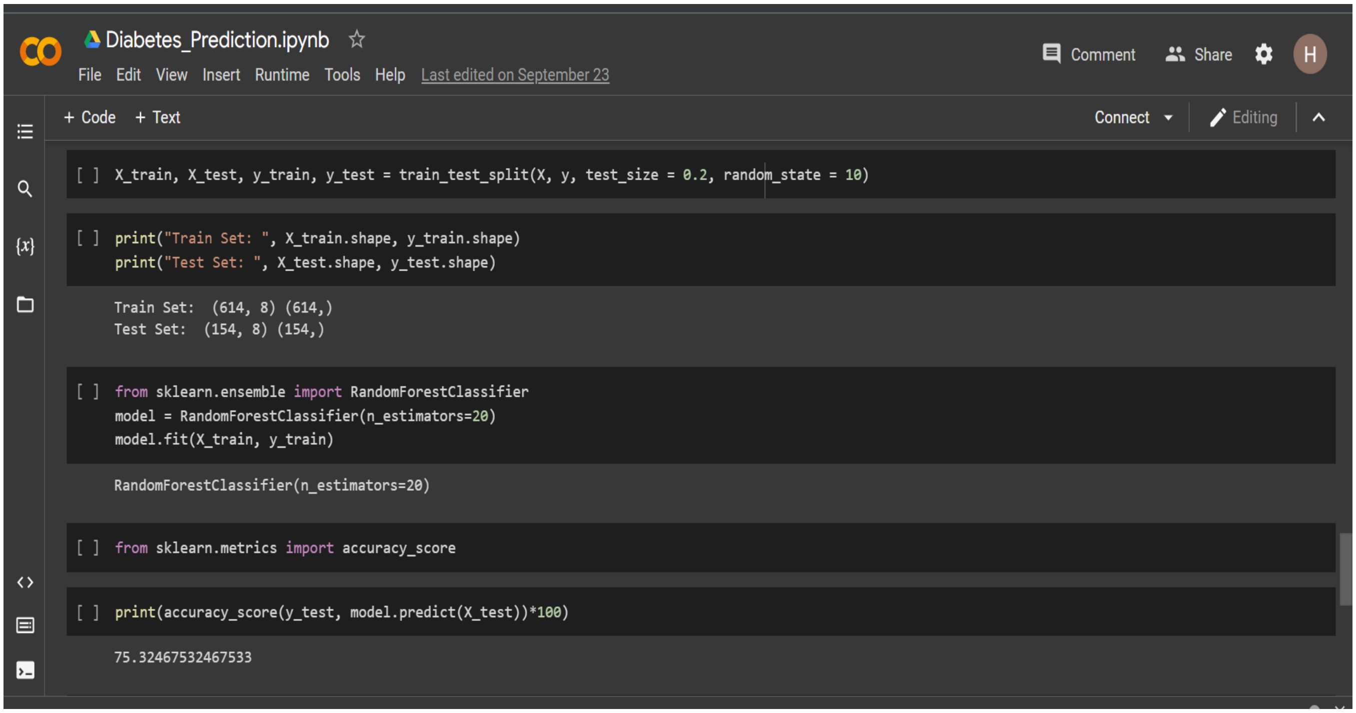

Diabetes_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment Share H
+ Code + Text
Connect Editing

```
data.corr()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
Pregnancies	1.000000	0.129459	0.141282	-0.081672	-0.073535	0.017683	-0.033523	0.544341	0.221898
Glucose	0.129459	1.000000	0.152590	0.057328	0.331357	0.221071	0.137337	0.263514	0.466581
BloodPressure	0.141282	0.152590	1.000000	0.207371	0.088933	0.281805	0.041265	0.239528	0.065068
SkinThickness	-0.081672	0.057328	0.207371	1.000000	0.436783	0.392573	0.183928	-0.113970	0.074752
Insulin	-0.073535	0.331357	0.088933	0.436783	1.000000	0.197859	0.185071	-0.042163	0.130548
BMI	0.017683	0.221071	0.281805	0.392573	0.197859	1.000000	0.140647	0.036242	0.292695
DiabetesPedigreeFunction	-0.033523	0.137337	0.041265	0.183928	0.185071	0.140647	1.000000	0.033561	0.173844
Age	0.544341	0.263514	0.239528	-0.113970	-0.042163	0.036242	0.033561	1.000000	0.238356
Outcome	0.221898	0.466581	0.065068	0.074752	0.130548	0.292695	0.173844	0.238356	1.000000

```
[ ] import seaborn as sns
```

```
[ ] plt.figure(figsize=(10,10))
```


Diabetes_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment Share H
+ Code + Text
Connect Editing

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 10)
```

```
[ ] print("Train Set: ", X_train.shape, y_train.shape)
    print("Test Set: ", X_test.shape, y_test.shape)

    Train Set:  (614, 8) (614,)
    Test Set:  (154, 8) (154,)
```

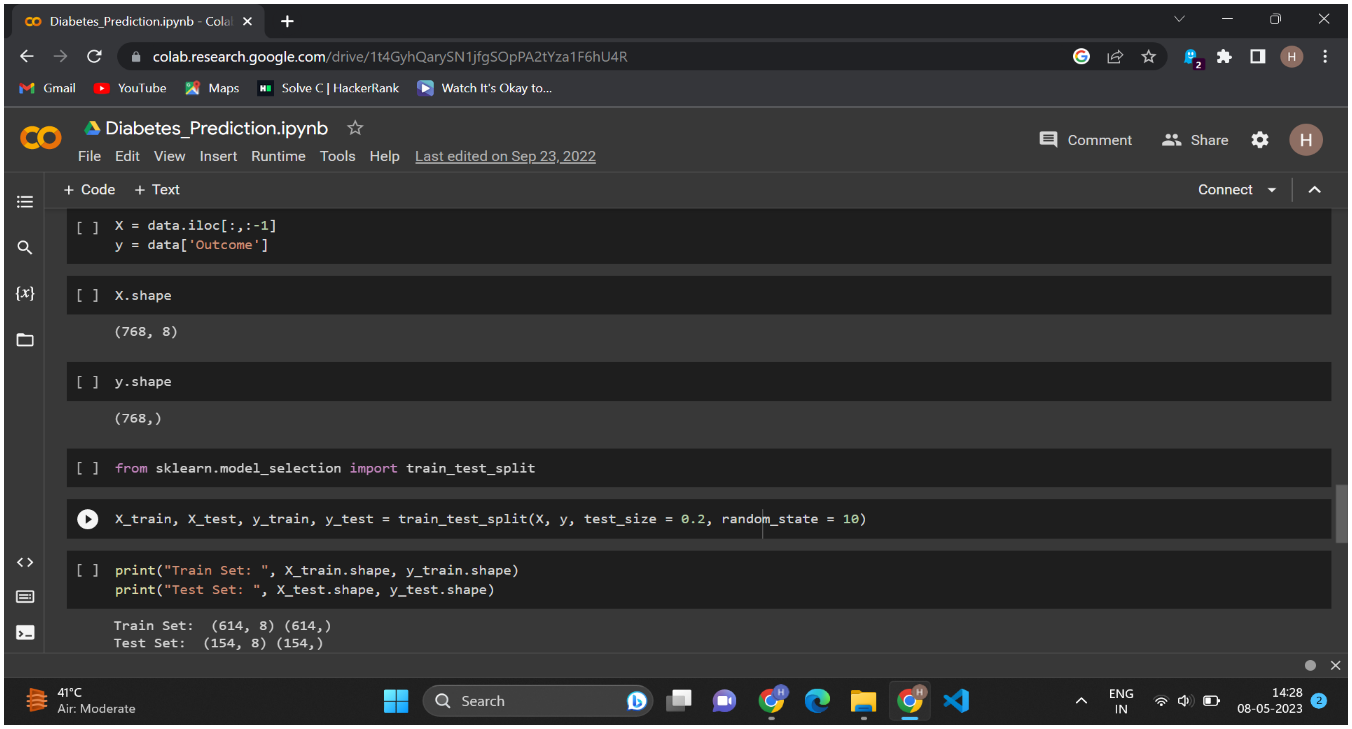
```
[ ] from sklearn.ensemble import RandomForestClassifier
    model = RandomForestClassifier(n_estimators=20)
    model.fit(X_train, y_train)

    RandomForestClassifier(n_estimators=20)
```

```
[ ] from sklearn.metrics import accuracy_score
```

```
[ ] print(accuracy_score(y_test, model.predict(X_test))*100)

    75.32467532467533
```


Diabetes_Prediction.ipynb - Colab
colab.research.google.com/drive/1t4GyhQarySN1jfgSOpPA2tYza1F6hU4R
Gmail
YouTube
Maps
Solve C | HackerRank
Watch It's Okay to...
Diabetes_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on Sep 23, 2022
Comment
Share
+ Code + Text
Connect
41°C
Air: Moderate
Search
ENG
IN
14:28
08-05-2023

```
X = data.iloc[:,:-1]
y = data['Outcome']
```

```
X.shape
```

```
(768, 8)
```

```
y.shape
```

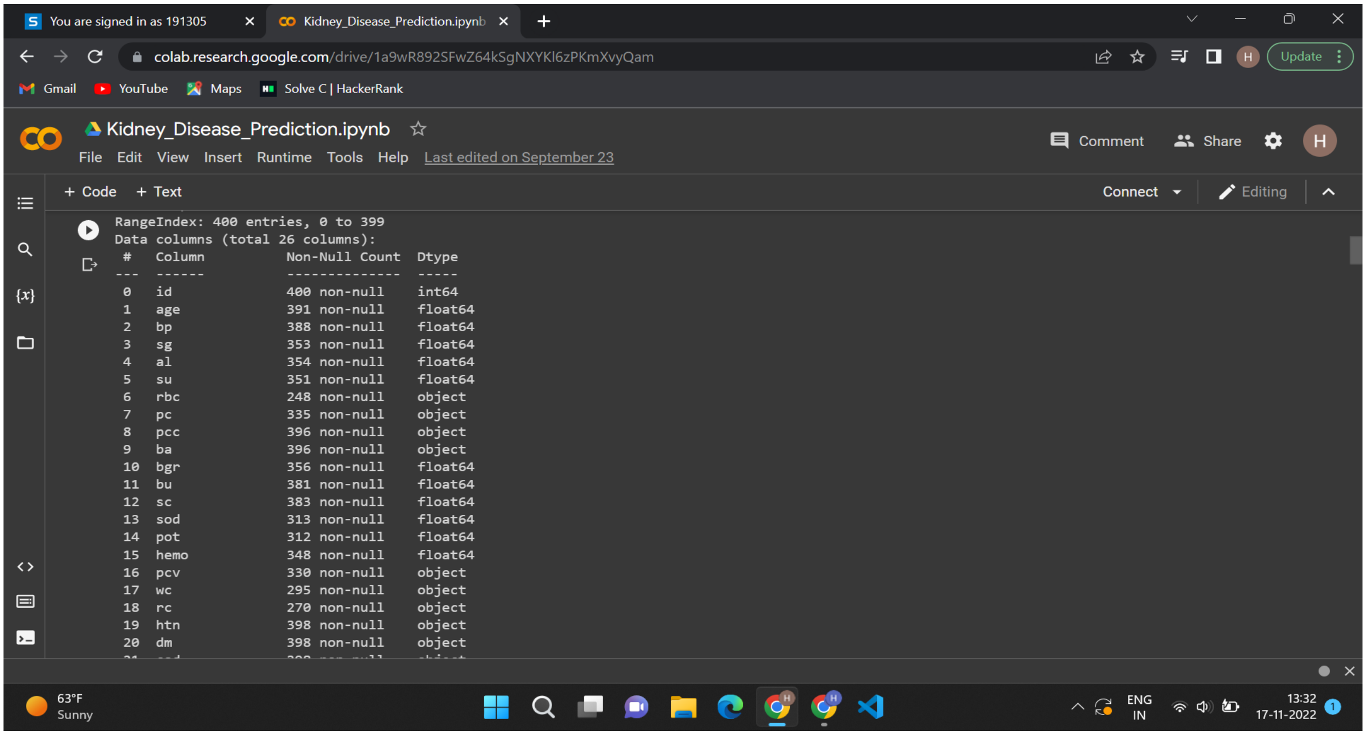
```
(768,)
```

```
from sklearn.model_selection import train_test_split
```

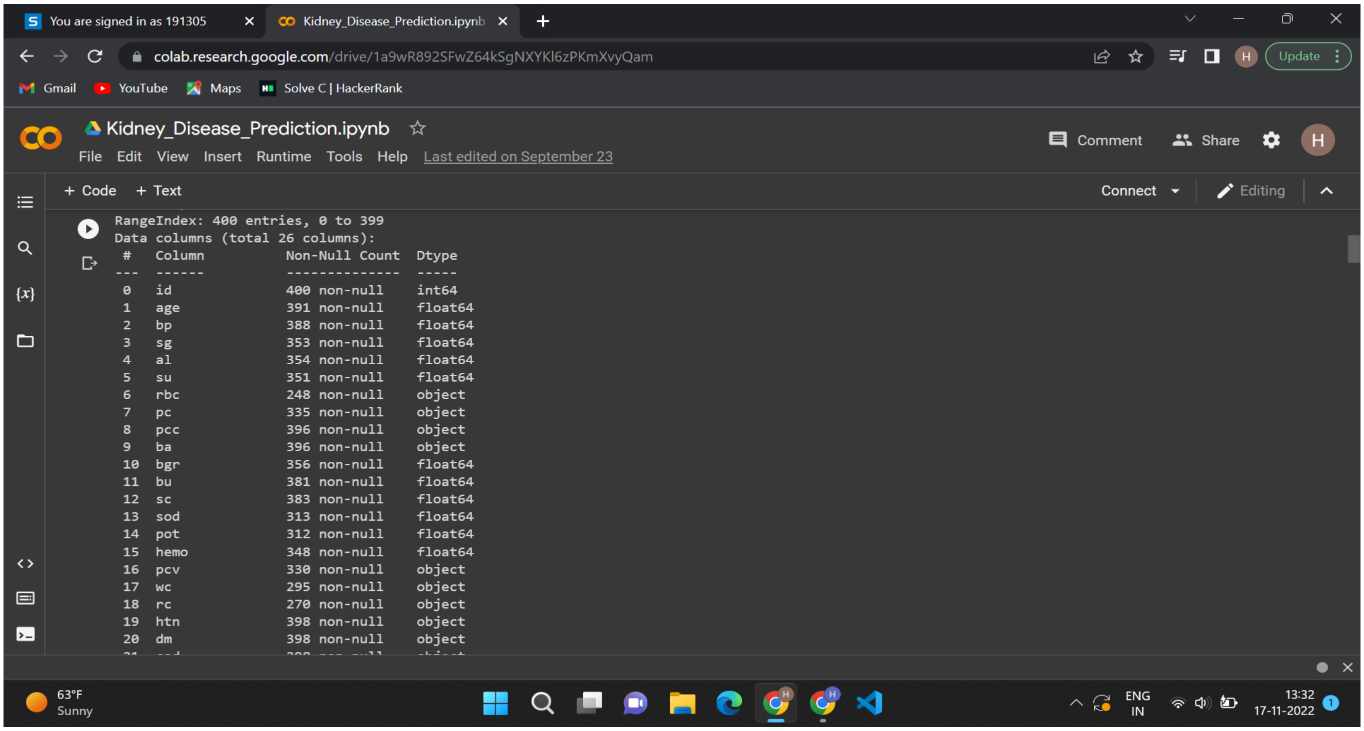
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 10)
```

```
print("Train Set: ", X_train.shape, y_train.shape)
print("Test Set: ", X_test.shape, y_test.shape)
```

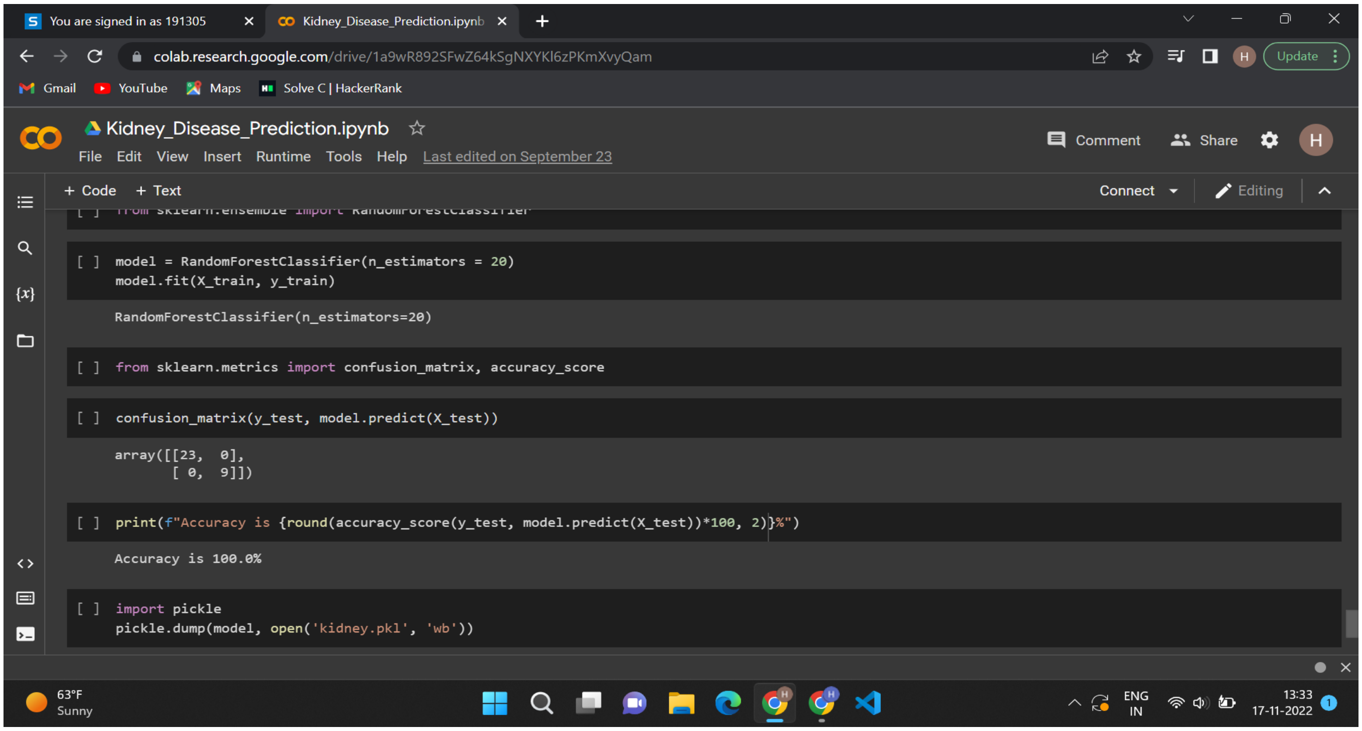
```
Train Set:  (614, 8) (614,)
Test Set:  (154, 8) (154,)
```


You are signed in as 191305
Kidney_Disease_Prediction.ipynb
colab.research.google.com/drive/1a9wR892SFwZ64kSgNXYKl6zPKmXvyQam
Update
Gmail
YouTube
Maps
Solve C | HackerRank
Kidney_Disease_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment
Share
+ Code + Text
Connect
Editing
63°F
Sunny
ENG
IN
13:32
17-11-2022

```
RangeIndex: 400 entries, 0 to 399
Data columns (total 26 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   id              400 non-null    int64
 1   age             391 non-null    float64
 2   bp              388 non-null    float64
 3   sg              353 non-null    float64
 4   al              354 non-null    float64
 5   su              351 non-null    float64
 6   rbc             248 non-null    object
 7   pc              335 non-null    object
 8   pcc             396 non-null    object
 9   ba              396 non-null    object
 10  bgr             356 non-null    float64
 11  bu              381 non-null    float64
 12  sc              383 non-null    float64
 13  sod             313 non-null    float64
 14  pot             312 non-null    float64
 15  hemo            348 non-null    float64
 16  pcv             330 non-null    object
 17  wc              295 non-null    object
 18  rc              270 non-null    object
 19  htn             398 non-null    object
 20  dm              398 non-null    object
```


You are signed in as 191305
Kidney_Disease_Prediction.ipynb
colab.research.google.com/drive/1a9wR892SFwZ64kSgNXYKl6zPKmXvyQam
Gmail
YouTube
Maps
Solve C | HackerRank
Kidney_Disease_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment
Share
+ Code
+ Text
Connect
Editing
63°F
Sunny
ENG
IN
13:32
17-11-2022

```
RangeIndex: 400 entries, 0 to 399
Data columns (total 26 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   id              400 non-null    int64
 1   age             391 non-null    float64
 2   bp              388 non-null    float64
 3   sg              353 non-null    float64
 4   al              354 non-null    float64
 5   su              351 non-null    float64
 6   rbc             248 non-null    object
 7   pc              335 non-null    object
 8   pcc             396 non-null    object
 9   ba              396 non-null    object
 10  bgr             356 non-null    float64
 11  bu              381 non-null    float64
 12  sc              383 non-null    float64
 13  sod             313 non-null    float64
 14  pot             312 non-null    float64
 15  hemo            348 non-null    float64
 16  pcv             330 non-null    object
 17  wc              295 non-null    object
 18  rc              270 non-null    object
 19  htn             398 non-null    object
 20  dm              398 non-null    object
```


You are signed in as 191305
Kidney_Disease_Prediction.ipynb
colab.research.google.com/drive/1a9wR892SFwZ64kSgNXYKl6zPKmXvyQam
Gmail
YouTube
Maps
Solve C | HackerRank
Kidney_Disease_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on September 23
Comment
Share
+ Code
+ Text
Connect
Editing
63°F
Sunny
ENG
IN
13:33
17-11-2022

```
from sklearn.ensemble import RandomForestClassifier
```

```
model = RandomForestClassifier(n_estimators = 20)
model.fit(X_train, y_train)
```

```
RandomForestClassifier(n_estimators=20)
```

```
from sklearn.metrics import confusion_matrix, accuracy_score
```

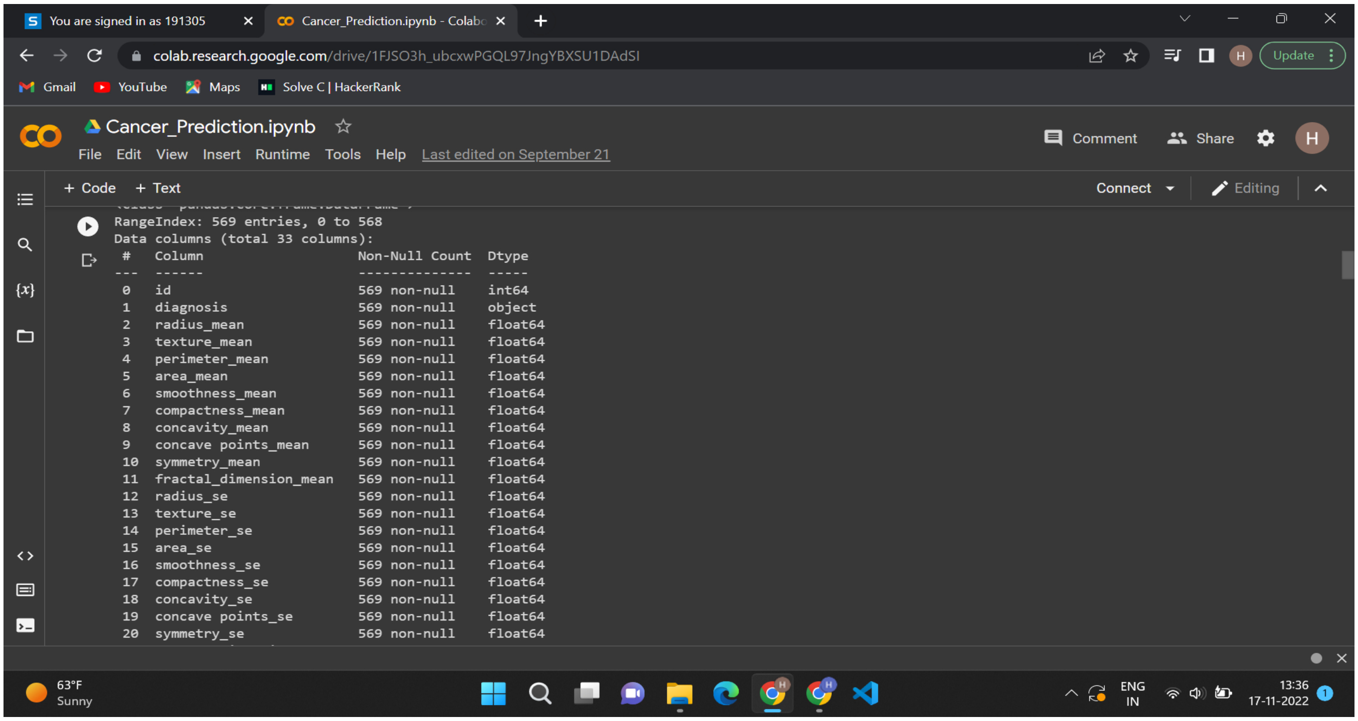
```
confusion_matrix(y_test, model.predict(X_test))
```

```
array([[23,  0],
       [ 0,  9]])
```

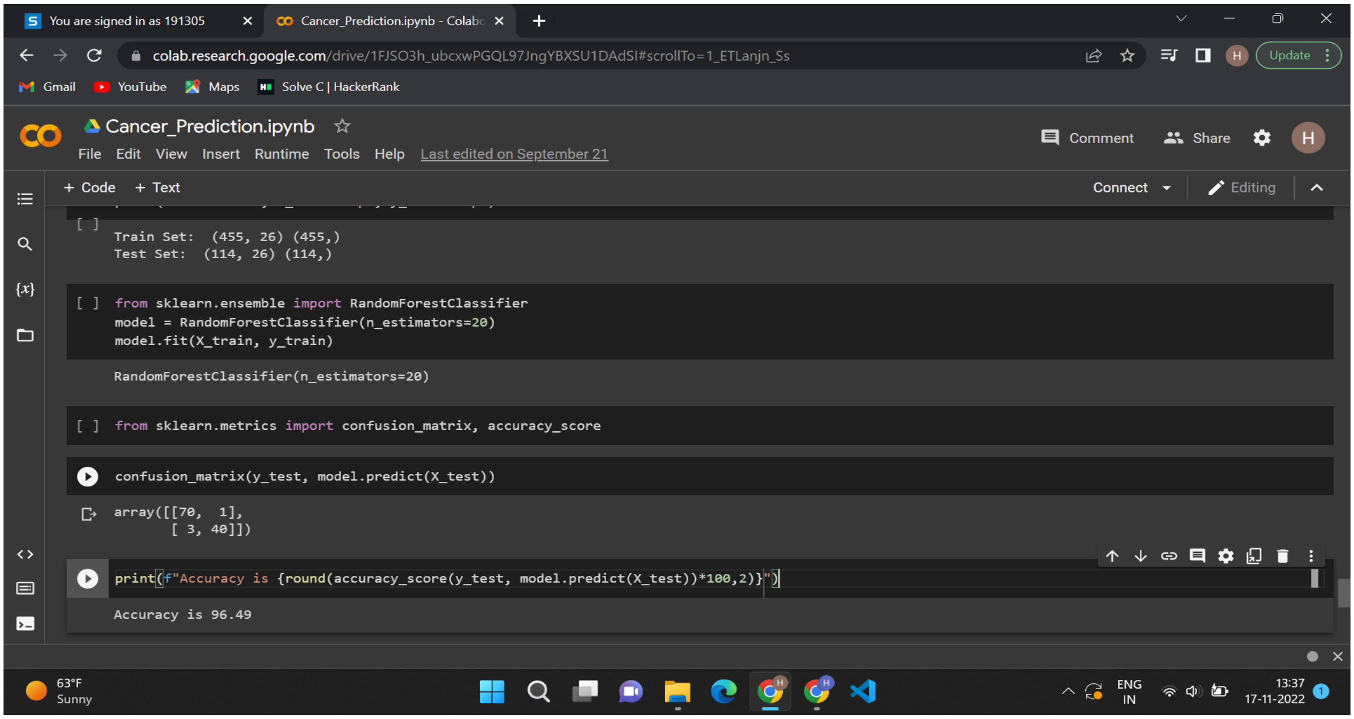
```
print(f"Accuracy is {round(accuracy_score(y_test, model.predict(X_test))*100, 2)}%")
```

```
Accuracy is 100.0%
```

```
import pickle
pickle.dump(model, open('kidney.pkl', 'wb'))
```



```
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   id                       569 non-null    int64  
 1   diagnosis                569 non-null    object 
 2   radius_mean              569 non-null    float64
 3   texture_mean             569 non-null    float64
 4   perimeter_mean           569 non-null    float64
 5   area_mean                569 non-null    float64
 6   smoothness_mean          569 non-null    float64
 7   compactness_mean         569 non-null    float64
 8   concavity_mean           569 non-null    float64
 9   concave points_mean      569 non-null    float64
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
 19  concave points_se        569 non-null    float64
 20  symmetry_se              569 non-null    float64
```



You are signed in as 191305
Cancer_Prediction.ipynb - Colab
colab.research.google.com/drive/1FJSO3h_ubcxwPGQL97JngYBXSU1DAdSI#scrollTo=1_ETLanjn_Ss
Gmail
YouTube
Maps
Solve C | HackerRank
Update
Cancer_Prediction.ipynb
File
Edit
View
Insert
Runtime
Tools
Help
Last edited on September 21
Comment
Share
+ Code
+ Text
Connect
Editing
63°F
Sunny
ENG
IN
13:37
17-11-2022

```
Train Set:  (455, 26) (455,)
Test Set:  (114, 26) (114,)
```

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=20)
model.fit(X_train, y_train)
```

```
RandomForestClassifier(n_estimators=20)
```

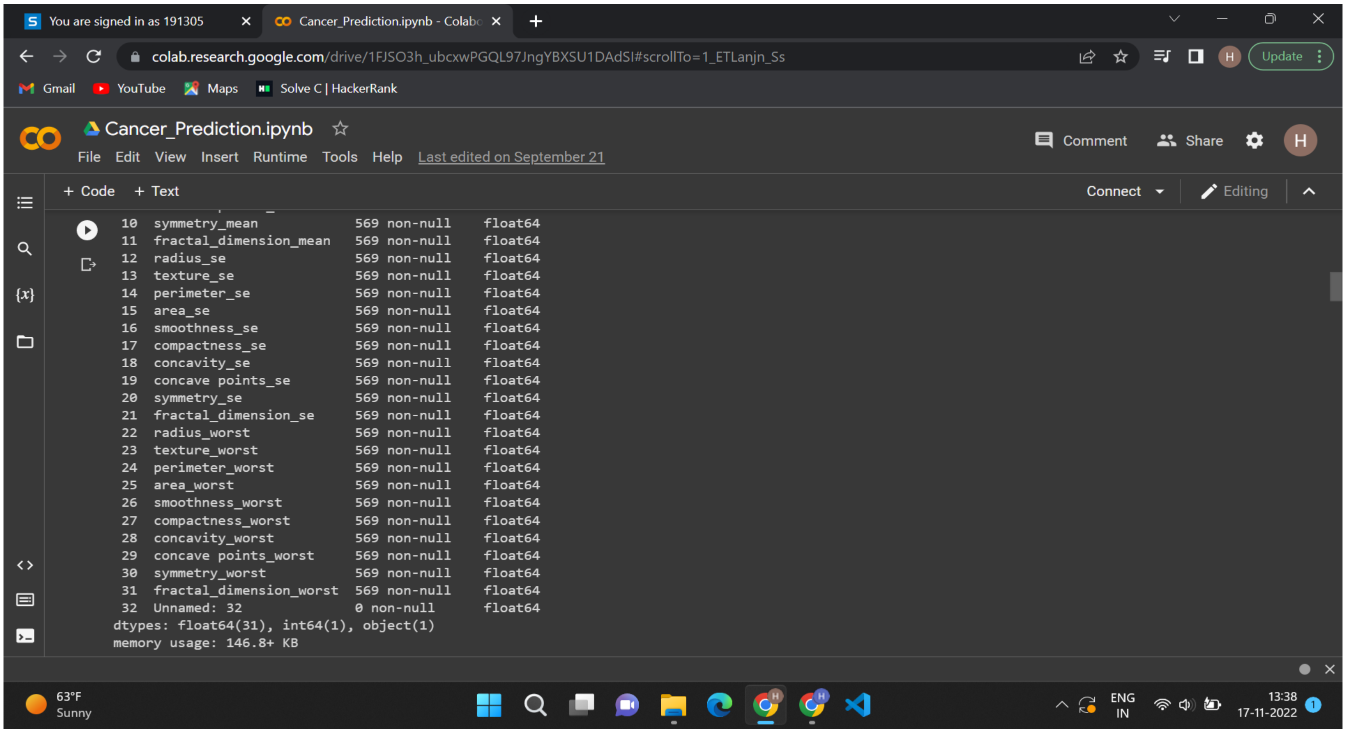
```
from sklearn.metrics import confusion_matrix, accuracy_score
```

```
confusion_matrix(y_test, model.predict(X_test))
```

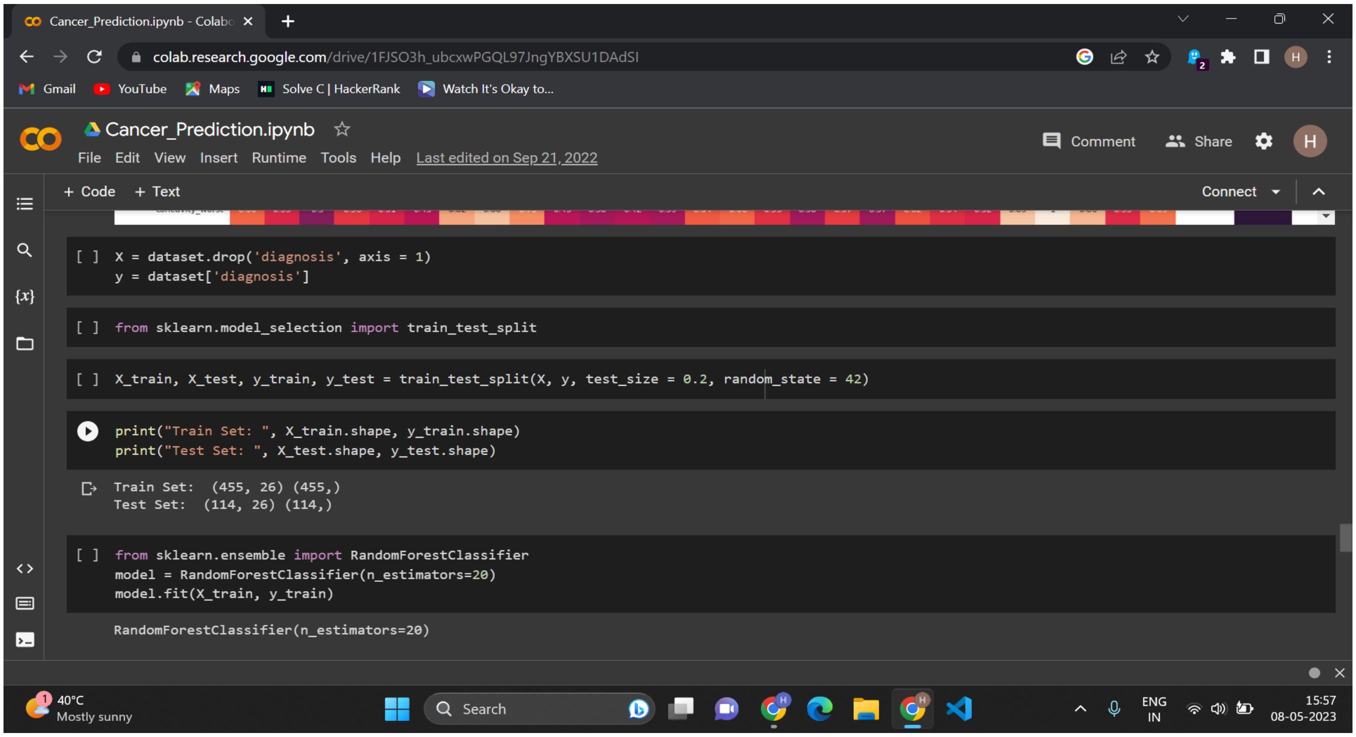
```
array([[70,  1],
       [ 3, 40]])
```

```
print(f"Accuracy is {round(accuracy_score(y_test, model.predict(X_test))*100,2)}")
```

```
Accuracy is 96.49
```


You are signed in as 191305
Cancer_Prediction.ipynb - Colab
colab.research.google.com/drive/1FJSO3h_ubcxwPGQL97JngYBXSU1DAdSI#scrollTo=1_ETLanjn_Ss
Gmail
YouTube
Maps
Solve C | HackerRank
Update
Cancer_Prediction.ipynb
File
Edit
View
Insert
Runtime
Tools
Help
Last edited on September 21
Comment
Share
+ Code
+ Text
Connect
Editing
63°F
Sunny
ENG
IN
13:38
17-11-2022

```
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
 19  concave points_se        569 non-null    float64
 20  symmetry_se              569 non-null    float64
 21  fractal_dimension_se     569 non-null    float64
 22  radius_worst             569 non-null    float64
 23  texture_worst            569 non-null    float64
 24  perimeter_worst          569 non-null    float64
 25  area_worst               569 non-null    float64
 26  smoothness_worst         569 non-null    float64
 27  compactness_worst        569 non-null    float64
 28  concavity_worst          569 non-null    float64
 29  concave points_worst     569 non-null    float64
 30  symmetry_worst           569 non-null    float64
 31  fractal_dimension_worst  569 non-null    float64
 32  Unnamed: 32              0 non-null      float64
dtypes: float64(31), int64(1), object(1)
memory usage: 146.8+ KB
```



```
[ ] X = dataset.drop('diagnosis', axis = 1)
    y = dataset['diagnosis']
```

```
[ ] from sklearn.model_selection import train_test_split
```

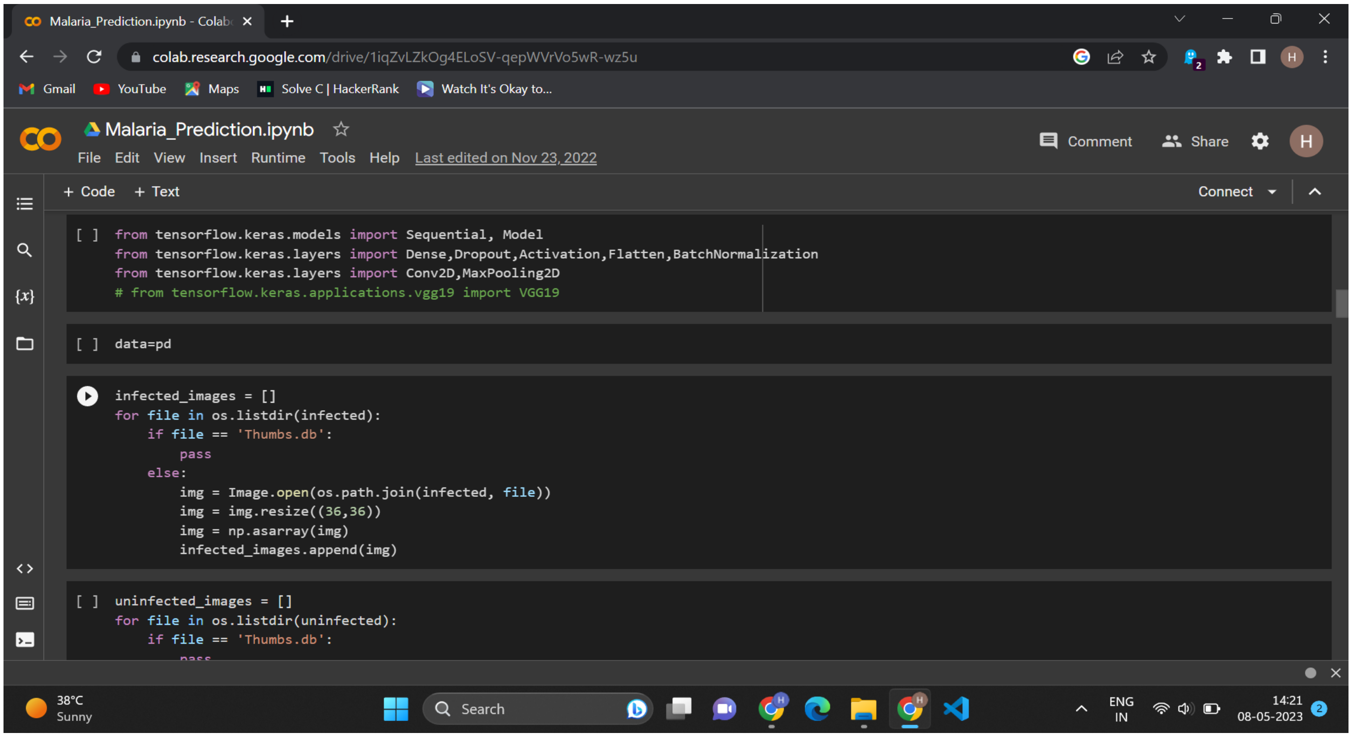
```
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 42)
```

```
print("Train Set: ", X_train.shape, y_train.shape)
print("Test Set: ", X_test.shape, y_test.shape)
```

```
Train Set:  (455, 26) (455,)
Test Set:  (114, 26) (114,)
```

```
[ ] from sklearn.ensemble import RandomForestClassifier
    model = RandomForestClassifier(n_estimators=20)
    model.fit(X_train, y_train)
```

```
RandomForestClassifier(n_estimators=20)
```

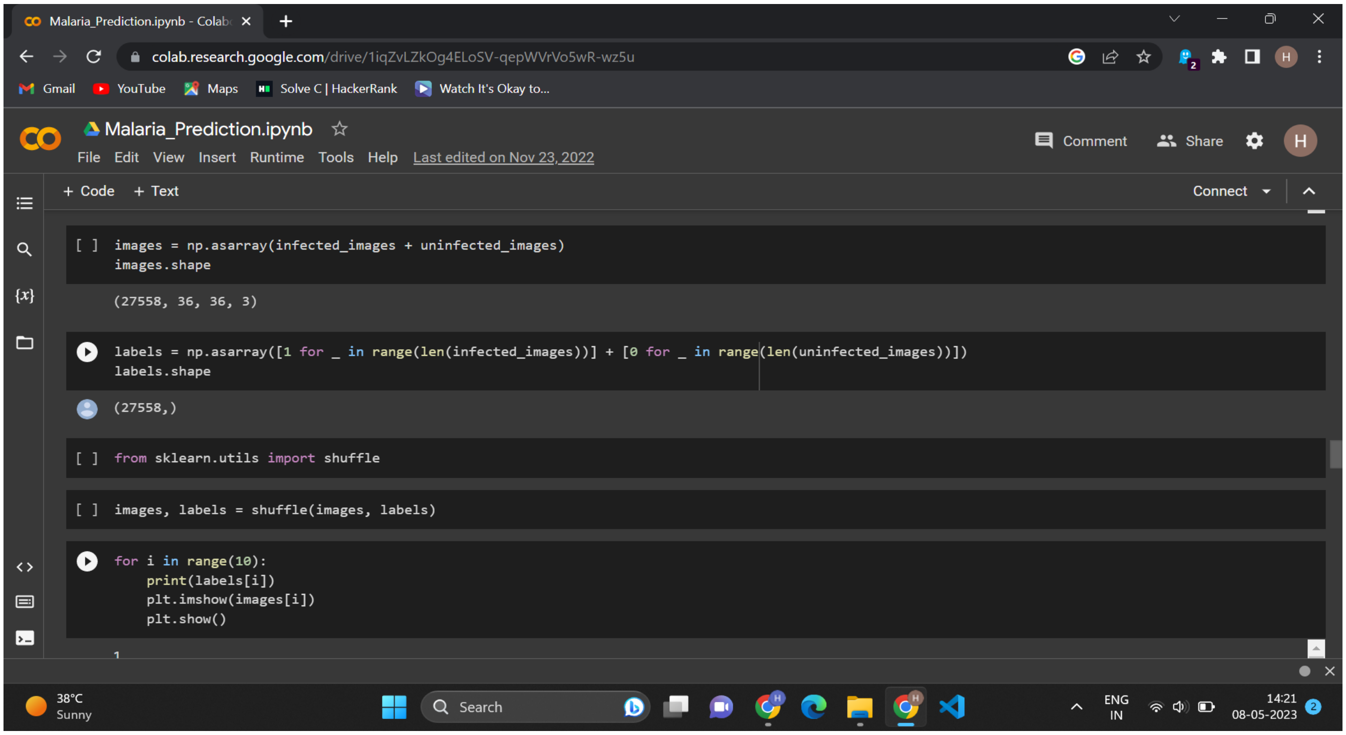

Malaria_Prediction.ipynb - Colab
colab.research.google.com/drive/1iqZvLZkOg4ELoSV-qepWVrVo5wR-wz5u
Gmail
YouTube
Maps
Solve C | HackerRank
Watch It's Okay to...
Malaria_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on Nov 23, 2022
Comment
Share
+ Code + Text
Connect
38°C
Sunny
Search
ENG
IN
14:21
08-05-2023

```
[ ] from tensorflow.keras.models import Sequential, Model
    from tensorflow.keras.layers import Dense,Dropout,Activation,Flatten,BatchNormalization
    from tensorflow.keras.layers import Conv2D,MaxPooling2D
    # from tensorflow.keras.applications.vgg19 import VGG19
```

```
[ ] data=pd
```

```
infected_images = []
for file in os.listdir(infected):
    if file == 'Thumbs.db':
        pass
    else:
        img = Image.open(os.path.join(infected, file))
        img = img.resize((36,36))
        img = np.asarray(img)
        infected_images.append(img)
```

```
[ ] uninfected_images = []
    for file in os.listdir(uninfected):
        if file == 'Thumbs.db':
```


Malaria_Prediction.ipynb - Colab
colab.research.google.com/drive/1iqZvLZkOg4ELoSV-qepWVrVo5wR-wz5u
Gmail
YouTube
Maps
Solve C | HackerRank
Watch It's Okay to...
Malaria_Prediction.ipynb
File Edit View Insert Runtime Tools Help Last edited on Nov 23, 2022
Comment
Share
+ Code + Text
Connect
38°C
Sunny
Search
ENG
IN
14:21
08-05-2023

```
[ ] images = np.asarray(infected_images + uninfected_images)
    images.shape

    (27558, 36, 36, 3)
```

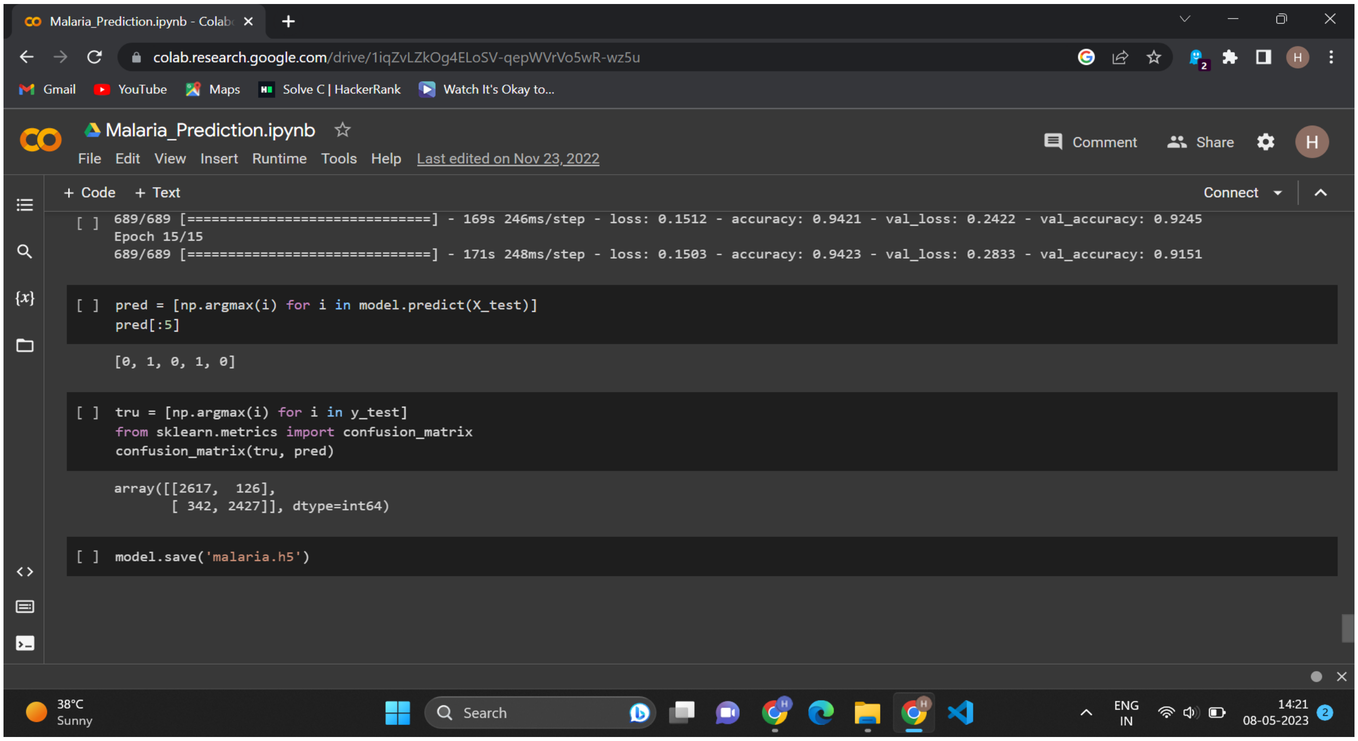
```
labels = np.asarray([1 for _ in range(len(infected_images))] + [0 for _ in range(len(uninfected_images))])
labels.shape

(27558,)
```

```
[ ] from sklearn.utils import shuffle
```

```
[ ] images, labels = shuffle(images, labels)
```

```
for i in range(10):
    print(labels[i])
    plt.imshow(images[i])
    plt.show()
```


Malaria_Prediction.ipynb - Colab
colab.research.google.com/drive/1iqZvLZkOg4ELoSV-qepWVrVo5wR-wz5u
Gmail
YouTube
Maps
Solve C | HackerRank
Watch It's Okay to...
Malaria_Prediction.ipynb
File Edit View Insert Runtime Tools Help
Last edited on Nov 23, 2022
Comment
Share
+ Code
+ Text
Connect
38°C
Sunny
Search
ENG
IN
14:21
08-05-2023

```
689/689 [==============================] - 169s 246ms/step - loss: 0.1512 - accuracy: 0.9421 - val_loss: 0.2422 - val_accuracy: 0.9245
Epoch 15/15
689/689 [==============================] - 171s 248ms/step - loss: 0.1503 - accuracy: 0.9423 - val_loss: 0.2833 - val_accuracy: 0.9151
```

```
pred = [np.argmax(i) for i in model.predict(X_test)]
pred[:5]
```

```
[0, 1, 0, 1, 0]
```

```
tru = [np.argmax(i) for i in y_test]
from sklearn.metrics import confusion_matrix
confusion_matrix(tru, pred)
```

```
array([[2617,  126],
       [ 342, 2427]], dtype=int64)
```

```
model.save('malaria.h5')
```