Data Science project report on “DIABETES RECOGNITION”

SUBMITTED BY,
NISHMITHA

TABLE OF CONTENTS:

Chapter 1:

ABSTRACT

Many of the interesting and important applications of machine learning are seen in a medical organization. The notion of machine learning has swiftly become very appealing to healthcare industries. The predictions and analysis made by the research community for medical dataset support the people by taking proper care and precautions by preventing diseases. Through a set of medical datasets, different methods are used extensively in developing the decision support systems for disease prediction. This project explains various aspects of machine learning, the types of algorithms which can help in decision making and prediction and also various applications of machine learning in the field of medicine focusing on the prediction of diabetes through machine learning. Diabetes is one of the most increasing diseases in the world and it requires continuous monitoring. To check this, we explore various machine learning algorithms which will help in early prediction of these disease.

Diabetes is a metabolic disease affecting a multitude of people worldwide. Its incidence rates are increasing alarmingly every year. If untreated, diabetes-related complications in many vital organs of the body may turn fatal. Early detection of diabetes is very important for timely treatment which can stop the disease progressing to such complications. RR-interval signals known as heart rate variability (HRV) signals (derived from electrocardiogram (ECG) signals) can be effectively used for the non-invasive detection of diabetes. This research paper presents a methodology for classification of diabetic and normal HRV signals using deep learning architectures. We employ long short-term memory (LSTM), convolutional neural network (CNN) and its combinations for extracting complex temporal dynamic features of the input HRV data. These features are passed into support vector machine (SVM) for classification. We have obtained the performance improvement of 0.03% and 0.06% in CNN and CNN-LSTM architecture respectively compared to our earlier work without using SVM. The classification system proposed can help the clinicians to diagnose diabetes using ECG signals with a very high accuracy of 95.7%.

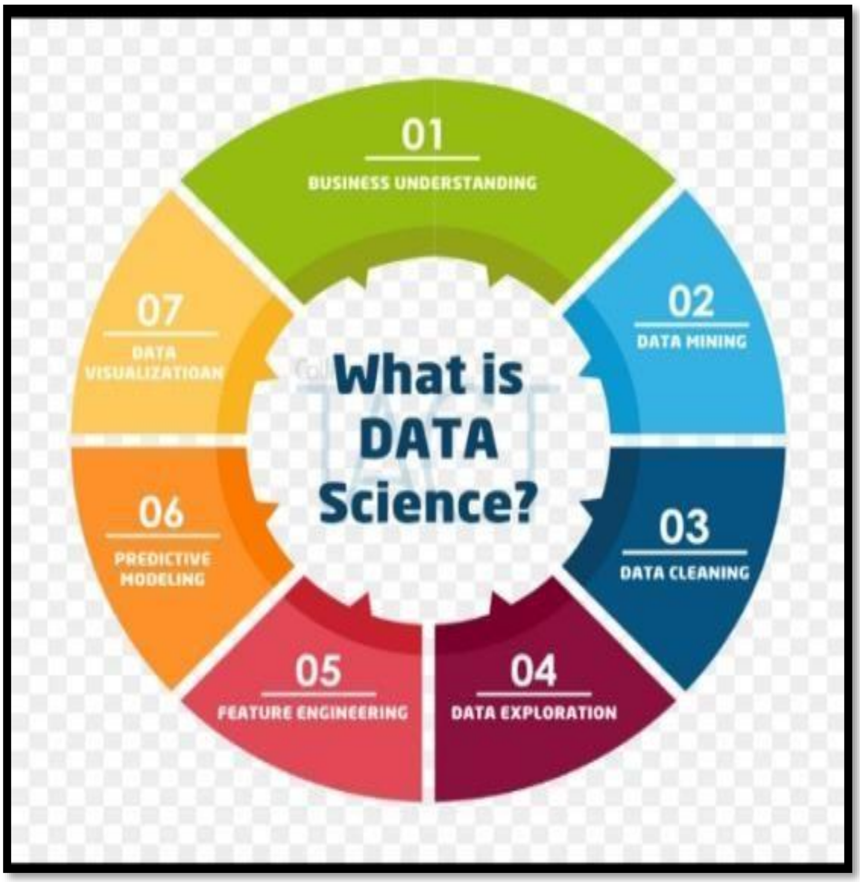
Chapter 2:

INTRODUCTION

2.1 DATA SCIENCE:

Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and to extract knowledge and insights from noisy, structured and unstructured data, and apply knowledge and actionable insights from data across a broad range of application domains. Data science is related to data mining, machine learning and big data.

Data science is a "concept to unify statistics, data analysis, informatics, and their related methods" in order to "understand and analyse actual phenomena" with data. It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, information science, and domain knowledge. However, data science is different from computer science and information science. Turing Award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational, and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.



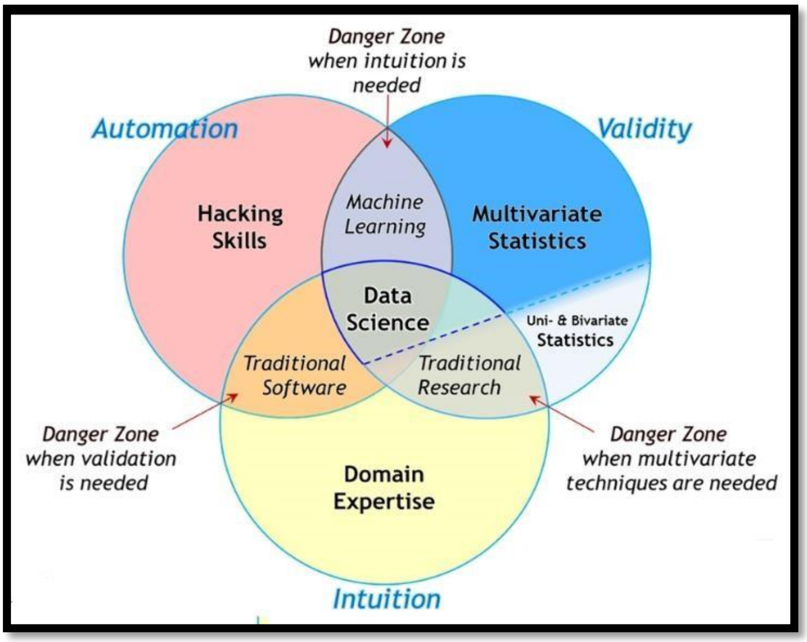
A data scientist is someone who creates programming code and combines it with statistical knowledge to create insights from data. Data scientists also rely heavily on artificial intelligence, especially its subfields of machine learning and deep learning, to create models and make predictions using algorithms and other techniques.

How Does Data Science Work?

Data science involves a plethora of disciplines and expertise areas to produce a holistic, through and refined look into raw data. Data scientists must be skilled in everything from data engineering, math, statistics, advanced computing and visualizations to be able to effectively sift through muddled masses of information and communicate only the most vital bits that will help drive innovation and efficiency.

WHAT CAN DATA SCIENCE USED FOR?

- Automation and decision-making (background checks, credit worthiness, etc.)
- Classifications (in an email server, this could mean classifying emails as “important” or “junk”)
- Forecasting (sales, revenue and customer retention)
- Pattern detection (weather patterns, financial market patterns, etc.)
- Recognition (facial, voice, text, etc.)
- Recommendations (based on learned preferences, recommendation engines can refer you to movies, restaurant and books you may like)


Danger Zone when intuition is needed
Automation
Validity
Hacking Skills
Machine Learning
Multivariate Statistics
Data Science
Uni- & Bivariate Statistics
Traditional Software
Traditional Research
Danger Zone when validation is needed
Danger Zone when multivariate techniques are needed
Domain Expertise
Intuition

2.2 MACHINE LEARNING

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.

Why is machine learning important?

Machine learning is important because it gives enterprises a view of trends in customer behaviour and business operational patterns, as well as supports the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning a central part of their operations. Machine learning has become a significant competitive differentiator for many companies.

What are the different types of machine learning?

Classical machine learning is often categorized by how an algorithm learns to become more accurate in its predictions. There are four basic approaches: supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The type of algorithm data scientists chose to use depends on what type of data they want to predict.

- **Supervised learning:** In this type of machine learning, data scientists supply algorithms with labelled training data and define the variables they want the algorithm to assess for correlations. Both the input and the output of the algorithm is specified.
- **Unsupervised learning:** This type of machine learning involves algorithms that train on unlabelled data. The algorithm scans through data sets looking for any meaningful connection. The data that algorithms train on as well as the predictions or recommendations they output are predetermined.
- **Semi-supervised learning:** This approach to machine learning involves a mix of the two preceding types. Data scientists may feed an algorithm mostly labelled training data, but the model is free to explore the data on its own and develop its own understanding of the data set.
- **Reinforcement learning:** Data scientists typically use reinforcement learning to teach a machine to complete a multi-step process for which there are clearly defined

rules. Data scientists program an algorithm to complete a task and give it positive or negative cues as it works out how to complete a task. But for the most part, the algorithm decides on its own what steps to take along the way.

How does supervised machine learning work?

Supervised machine learning requires the data scientist to train the algorithm with both labelled inputs and desired outputs. Supervised learning algorithms are good for the following tasks:

- **Binary classification:** Dividing data into two categories.
- **Multi-class classification:** Choosing between more than two types of answers.
- **Regression modelling:** Predicting continuous values.
- **Assembling:** Combining the predictions of multiple machine learning models to produce an accurate prediction.

How does unsupervised machine learning work?

Unsupervised machine learning algorithms do not require data to be labelled. They sift through unlabelled data to look for patterns that can be used to group data points into subsets. Most types of deep learning, including neural networks, are unsupervised algorithms. Unsupervised learning algorithms are good for the following tasks:

- **Clustering:** Splitting the dataset into groups based on similarity.
- **Anomaly detection:** Identifying unusual data points in a data set.
- **Association mining:** Identifying sets of items in a data set that frequently occur together.
- **Dimensionality reduction:** Reducing the number of variables in a data set.

How does semi-supervised learning work?

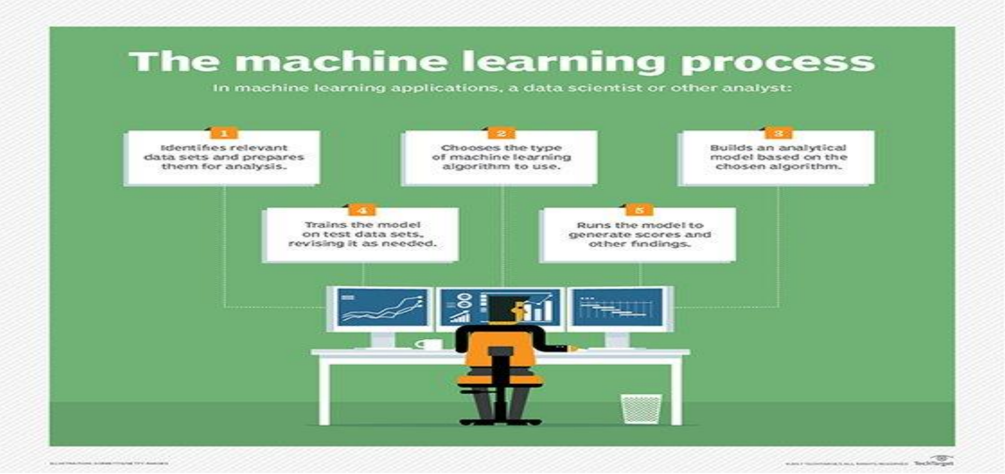
Semi-supervised learning works by data scientists feeding a small amount of labelled training data to an algorithm. From this, the algorithm learns the dimensions of the data set, which it can then apply to new, unlabelled data. The performance of algorithms typically improves when they train on labelled data sets. But labelling data can be time consuming and expensive. Semi-supervised learning strikes a middle ground between the performance of supervised learning and the efficiency of unsupervised learning. Some areas where semi-supervised learning is used include:

- **Machine translation:** Teaching algorithms to translate language based on less than a full dictionary of words.
- **Fraud detection:** Identifying cases of fraud when you only have a few positive examples.
- **Labelling data:** Algorithms trained on small data sets can learn to apply data labels to larger sets automatically.

How does reinforcement learning work?

Reinforcement learning works by programming an algorithm with a distinct goal and a prescribed set of rules for accomplishing that goal. Data scientists also program the algorithm to seek positive rewards -- which it receives when it performs an action that is beneficial toward the ultimate goal -- and avoid punishments -- which it receives when it performs an action that gets it farther away from its ultimate goal. Reinforcement learning is often used in areas such as:

- **Robotics:** Robots can learn to perform tasks the physical world using this technique.
- **Video gameplay:** Reinforcement learning has been used to teach bots to play a number of video games.
- **Resource management:** Given finite resources and a defined goal, reinforcement learning can help enterprises plan out how to allocate resources.



The machine learning process

In machine learning applications, a data scientist or other analyst:

1 Identifies relevant data sets and prepares them for analysis.

2 Chooses the type of machine learning algorithm to use.

3 Builds an analytical model based on the chosen algorithm.

4 Trains the model on test data sets, revising it as needed.

5 Runs the model to generate scores and other findings.

2.3 DIABETES MELLITUS

Diabetes Mellitus (DM), sometimes called diabetes, is a concept for a variety of disorders that include how the body converts food into energy. Once one consumes food, the body converts it into sugar named glucose and transfers it to the bloodstream. The pancreas produces insulin, which is a hormone that tends to transfer glucose from the blood to the cells that utilize it for energy.

If you have DM and don't seek medication, the body doesn't produce insulin as it does. Very much glucose persists in the body, a disorder commonly called high blood sugar. It may trigger severe or life-threatening health issues. DM develops in various ways, depending on the source

Prediabetes

Prediabetes occurs whenever the blood sugar rises higher than it should be, but still not strong enough for the doctor to recognize diabetes. With prediabetes, the risk of type-II diabetes and cardiac disease will increase. Exercising further and reducing excess weight, sometimes less than 5%−7% of the body weight, will reduce these dangers.

Type-I diabetes

Type-I diabetes is often denoted to as insulin-dependent DM. This also named juvenile-onset DM, as it frequently occurs in infancy. Diabetes with type-I is an autoimmune condition. This occurs as the body threatens the pancreas with antibodies. The organ is weakened and does not produce insulin. Body genes can cause this sort of diabetes. This may also happen due to complications with cells in the pancreas that produce insulin. Many of the health issues that might happen with the type-I result due to disruption to narrow blood vessels in the kidneys (diabetic nephropathy), eyes (termed diabetic retinopathy), and nerves (diabetic neuropathy). Anyone of type-I is often at greater risk of heart failure and stroke.

Type-II diabetes

Type-II diabetes has been known to be named non-insulin-dependent or adult-onset diabetes. Yet it has been prevalent in children and teenagers in the last 20 years, mainly as more young people are obese or overweight. Approximately 90% of patients have type-II diabetes

The pancreas normally releases some insulin while you have type-II diabetes. Yet either that isn't enough, or the body doesn't utilize it as it would. Type-II DM is also relatively mild than type-I diabetes. Yet it may also cause significant health problems, particularly in the small blood vessels in nerves, kidneys, and eyes. Type-II also raises the chances of stroke and heart failure.

Those who are overweight — more than 20% above their ideal body weight due to their height — have a very elevated chance of type-II diabetes and the health complications that may occur. Obesity also induces insulin resistance, and the pancreas has to function harder to produce more insulin.

Other forms of diabetes

The origin could be other factors in 1% to 5% of individuals who have diabetes. Those have pancreatic disorders, other operations and drugs, and illnesses. Under these situations, the doctor may want to keep an eye on the blood sugar levels.

While comprehensive DM research has generated considerable information over the previous few decades on a) etiopathology (cellular mechanisms and environmental or genetic causes), b) diagnosis and c) disease detection, diagnosis, and control, more remains to be found, unfolded, explained and demarcated. In this attempt, depending on a huge, fast and increasingly growing body of clinical evidence and research helps to provide a substantial base for effective evaluation and follow-up. ML and AI, therefore, appear as core technologies, leading significantly to clinical decision-making. The goal is therefore to connect the assessment of the data with the treatment and the smart decision-making in the implementation and application of drugs.

Chapter 3:

IMPLEMENTATION

3.1 SOFTWARE REQUIREMENTS

The following software was used for the implementation of the system:

- Windows 11
- Python 3.10.2
- Jupyter notebook
  - ❖ Pandas(1.3.4)
  - ❖ Scikit learn(0.24.2)

```
I Python          : 7.29.0
ipykernel        : 6.4.1
ipywidgets       : 7.6.5
jupyter_client   : 6.1.12
jupyter_core     : 4.8.1
jupyter_server   : 1.4.1
jupyterlab       : 3.2.1
nbclient         : 0.5.3
nbconvert        : 6.1.0
nbformat         : 5.1.3
notebook         : 6.4.5
qtconsole        : 5.1.1
traitlets        : 5.1.0
```

3.2 METHODOLOGY

This section explains the stepwise procedure used to analyse the data and to predict the data accuracy for prediction of diabetes. The system consists of the following main steps:

I have selected a diabetic dataset which consists of 768 instances classified into two classes : diabetic and non-diabetic with eight different risk factors: number of times pregnant , plasma glucose concentration of two hours in an oral glucose tolerance test , diastolic blood pressure, triceps skin fold thickness, two – hour serum insulin , body mass index , diabetes pedigree function and age.

Feature Selection is the procedure where we automatically or manually select those features which contribute most to your prediction variable or output you are interested in. If there are irrelevant features in our data then it can decrease the accuracy of the models.

1. We are taking a diabetic dataset.

2. For pre-processing step, the system uses Feature selection method : Forward feature selection and Backward Feature selection. We train three different classifiers and decide which classifier provides high accuracy. We have used these classifiers which are Random forest classifier, Logistic Regression and K Neighbor Classifier.

3. We found Random forest classifier to be the best out of all the three classifiers in the aspects of accuracy, since it gives better accuracy.

Chapter 4:

SYSTEM DESIGN

4.1 Machine learning classifiers used in diagnosis of diabetes

The variation in glucose levels is cause of diabetes. Insulin balances the blood glucose level in the body, deficiency of which cause diabetes. For the prediction of diabetes machine learning is used, these have many steps like image pre-processing/data pre-processing followed by a feature extraction and then classification. We can use any of the mentioned machine learning classifiers to predict this disease. In the above section we have learning about many classification algorithms, we can either use any one of these to predict the disease or we can explore the techniques to use the hybrid methodology to improve the accuracy over using a single one. Currently, the researches have used a single classification algorithm and have come up to accuracy of 70 to 80% for detection of the diabetes disease. Depending on the application and nature of the dataset used we can use any classification algorithms mentioned below. As there are different applications, we cannot differentiate which of the algorithms are superior or not. Each of classifiers have its own way of working and classification. Let us discuss each of them in details.

- **Logistic Regression**: Logic regression is used for Predictive Learning Model. To determine output in this classifier, we use a statistical method to analyse the dataset. These data set can have one or more than one independent values. The output is calculated with a data in which there could be two outputs. The aim of this classification algorithm is to find the relationship between the dichotomous category and predictor variables.

- **Random Forest**: This classification algorithm is similar to ensemble learning method of classification. The regression and other tasks, work by building a group of decision trees at training data level and during the output of the class, which could be the mode of classification or prediction regression for individual trees. This classifier accuracy for decision trees practice of overfitting the training data set.

- **Nearest Neighbour**: As the name suggests the nearest neighbour algorithm is based on the nearest neighbour and this classification algorithm is supervised. It is also called ask nearest neighbour classification algorithm. A cluster of labelled points are used to understand how the other points should be labelled. For labelling a new point it checks the already labelled points which could be closest to the point to be labelled, i.e., closest to the neighbour. In this way depending on the votes of the neighbour the new point is labelled the same label which most of neighbours have. In in algorithm ‘k’ is the number of neighbours which are checked.

Other possible classifiers:

- **Naive Bayes Classifier**: This classifier can also be known as a Generative Learning Model. The classification here is based on Baye's Theorem, it assumes independent predictors. In simple words, this classifier will assume that the existence of specific features in a class is not related to the existence of any other feature. If there is dependency among the features of each other or on the presence of other features, all of these will be considered as an independent contribution to the probability of the output. This classification algorithm is very much useful to large datasets and is very easy to use.

- **Decision Trees**: This classification algorithm builds the regression models. These models are builder in form of structure which is similar to tree - a tree like structure is created by this classifier. It keeps on dividing the data set into subsets and smaller subsets which develops an associated tree, incrementally. The decision tree is finally created which has decision nodes and leaf nodes. In this tree the leaf node will have details about the classification or the decision taken for classification whereas the decision will have branches. The highest decision node which will be at the top of the tree will correspond to the root node. This will be the best predictor.

- **Neural Network**: As the name suggests this classifier has units known as neurons, which are arranged in layers that convert the input vector to relevant output. Each single neuron takes an input, this is most often a non-linear input, this is given to a function which is them passed to next layer to get the output. The input given to the first layer will act as an output for the next layer and so on, thus this classification algorithm follows a feed-forward method. But in this method, there is no feedback to the previous layer, so weighting is also given to the signals passing through the neurons and the layers, these signal then are turned into a training phase this eventually then become a network to handle any particular problem. .

- **Support vector machine (SVM)**: This is also one of the classification algorithms which is supervised and is easy to use. It can used for both classification and regression applications, but it is more famous to be used in classification applications. In this algorithm each point which is a data item is plotted in a dimensional space, this space is also known as n dimensional plane, where the 'n' represents the number of features of the data. The classification is done based on the differentiation in the classes, these classes are data set points present in different planes.

- **XG Boost**: Recently, the researches have come across an algorithm "Boost" and its usage is very useful for machine learning classification. It is very much fast and its performance is better as it is an execution of a boosted decision tree. This classification model is used to improve the performance of the model and also to improve the speed. [21] We have already learned about all the machine learning classification algorithms and approaches used to predict the disease. After doing this survey we would be proposing to use more than one classification algorithm along with any of the learning approaches which will improve the prediction accuracy of the disease by more than 80%.

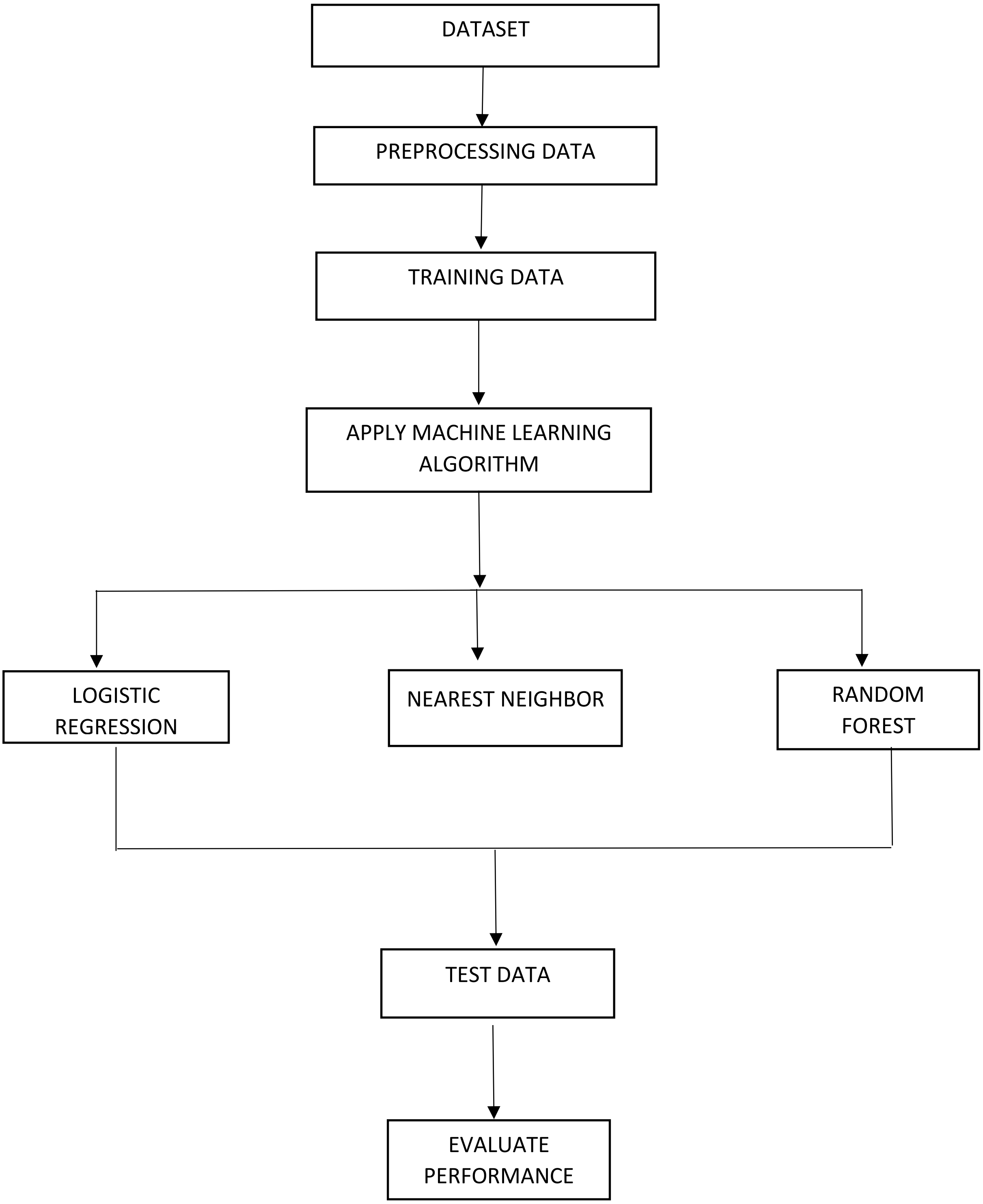
We have already learned about all the machine learning classification algorithms and approaches used to predict the disease. After doing this survey we would be proposing to use more than one classification algorithm along with any of the learning approaches which will improve the prediction accuracy of the disease by more than 80%.

It is good to use the combination of more than 2 classifiers to get the desired accuracy. We shall be using Decision tree along with other classifiers, we shall design a model to evaluate the training data We shall evaluate each of the classifier and either use XG Boost along with Decision tree/ RF/ SVM / Naive Bayes or we can use Decision Tree / RF along with the Naive Bayes.by using the combination mention in this section we shall improve the accuracy by more than 80%.

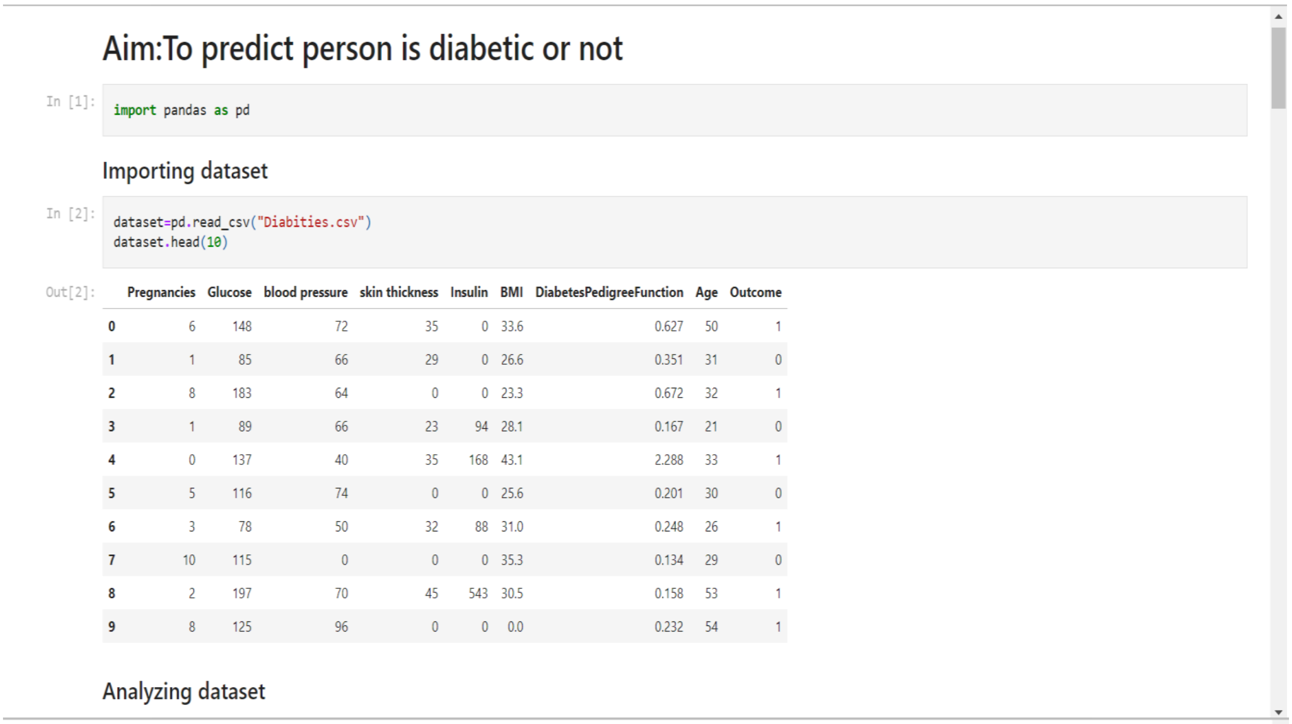
In our case we found Random forest classifier to be the best out of all the three classifiers in the aspects of accuracy, since it gives better accuracy.

4.2 PROPOSED SYSTEM

The proposed system predicts the disease of diabetes in patients with maximum accuracy. We shall talk about various machine learning, the algorithm which can help in decision making and prediction. We shall use more than one algorithm to get better accuracy of prediction.



4.3 Code



Aim:To predict person is diabetic or not

In [1]:

```
import pandas as pd
```

Importing dataset

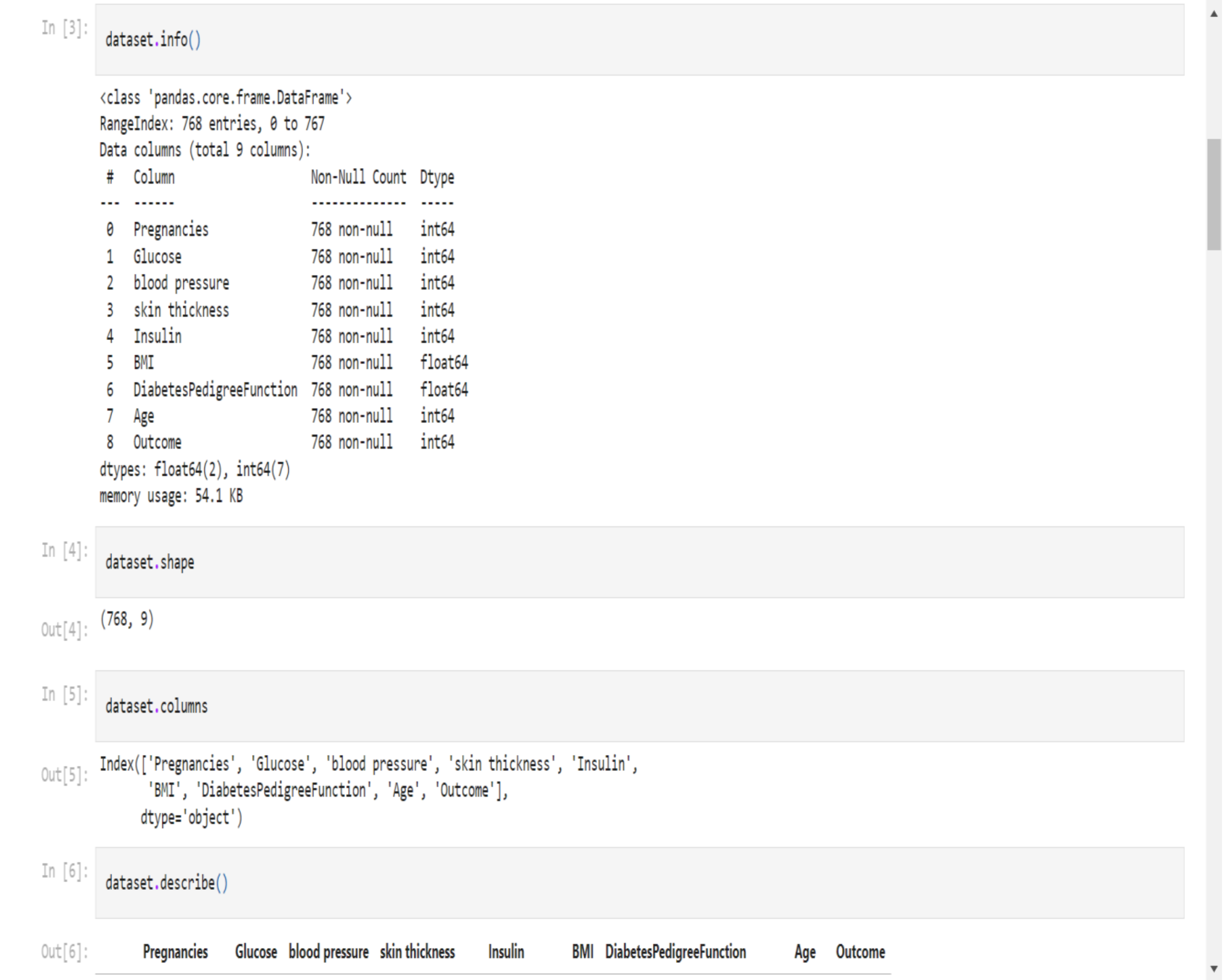
In [2]:

```
dataset=pd.read_csv("Diabities.csv")
dataset.head(10)
```

Out[2]:

	Pregnancies	Glucose	blood pressure	skin thickness	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

Analyzing dataset



In [3]:

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Pregnancies               768 non-null    int64  
 1   Glucose                   768 non-null    int64  
 2   blood pressure            768 non-null    int64  
 3   skin thickness            768 non-null    int64  
 4   Insulin                   768 non-null    int64  
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64  
 8   Outcome                   768 non-null    int64  
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

In [4]:

```
dataset.shape
```

Out[4]:

```
(768, 9)
```

In [5]:

```
dataset.columns
```

Out[5]:

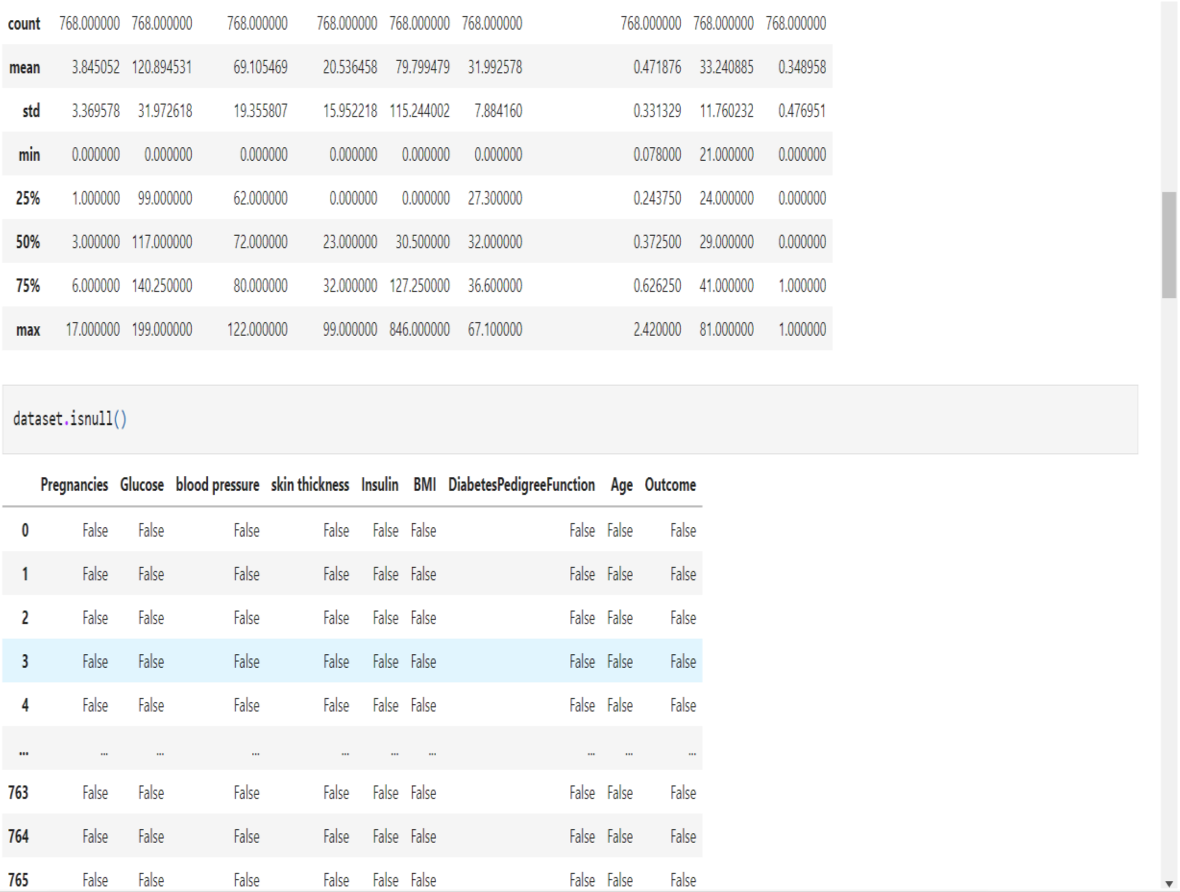
```
Index(['Pregnancies', 'Glucose', 'blood pressure', 'skin thickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')
```

In [6]:

```
dataset.describe()
```

Out[6]:

	Pregnancies	Glucose	blood pressure	skin thickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome



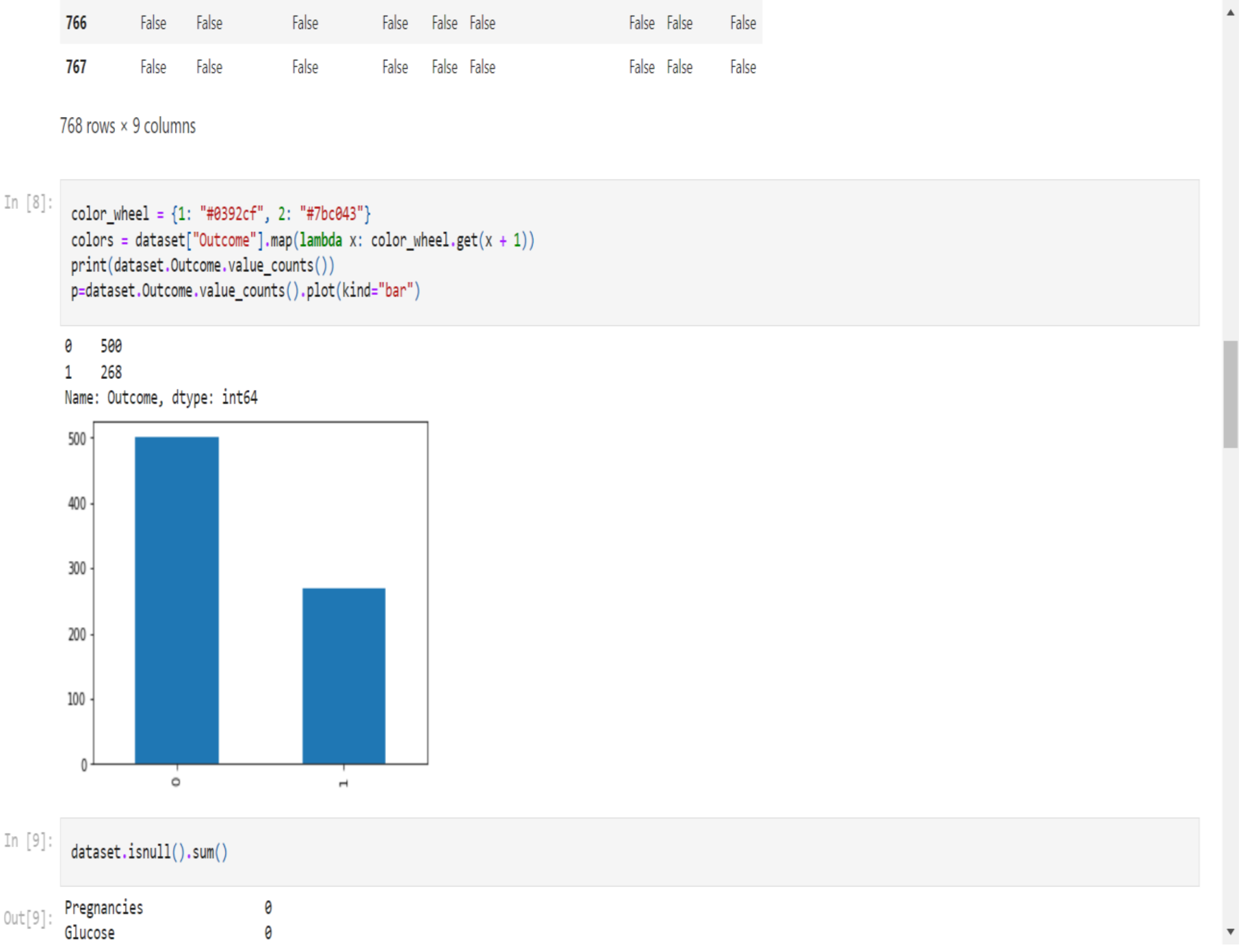
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	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

In [7]:

```
dataset.isnull()
```

Out[7]:

	Pregnancies	Glucose	blood pressure	skin thickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
763	False	False	False	False	False	False	False	False	False
764	False	False	False	False	False	False	False	False	False
765	False	False	False	False	False	False	False	False	False
766	False	False	False	False	False	False	False	False	False
767	False	False	False	False	False	False	False	False	False



768 rows × 9 columns

In [8]:

```
color_wheel = {1: "#0392cf", 2: "#7bc043"}
colors = dataset["Outcome"].map(lambda x: color_wheel.get(x + 1))
print(dataset.Outcome.value_counts())
p=dataset.Outcome.value_counts().plot(kind="bar")
```

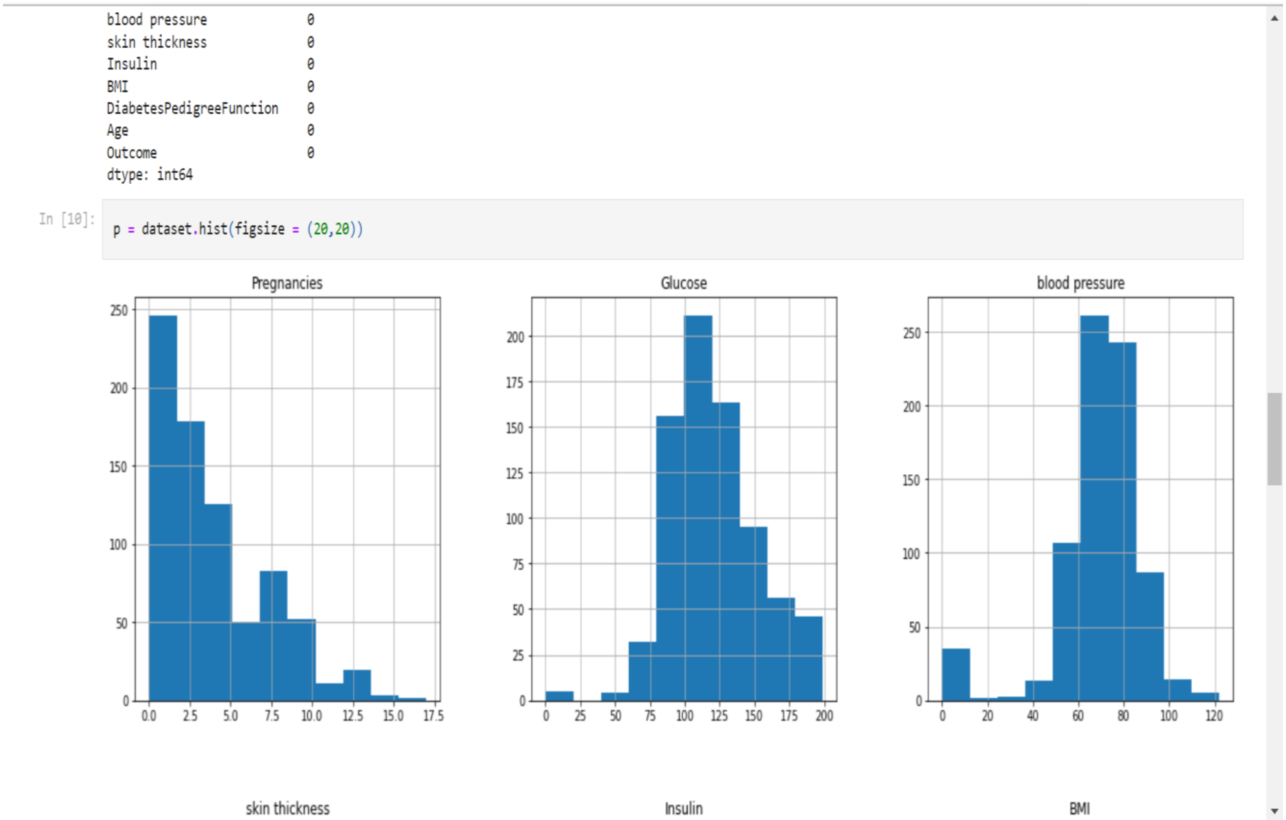
```
0    500
1    268
Name: Outcome, dtype: int64
```

In [9]:

```
dataset.isnull().sum()
```

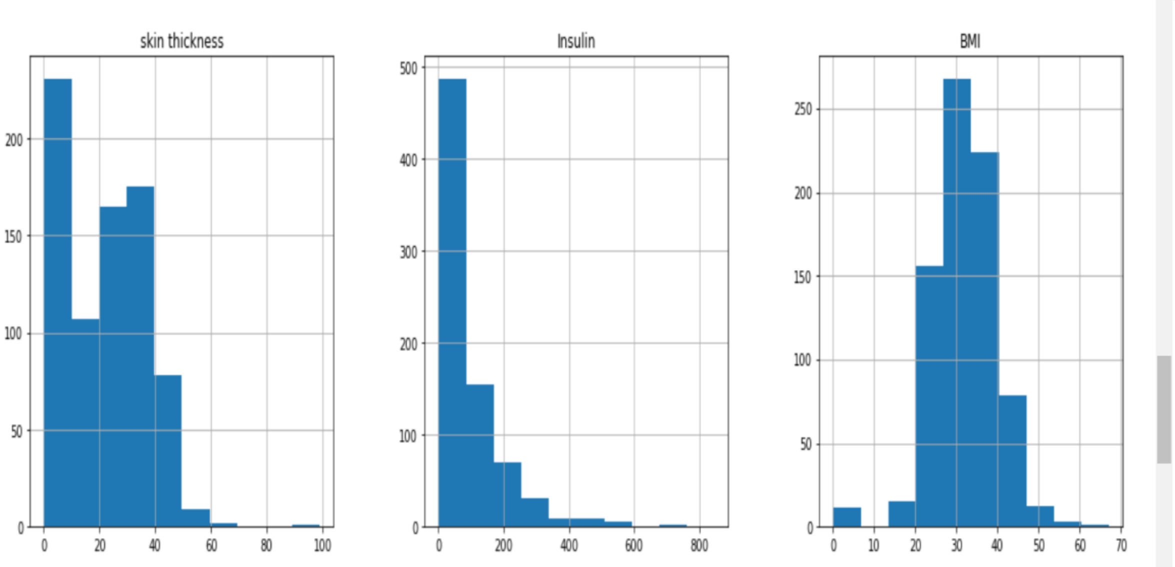
Out[9]:

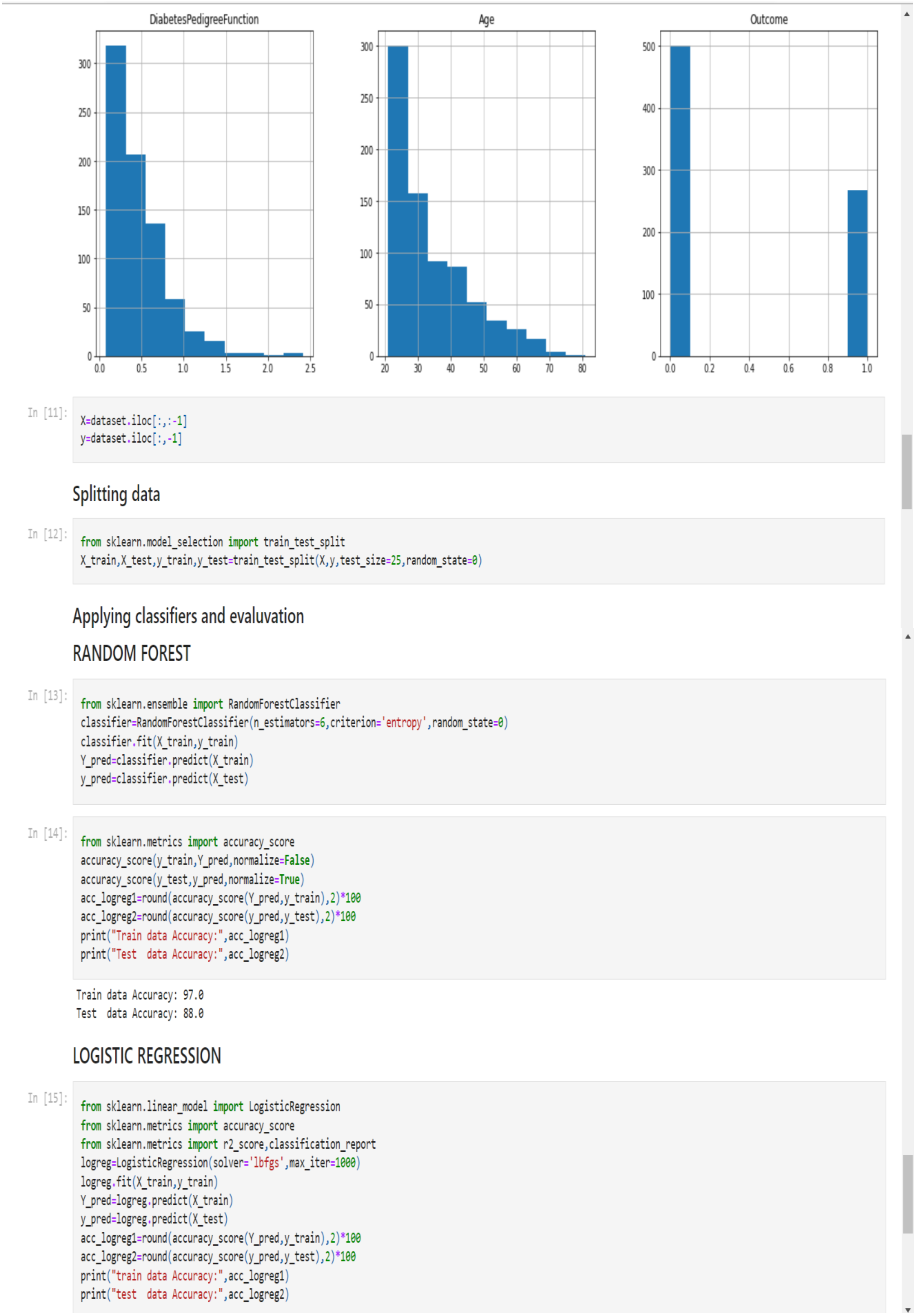
```
Pregnancies                 0
Glucose                     0
```



```
blood pressure              0
skin thickness              0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

```
In [10]: p = dataset.hist(figsize = (20,20))
```





```
X=dataset.iloc[:,:-1]
y=dataset.iloc[:,-1]
```

Splitting data

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=25,random_state=0)
```

Applying classifiers and evaluvation

RANDOM FOREST

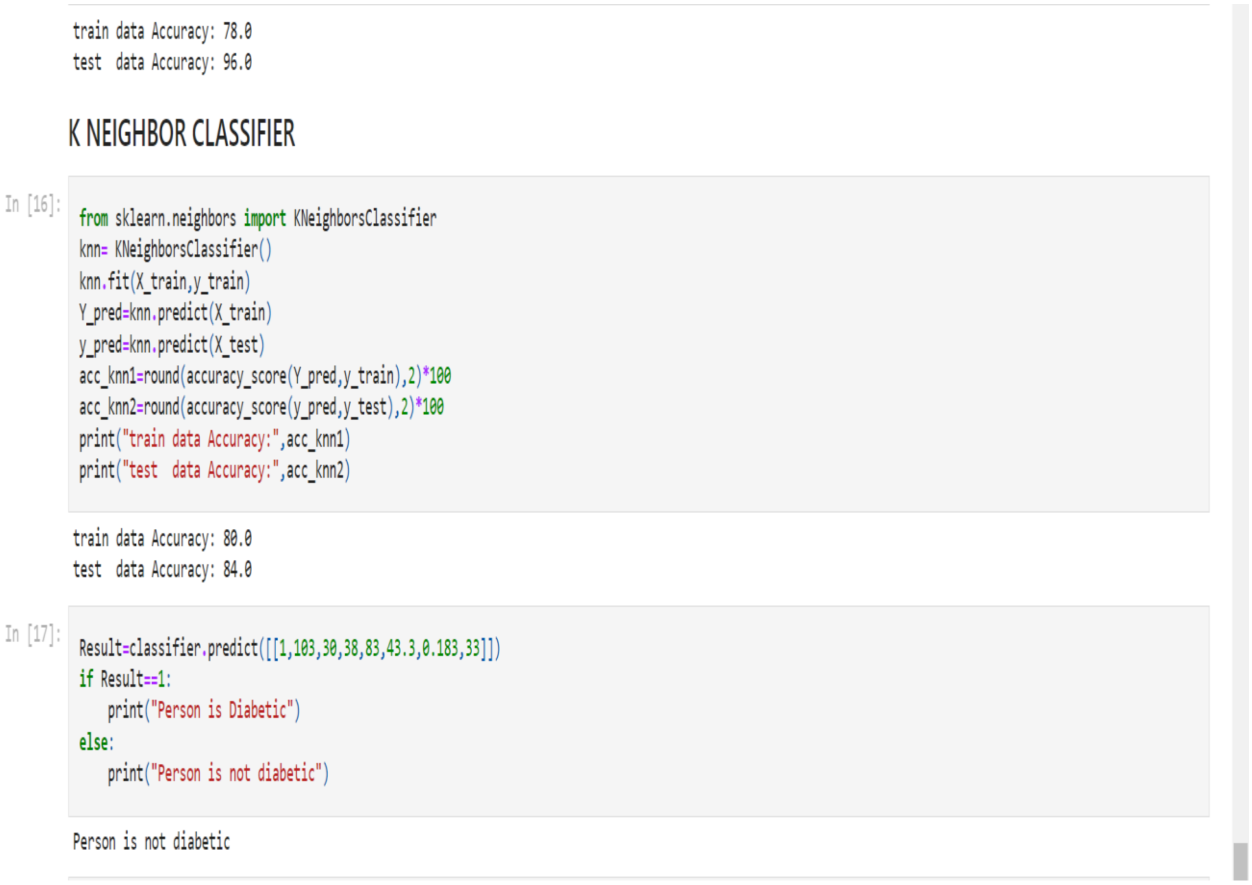
```
from sklearn.ensemble import RandomForestClassifier
classifier=RandomForestClassifier(n_estimators=6,criterion='entropy',random_state=0)
classifier.fit(X_train,y_train)
Y_pred=classifier.predict(X_train)
y_pred=classifier.predict(X_test)
```

```
from sklearn.metrics import accuracy_score
accuracy_score(y_train,Y_pred,normalize=False)
accuracy_score(y_test,y_pred,normalize=True)
acc_logreg1=round(accuracy_score(Y_pred,y_train),2)*100
acc_logreg2=round(accuracy_score(y_pred,y_test),2)*100
print("Train data Accuracy:",acc_logreg1)
print("Test  data Accuracy:",acc_logreg2)
```

```
Train data Accuracy: 97.0
Test  data Accuracy: 88.0
```

LOGISTIC REGRESSION

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.metrics import r2_score,classification_report
logreg=LogisticRegression(solver='lbfgs',max_iter=1000)
logreg.fit(X_train,y_train)
Y_pred=logreg.predict(X_train)
y_pred=logreg.predict(X_test)
acc_logreg1=round(accuracy_score(Y_pred,y_train),2)*100
acc_logreg2=round(accuracy_score(y_pred,y_test),2)*100
print("train data Accuracy:",acc_logreg1)
print("test  data Accuracy:",acc_logreg2)
```



```
train data Accuracy: 78.0
test  data Accuracy: 96.0
```

K NEIGHBOR CLASSIFIER

In [16]:

```
from sklearn.neighbors import KNeighborsClassifier
knn= KNeighborsClassifier()
knn.fit(X_train,y_train)
Y_pred=knn.predict(X_train)
y_pred=knn.predict(X_test)
acc_knn1=round(accuracy_score(Y_pred,y_train),2)*100
acc_knn2=round(accuracy_score(y_pred,y_test),2)*100
print("train data Accuracy:",acc_knn1)
print("test  data Accuracy:",acc_knn2)
```

```
train data Accuracy: 80.0
test  data Accuracy: 84.0
```

In [17]:

```
Result=classifier.predict([[1,103,30,38,83,43.3,0.183,33]])
if Result==1:
    print("Person is Diabetic")
else:
    print("Person is not diabetic")
```

```
Person is not diabetic
```

Chapter 5:

IMPORTED LIBRARIES AND ITS MODULES

5.1. PANDAS

pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.[2] The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals.[3] Its name is a play on the phrase "Python data analysis" itself.[4] Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.

Library features

- Data Frame object for data manipulation with integrated indexing.
- Tools for reading and writing data between in-memory data structures and different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of data sets.
- Label-based slicing, fancy indexing, and sub setting of large data sets.
- Data structure column insertion and deletion.
- Group by engine allowing split-apply-combine operations on data sets.
- Data set merging and joining.
- Hierarchical axis indexing to work with high-dimensional data in a lower-dimensional data structure.
- Time series-functionality: Date range generation[6] and frequency conversions, moving window statistics, moving window linear regressions, date shifting and lagging.
- Provides data filtration.

The library is highly optimized for performance, with critical code paths written in CPython or C.

5.2 SK LEARN

Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine leraning library for the python programming language.learning library . It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn is a NumFOCUS fiscally sponsored project.

The scikit-learn project started as scikits.learn, a Google Summer of Code project by French data scientist David Cournapeau. Its name stems from the notion that it is a "SciKit" (SciPy Toolkit), a separately-developed and distributed third-party extension to SciPy.[5] The original codebase was later rewritten by other developers. In 2010 Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort and Vincent Michel, all from the French Institute for Research in Computer Science and Automation in Rocquencourt, France, took leadership of the project and made the first public release on February the 1st 2010.[6] Of the various scikits, scikit-learn as well as scikit-image were described as "well-maintained and popular" in November 2012.[7] Scikit-learn is one of the most popular machine learning libraries on GitHub.

Scikit-learn is largely written in Python, and uses NumPy extensively for high-performance linear algebra and array operations. Furthermore, some core algorithms are written in Cython to improve performance. Support vector machines are implemented by a Cython wrapper around LIBSVM; logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR. In such cases, extending these methods with Python may not be possible.

Scikit-learn integrates well with many other Python libraries, such as Matplotlib and plotly for plotting, NumPy for array vectorization, Pandas data frames, SciPy, and many more.

5.2.1 sklearn.model_selection

It is a Python library that offers various features for data processing that can be used for classification, clustering, and model selection. Model_selection is a method for setting a blueprint to analyze data and then using it to measure new data

5.2.2 sklearn.ensemble

The sklearn. ensemble module includes two averaging algorithms based on randomized decision trees: the Random Forest algorithm and the Extra-Trees method. Both algorithms are perturb-and-combine techniques [B1998] specifically designed for trees.

5.2.3 sklearn.metrix

The sklearn. metrics module implements several loss, score, and utility functions to measure classification performance. Some metrics might require probability estimates of the positive class, confidence values, or binary decisions values.

5.2.4 sklearn. Linear_model

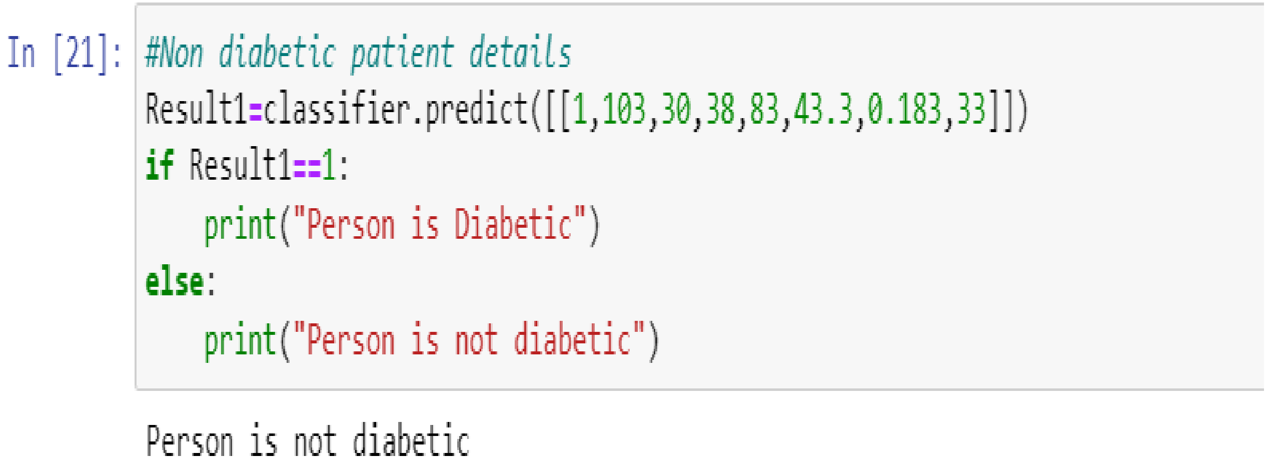
linear_model is a class of the sklearn module if contain different functions for performing machine learning with linear models. The term linear model implies that the model is specified as a linear combination of features.

5.2.5 sklearn. neighbors

sklearn. neighbors provides functionality for unsupervised and supervised neighbors-based learning methods. ... The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning).

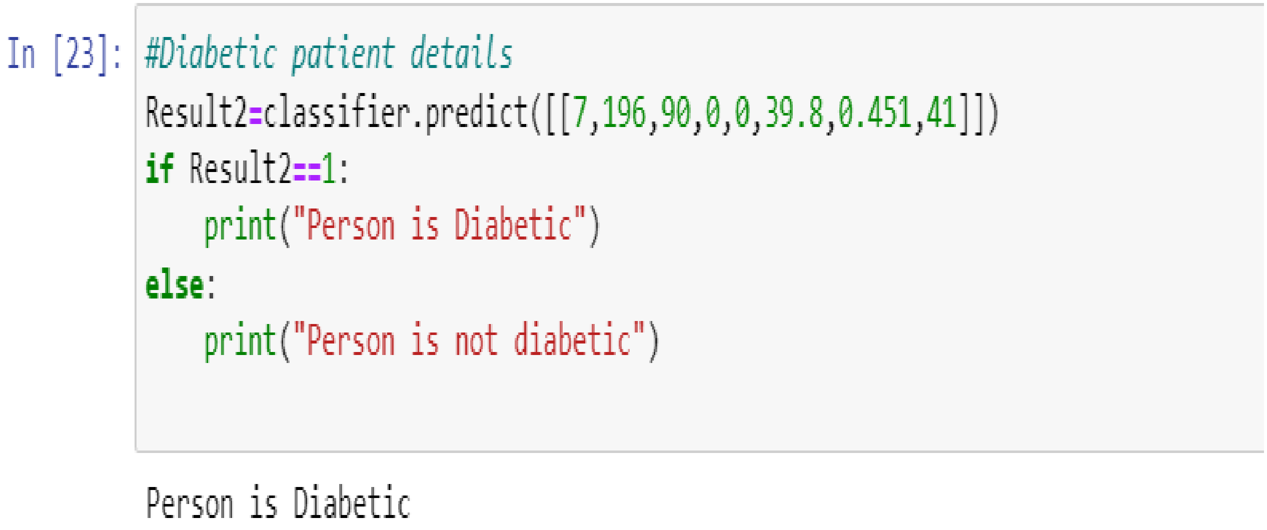
Chapter 6:

RESULT



```
In [21]: #Non diabetic patient details
         Result1=classifier.predict([[1,103,30,38,83,43.3,0.183,33]])
         if Result1==1:
             print("Person is Diabetic")
         else:
             print("Person is not diabetic")

         Person is not diabetic
```



```
In [23]: #Diabetic patient details
         Result2=classifier.predict([[7,196,90,0,0,39.8,0.451,41]])
         if Result2==1:
             print("Person is Diabetic")
         else:
             print("Person is not diabetic")

         Person is Diabetic
```

Chapter 7:

CONCLUSION AND FUTURE

7.1 . CONCLUSION

We have summarized that the most representative applications of artificial intelligence in diabetes education and management. As described above, the diabetes education and management is an essential means of improving the quality of disease management. As a consequence, the integration of education and management approaches with mobile health and AI technologies has become an unstoppable trend. However, we've noticed that there are cons and pros of the wide range of AI-based methods. However, the applicability of one algorithm is problem and data specific. For example, in diabetes classification analysis, based on the classification criteria and characteristic distribution of data, some instances may generate good results with directly applying standard methods supplied by common data analysis tools. However, there are also situations where more advanced models are to be developed to infer a clearer layout of the analysing object.

The machine learning methods can support the doctors to identify and cure diabetic diseases. We shall conclude that the improvement in classification accuracy helps to make the machine learning models get better results. The performance analysis is in terms of accuracy rate among all the classification techniques such as decision tree, logistic regression, k-nearest neighbours, naive bayes, and SVM , random forest , adaboost , xgboost. We have also seen that the accuracy of the existing system is less than 70% hence we proposed to use a combination of classifiers known as Hybrid Approach. Hybrid approach takes advantage by aggregating the merits of two or more techniques. We have found that our system provides us with 75.32 % of accuracy for Decision Tree Classifier, 77.48% accuracy for XGBoost Classifier, 75.75 % accuracy for Voting Classifier and finally 80 percentage of accuracy when using Stacking Classifier and ADA Boost. We have therefore found that the best among all the above classifiers is Stacking Classifier and Adaboost.

6.2. FUTURE SCOPE

In future, if we get a large set of diabetic datasets, we can perform comparative analysis for analysing the performance of each algorithm as well as the Hybrid algorithm so that the best one can be applied for predictive analysis. A particular method to identify diabetes is not very sophisticated way for initial diabetes detection and it is not fully accurate for predicting diseases. That's why we need a smart hybrid predictive analytics diabetes diagnostic system that can effectively work with accuracy and efficiency. We can use data mining , neural network for exploring and utilizing to support medical decision, which improves in diagnosing the risk for pregnant diabetes. Due to the dataset, we have till date are not up to the mark , we cannot predict the type of diabetes, so in future we aim to predicting type of diabetes and explore it, which may improve the accuracy of predicting diabetes. We can also study the causes of diabetes and how to avoid having diabetes.

Chapter 7:

REFERENCES

- https://www.geeksforgeeks.org/machine-learning/
- https://www.kaggle.com/learn/intro-to-machine-learning
- https://www.geeksforgeeks.org/overview-of-data-science/