Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

DIABETES PREDICTION USING MACHINE LEARNING

By

SOMARDH JAISWAL – 20BCE10880

ABHINAV TANWAR – 20BCE10850

AADITYA SREENIVASAN – 20BCE10738

AKSHAT MEHROTRA – 20BCE10246

A Project Report Submitted to

SmartBridge – Externship Program

1. Introduction

In recent years, the prevalence of diabetes has been on the rise, posing a significant global health challenge. Early detection and intervention are crucial for managing the condition effectively and preventing complications. Machine learning, a powerful branch of artificial intelligence, has emerged as a promising tool for predicting and diagnosing diabetes. In this introduction, we will explore the concept of a diabetes prediction application that leverages machine learning algorithms to forecast the risk of developing diabetes

1.1 Overview

Diabetes is a chronic metabolic disorder characterized by high blood sugar levels. It occurs when the body either does not produce enough insulin (a hormone that regulates blood sugar) or is unable to effectively use the insulin it produces. This results in abnormal carbohydrate, protein, and fat metabolism.

1.2 Purpose

The aim of this project is to predict the likelihood of developing diabetes using the logistic regression algorithm and implement it through Flask, a web framework. The project focuses on utilizing logistic regression, a machine learning algorithm, to estimate the probability of an individual developing diabetes based on relevant factors.

Flask, a framework for building web applications, is employed to deploy the predictive model. By applying logistic regression and integrating it with Flask, the project enables users to input their information and obtain a prediction of their probability of developing diabetes. This approach provides a user-friendly and accessible platform for diabetes prediction.

2. Literature Survey

Here are some of the research papers on “Diabetes prediction papers” done by others.

- Shouman, M., Turner, T., & Stocker, R. (2015). Applying machine learning techniques to predict type 2 diabetes mellitus. In 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) (pp. 131-136). IEEE.

- Jagannath, S., Archana, M., & Murthy, G. V. S. (2019). Predicting diabetes using machine learning techniques. In 2019 International Conference on Advanced Computation and Telecommunication (ICACAT) (pp. 65-70). IEEE.

- Haldar, A., Bhowmick, S., & Ghosh, S. (2019). Diabetes prediction using machine learning: A literature review. In 2019 International Conference on Data Science and Communication (ICDSC) (pp. 1-5). IEEE.

- Menaka, R., & Arun, V. (2020). Predicting diabetes using machine learning techniques: A systematic review. Computer Methods and Programs in Biomedicine, 191, 105412.

2.1 Existing Problem

Ensuring Patient Adherence: One of the primary obstacles involves guaranteeing that patients follow the prescribed treatment regimen. When it comes to diabetes management, making necessary lifestyle changes such as maintaining a healthy diet, engaging in regular exercise, and adhering to medication instructions can be challenging for patients. These difficulties in compliance can lead to problems in effectively controlling blood sugar levels and effectively managing the disease.

Self-Monitoring: Diabetic individuals are often required to regularly monitor their blood sugar levels themselves. This process typically involves conducting fingerstick tests or utilizing continuous glucose monitoring devices. However, some patients may perceive this task as burdensome or may not fully grasp the significance of consistent blood sugar monitoring. As a result, they may exhibit inconsistent or inadequate monitoring practices.

Education and Empowerment: Diabetes is a complex condition that necessitates patients having a comprehensive understanding of the disease and its management. Healthcare providers face the challenge of educating and empowering patients to actively participate in their own care. This includes teaching them about proper nutrition, managing medications, practicing self-care, and recognizing signs and symptoms of potential complications.

Comorbidities and Personalized Care: Many individuals with diabetes also experience additional health conditions such as hypertension, cardiovascular disease, or kidney problems. Addressing these coexisting conditions alongside diabetes requires a holistic approach and the development of personalized care plans that cater to the specific needs of each patient. Coordinating care and ensuring effective management of multiple conditions can pose challenges for healthcare providers.

Time Constraints: Doctors often have limited time available for each patient visit. Comprehensive diabetes management necessitates a thorough evaluation, reviewing blood sugar records, adjusting medications as needed, and addressing any concerns or inquiries. Time limitations may restrict the doctor's ability to provide comprehensive care and personalized counseling, which could potentially impact patient outcomes.

2.2 Proposed Solution

A predictive application designed for diabetes can examine diverse health information, including blood sugar levels, physical activity, diet, and relevant parameters. By continuously monitoring and analyzing this data, the application can detect patterns and issue advance notifications or forecasts regarding the likelihood of developing diabetes or encountering fluctuations in blood glucose levels. Prompt detection empowers individuals to take preventive measures and adopt lifestyle changes to manage or postpone the onset of diabetes.

Tailored Suggestions: These applications have the ability to provide customized recommendations based on an individual's health data. They offer insights into dietary choices, exercise routines, and medication management to help maintain optimal blood sugar levels. By tailoring the suggestions to each user's specific requirements, the application supports diabetes management and reduces the risk of complications.

Enhanced Self-Care: Diabetic prediction applications often incorporate features for self-monitoring and self-care. They enable tracking of daily blood sugar readings, medication usage, physical activity, and other health-related information. By offering reminders, alerts, and visual representations of data, these applications empower individuals to take charge of their condition and make well-informed decisions about their health.

Data Analysis and Reporting: With a substantial user base, these applications can collect anonymized data from multiple users, allowing for the identification of broader trends and patterns related to diabetes. Through analysis of this data, researchers and healthcare providers can gain valuable insights into the effectiveness of different interventions, identify risk factors, and enhance diabetes management strategies on a population level.

Remote Monitoring and Support: Remote monitoring and support are particularly important for individuals with diabetes, especially those with type 1 diabetes or high-risk cases. Diabetic prediction applications can integrate with wearable devices like continuous glucose monitors (CGMs) or insulin pumps to provide real-time data and alerts to both users and their healthcare providers. This facilitates remote monitoring and timely interventions, enabling adjustments to treatment plans and reducing the necessity for frequent in-person visits.

3. Theoritical Analysis

Theoretical analysis of diabetes prediction using machine learning involves understanding the problem, identifying suitable algorithms, evaluating performance metrics, and considering ethical considerations.

Data Collection: Collect a representative dataset that includes samples with relevant features and corresponding labels indicating diabetes or non-diabetes. Ensure the dataset is properly labeled and contains a sufficient number of samples to train and evaluate your machine learning model.

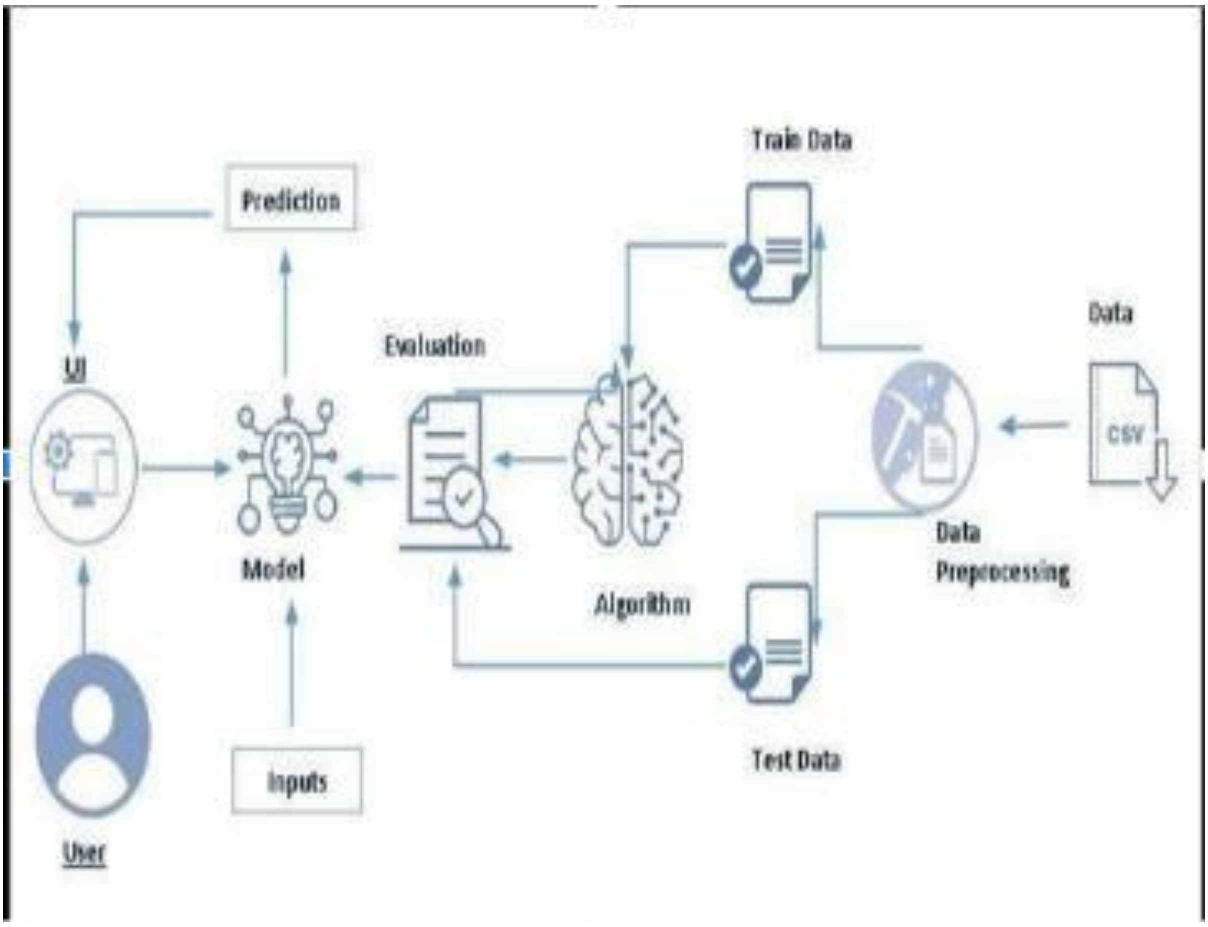
Data Preprocessing: Perform necessary preprocessing steps on the dataset, such as handling missing values, dealing with outliers, normalizing or scaling features, and encoding categorical variables if any. Split the dataset into training and test sets to evaluate the model's performance.

Algorithm Selection: Consider various machine learning algorithms suitable for binary classification. Some commonly used algorithms for diabetes prediction include logistic regression, decision trees, random forests, support vector machines (SVM), and artificial neural networks (ANN). Evaluate the strengths, weaknesses, and assumptions of each algorithm based on your dataset and problem requirements.

Model Training and Evaluation: Train the selected machine learning models on the training set. Use appropriate evaluation metrics such as accuracy, precision, recall, F1 score, or area under the receiver operating characteristic curve (AUC-ROC) to assess the performance of each model on the test set. Compare the results and choose the algorithm with the best performance.

3.1 Block Diagram

The block diagram that provides a visual representation of how our Project appears and how technically it is being managed.



The initiative has collected the necessary data. The data is then pre-processed using various techniques to make the data clean and optimal for model training and assessment. The data is then separated into training data and test data. The trained data is used to train the model, and an algorithm is created to facilitate the process. The model is then assessed relative to the test data. The information is then utilised for the prediction. The prediction is made based on the user's input, and the answer is then displayed on the user interface.

3.2 Hardware / Software designing

The diabetes prediction project using logistic regression and Flask deployment necessitates multiple software components to ensure the system's proper operation. Here is a thorough description of every software requirement:

- **Python:** Python is a flexible programming language that is widely used for machine learning and web development. It is the primary language used to implement the project's algorithms, data preprocessing, model training, and prediction.

- **Machine Learning Library:** The project utilises numerous machine learning libraries, including scikit-learn, Numpy, Mathplotlib, and pandas. These libraries provide dependable implementations of linear regression algorithms, feature selection methods, model evaluation metrics, and data preprocessing techniques.

- **Flask:** Flask is a lightweight Python web framework that facilitates the creation of web applications. It is used to develop a user interface that allows users to enter their body measurements, receive predictions, and view the results. Additionally, Flask facilitates the deployment of the linear regression model as a web API.

- **HTML/CSS/JavaScript:** Front-end web technologies such as HTML, CSS, and JavaScript are required for designing and developing the web application's user interface. HTML is utilised for content organisation, CSS for formatting and layout, and JavaScript for interactive elements and input validation.

- **Development Tools and IDEs:** Integrated Development Environments (IDEs) like PyCharm, Visual Studio Code, and Jupyter Notebook can facilitate coding, debugging, and collaboration. Version control systems such as Git and project management tools such as GitHub and GitLab can aid in code management andteam collaboration.

- **Deployment Platform:** The project may be deployed on cloud platforms such as Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. These platforms provide the infrastructure and services necessary to host the web application and scale traffic.

4. Experimental Investigations

Preprocessing & Exploratory Data Analysis:

Once we have successfully loaded the dataset then we look for the following things:

- **Is there any NULL values in the dataset? :** There were no NULL values as we checked for all the given features in the dataset.
- **Respective data types of the features in the dataset? :** All the features are of float type except for the "Age" attribute.
- **Statistical Description of the dataset? :** For this we used the function describe() to understand the count, mean, standard deviation, minimum, maximum and quartiles for each attribute.
- **Are there any duplicate values in the dataset? :** No, there are no duplicate values as such in the dataset.
- **Are there outliers in the dataset? :** In order to understand this we looked upon the boxplot of each feature in the dataset so that we can understand a five-number and get a visual representation about the presence of the dataset. After running the boxplot visualization we inferred that there are outliers in the dataset and additionally all the attributes are not at the same scale.
- **How to get rid of outliers? :** In order to get rid of outliers we used the concept of IQR i.e., Interquartile range which means we will remove those values for an attribute who are lesser than 1.5 * min or 1.5 * max. Hence, we get rid of the outliers.

Model Building Algorithm

- Logistic Regression:

It is a popular algorithm used in machine learning for binary classification problems. It models the relationship between a set of input features and a binary target variable, predicting the probability of the target variable belonging to a particular class.

This algorithm consists of:

Data Preparation: Begin by collecting and preprocessing your dataset. Ensure that your target variable is binary and your input features are numeric or appropriately encoded.

Model Representation: Logistic regression assumes a linear relationship between the input features and the log-odds of the target variable. The model representation can be defined as follows:

- Let X be the input feature matrix of shape (m, n), where m is the number of samples and n is the number of features.
- Let θ be the parameter vector of shape (n, 1), which contains the coefficients for each feature.
- Let y be the target variable vector of shape (m, 1), which contains the binary labels (0 or 1) for each sample.
- The linear relationship between the features and log-odds is given by: $z = X\theta$.

Hypothesis Function: The hypothesis function transforms the linear relationship into the range [0, 1] using a sigmoid function, representing the estimated probability of the target variable being 1. The hypothesis function is defined as follows:

- $h\theta(z) = 1 / (1 + e^{\wedge}(-z))$

Cost Function: The cost function measures the error between the predicted probabilities and the actual binary labels. In logistic regression, the cost function is derived from the likelihood of the observed data. The most commonly used cost function is the binary cross-entropy (or log loss) function:

- $J(\theta) = (-1/m) * \sum[y * \log(h\theta(z)) + (1 - y) * \log(1 - h\theta(z))]$

Parameter Optimization: The goal is to find the optimal parameter vector θ that minimizes the cost function. This is typically achieved using optimization algorithms such as gradient descent or advanced techniques like L-BFGS.

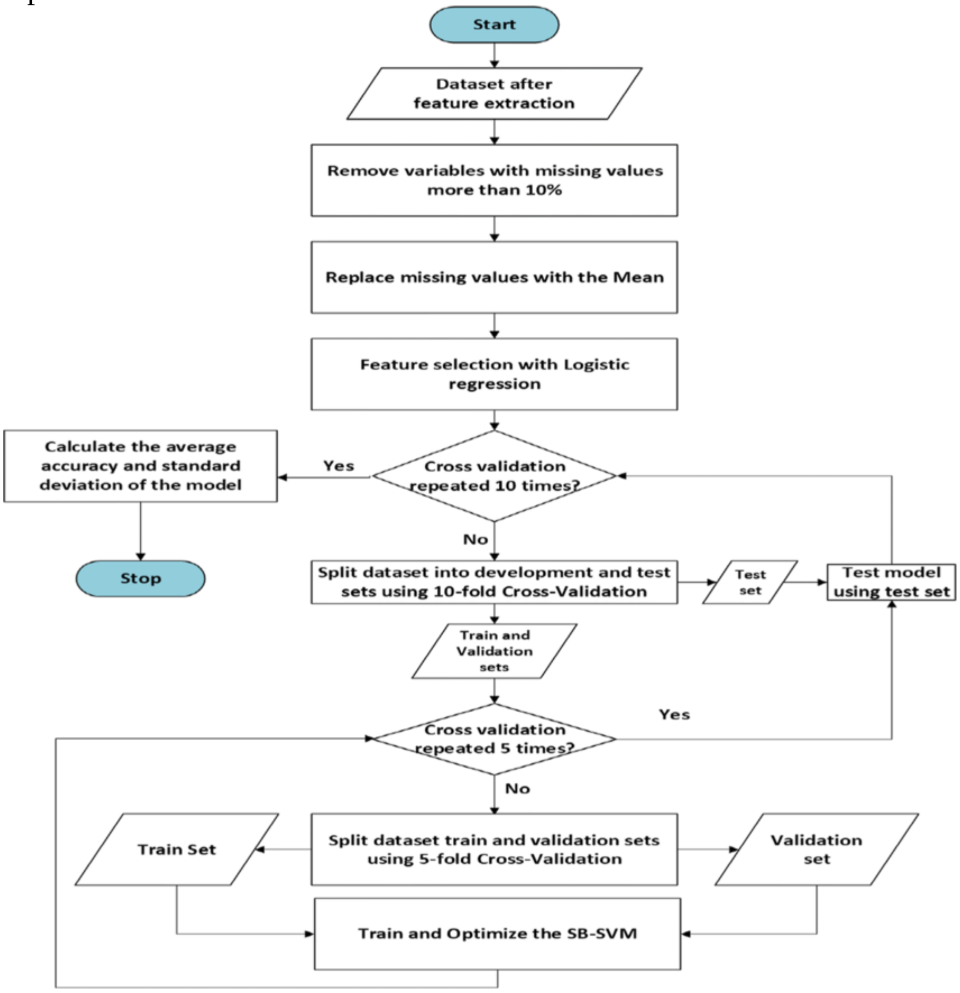
Training: Using the optimization algorithm, iteratively update the parameter vector θ to minimize the cost function. Repeat this process until convergence, or until a predefined number of iterations is reached.

Prediction: Once the model is trained and the optimal parameter vector θ is obtained, you can make predictions on new, unseen data. Given the input features x, calculate the predicted probability using the hypothesis function: y_pred = hθ(Xθ). Finally, apply a threshold (e.g., 0.5) to convert the probabilities into binary predictions.

Evaluation: Assess the performance of your logistic regression model using appropriate evaluation metrics such as accuracy, precision, recall, F1 score, or area under the receiver operating characteristic curve (AUC-ROC), depending on your problem requirements**.**

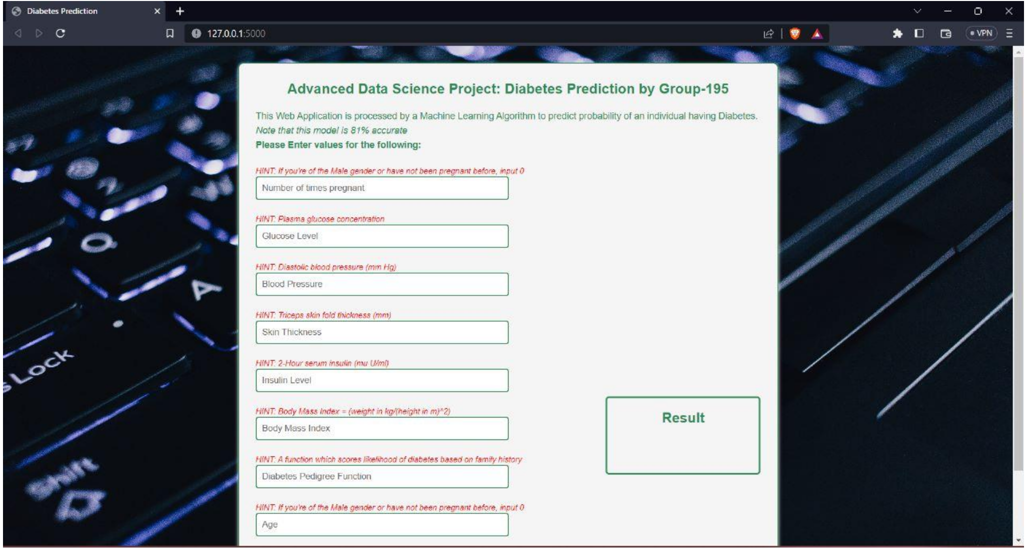
5. Flowchart

The flowchart represents the control flow of the project for accurate body fat prediction using linear regression and Flask deployment. It outlines the key steps involved in the process, starting from data collection and preprocessing, model training and evaluation, deployment using Flask, user interface development, input processing, prediction generation, and displaying the predicted body fat count to the user. The flowchart provides a visual representation of the sequential order of these steps, guiding the implementation of the project and illustrating the overall workflow from data input to the final output of body fat prediction.



Start
Dataset after feature extraction
Remove variables with missing values more than 10%
Replace missing values with the Mean
Feature selection with Logistic regression
Cross validation repeated 10 times?
Yes
Calculate the average accuracy and standard deviation of the model
Stop
No
Split dataset into development and test sets using 10-fold Cross-Validation
Test set
Test model using test set
Train and Validation sets
Cross validation repeated 5 times?
Yes
No
Train Set
Split dataset train and validation sets using 5-fold Cross-Validation
Validation set
Train and Optimize the SB-SVM

6. Result



Advanced Data Science Project: Diabetes Prediction by Group-195

This Web Application is processed by a Machine Learning Algorithm to predict probability of an individual having Diabetes.

Note that this model is 81% accurate

Please Enter values for the following:

HINT: If you're of the Male gender or have not been pregnant before, input 0
Number of times pregnant

HINT: Plasma glucose concentration
Glucose Level

HINT: Diastolic blood pressure (mm Hg)
Blood Pressure

HINT: Triceps skin fold thickness (mm)
Skin Thickness

HINT: 2-Hour serum insulin (mu U/ml)
Insulin Level

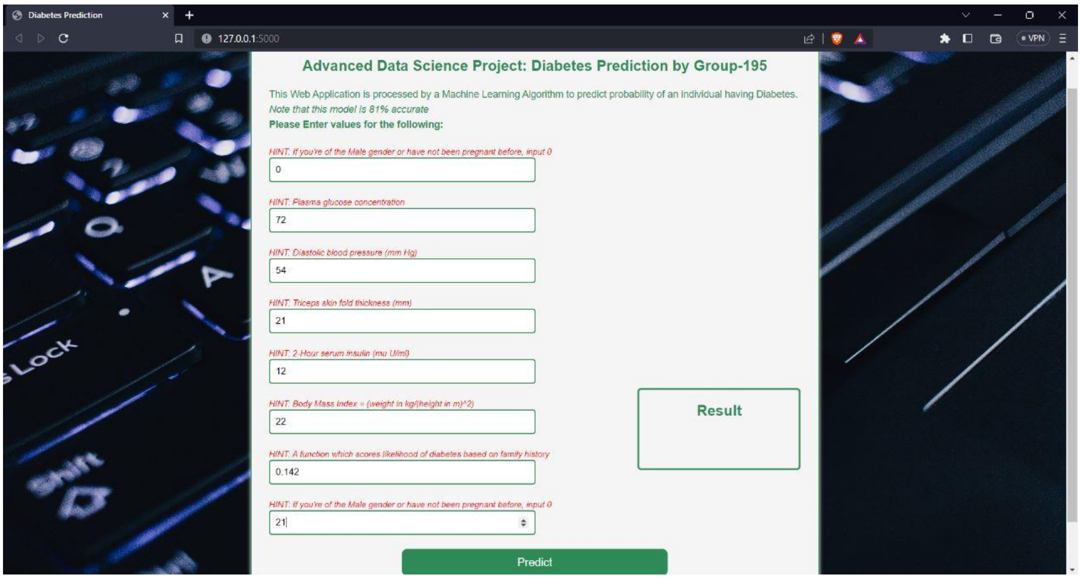
HINT: Body Mass Index = (weight in kg/(height in m)^2)
Body Mass Index

HINT: A function which scores likelihood of diabetes based on family history
Diabetes Pedigree Function

HINT: If you're of the Male gender or have not been pregnant before, input 0
Age

Result

User Input of Values:



Advanced Data Science Project: Diabetes Prediction by Group-195

This Web Application is processed by a Machine Learning Algorithm to predict probability of an individual having Diabetes.

Note that this model is 81% accurate

Please Enter values for the following:

HINT: If you're of the Male gender or have not been pregnant before, input 0
0

HINT: Plasma glucose concentration
72

HINT: Diastolic blood pressure (mm Hg)
54

HINT: Triceps skin fold thickness (mm)
21

HINT: 2-Hour serum insulin (mu U/ml)
12

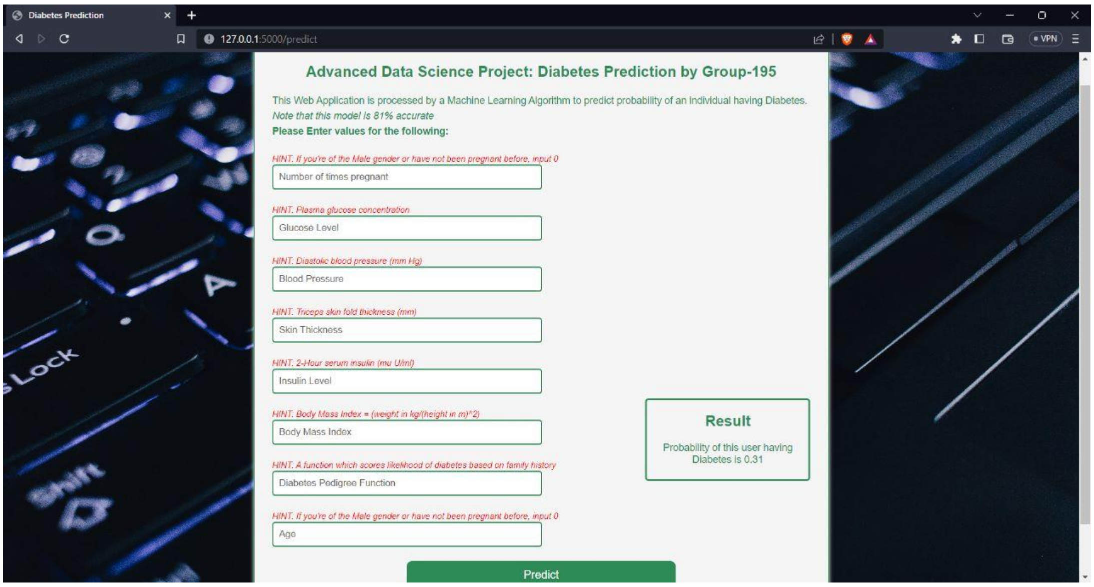
HINT: Body Mass Index = (weight in kg/(height in m)^2)
22

HINT: A function which scores likelihood of diabetes based on family history
0.142

HINT: If you're of the Male gender or have not been pregnant before, input 0
21

Result

Prediction:



Advanced Data Science Project: Diabetes Prediction by Group-195

This Web Application is processed by a Machine Learning Algorithm to predict probability of an individual having Diabetes.

Note that this model is 81% accurate

Please Enter values for the following:

HINT: If you're of the Male gender or have not been pregnant before, input 0

Number of times pregnant

HINT: Plasma glucose concentration

Glucose Level

HINT: Diastolic blood pressure (mm Hg)

Blood Pressure

HINT: Triceps skin fold thickness (mm)

Skin Thickness

HINT: 2-Hour serum insulin (mu U/ml)

Insulin Level

HINT: Body Mass Index = (weight in kg/(height in m)^2)

Body Mass Index

HINT: A function which scores likelihood of diabetes based on family history

Diabetes Pedigree Function

HINT: If you're of the Male gender or have not been pregnant before, input 0

Age

Predict

Result

Probability of this user having Diabetes is 0.31

7. Advantages & Disadvantages

Advantages of the method:

Early detection: A diabetes prediction system can identify individuals who are at risk of developing diabetes before symptoms appear. This early detection allows for timely intervention and preventive measures, reducing the risk of complications associated with diabetes.

Personalized risk assessment: Diabetes prediction systems can assess an individual's risk of developing diabetes based on various factors such as age, gender, family history, lifestyle, and medical history. This personalized risk assessment helps healthcare professionals tailor interventions and advice to each individual, improving the effectiveness of preventive strategies.

Improved patient outcomes: By identifying individuals at high risk of developing diabetes, a prediction system enables proactive management and intervention. This may include lifestyle modifications, such as dietary changes and increased physical activity, as well as medical interventions. Early intervention can lead to better glycemic control and a decreased risk of complications associated with diabetes, ultimately improving patient outcomes.

Cost-effectiveness: Predictive systems can potentially help reduce healthcare costs by focusing resources on high-risk individuals. By identifying those who are likely to develop diabetes, healthcare providers can allocate preventive measures and interventions to this specific group, rather than providing generalized care to the entire population. This targeted approach can optimize resource utilization and potentially reduce the economic burden associated with diabetes.

Empowering individuals: Diabetes prediction systems can empower individuals by providing them with personalized information about their risk factors. This knowledge enables individuals to take proactive steps towards reducing their risk of developing diabetes. They can make informed decisions about their lifestyle choices, monitor their health, and engage in preventive measures to improve their overall well-being.

Research and public health planning: Aggregate data collected by diabetes prediction systems can contribute to population-level research and public health planning. By analyzing trends and patterns, researchers and policymakers can identify high-risk populations, develop targeted interventions, and allocate resources strategically to prevent diabetes on a broader scale.

Disadvantages of the Method:

Limited accuracy: Although these applications employ various algorithms and models to predict diabetes risk, their accuracy may not always be reliable. The predictions are based on general patterns and statistical data, which may not take into account individual variations or specific medical conditions.

False positives and negatives: Diabetes prediction applications may sometimes produce false positive or false negative results. False positives can lead to unnecessary anxiety and medical interventions, while false negatives may provide a false sense of security, delaying necessary medical attention.

Lack of personalized guidance: While diabetes prediction applications can indicate the risk of developing diabetes, they often do not provide personalized guidance on preventive measures or lifestyle modifications. Users may receive a prediction without knowing how to take appropriate actions to mitigate their risk effectively.

Inadequate data security: Privacy and data security are significant concerns when using health-related applications. Diabetes prediction applications require access to personal health information, which must be handled securely. If the application lacks proper security measures, there is a risk of data breaches or unauthorized access to sensitive information.

Ethical considerations: Some diabetes prediction applications may not adhere to appropriate ethical guidelines. They might engage in data sharing or selling user information without informed consent, potentially compromising user privacy and trust.

Psychological impact: Receiving a diabetes risk prediction can cause unnecessary stress and anxiety, particularly for individuals who may be prone to health-related anxiety. False positive results or a heightened sense of constant risk can negatively impact mental well-being.

Lack of professional oversight: Diabetes prediction applications typically lack direct supervision or input from healthcare professionals. This absence may lead to inaccurate predictions or a lack of comprehensive medical advice, potentially missing other risk factors or health conditions that could contribute to diabetes.

8. Applications

The solution of diabetes prediction using logistic regression and Flask deployment can be implemented in a variety of contexts where it is advantageous to estimate diabetes probability in human body. Here are some instances:

- **Personal Health Monitoring:** Individuals can use the diabetes prediction application to monitor their health status and assess their risk of developing diabetes. By regularly inputting relevant health information, such as blood glucose levels, BMI, family history, and lifestyle habits, the application can provide personalized predictions and alert users to potential diabetes risk. This empowers individuals to make proactive changes in their lifestyle and seek medical advice if necessary.
- **Preventive Healthcare:** Healthcare providers can integrate the diabetes prediction application into their practices to offer preventive healthcare services. By encouraging patients to use the application and providing them with personalized risk assessments, healthcare professionals can identify high-risk individuals and implement targeted interventions. This can include lifestyle counseling, dietary recommendations, and exercise programs aimed at reducing the likelihood of developing diabetes.
- **Public Health Initiatives**: Diabetes prediction applications can contribute to public health initiatives by collecting anonymized data from a large user base. Aggregating this data can provide valuable insights into diabetes trends, risk factors, and prevalence rates across different populations. Public health agencies can utilize this information to develop targeted interventions, allocate resources effectively, and raise awareness about diabetes prevention strategies.
- **Research and Clinical Studies:** Diabetes prediction applications can serve as valuable tools for research and clinical studies. By collecting data from users who voluntarily participate in studies, researchers can analyze the data to identify novel risk factors, evaluate the effectiveness of interventions, and refine prediction models. This can lead to advancements in diabetes research, improved risk assessment algorithms, and the development of more accurate prediction models.
- **Health Insurance and Wellness Programs:** Health insurance companies and wellness program providers can integrate diabetes prediction applications into their offerings. By incentivizing users to utilize the application and providing personalized risk assessments, these entities can promote healthier behaviors, offer targeted wellness programs, and potentially reduce the financial burden associated with diabetes management and treatment.

9. Conclusion

Diabetes prediction application developed using machine learning techniques can provide valuable insights and predictions regarding an individual's risk of developing diabetes. By utilizing algorithms such as logistic regression, the application can analyze various factors such as age, weight, BMI, blood pressure, glucose levels, and family history to generate accurate predictions.

The application's machine learning models can learn from a large dataset of known diabetes cases and non-diabetic individuals to identify patterns and relationships between input features and the likelihood of diabetes occurrence. This allows the application to make predictions for new individuals based on their specific characteristics.

By providing individuals with personalized risk assessments, the diabetes prediction application can empower them to make proactive lifestyle choices, seek appropriate medical advice, and take preventive measures to manage their health effectively. Early detection and intervention can lead to better disease management and potentially prevent or delay the onset of diabetes-related complications.

In conclusion, a machine learning-based diabetes prediction application holds great promise in improving healthcare outcomes by enabling early identification of individuals at risk and facilitating timely interventions for diabetes prevention and management.

10.Future Scope

Improved Accuracy: With the availability of more data and continuous advancements in machine learning algorithms, there is potential for diabetes prediction applications to achieve heightened accuracy in identifying individuals who are at risk of developing diabetes. This can prove beneficial for early intervention and the implementation of preventive measures.

Integration with Wearable Devices: Given the rising popularity and advancements in wearable devices, it is possible for diabetes prediction applications to integrate with these devices. This integration would allow for the real-time collection of data, such as blood glucose levels, physical activity, sleep patterns, and heart rate. Continuous monitoring through wearable devices can enhance the precision of predictions and provide timely feedback to users.

Behavioral Interventions: Diabetes prediction applications can expand their capabilities beyond merely predicting the risk of developing diabetes. They can also offer personalized recommendations and behavioral interventions to assist individuals in adopting healthier habits, including modifications to their diet, engaging in physical activity, managing stress, and adhering to medication schedules.

Telemedicine Integration: Integrating diabetes prediction applications with telemedicine platforms enables remote consultations with healthcare professionals. Users can share their prediction results, track their progress, and receive tailored advice from healthcare providers. This integration enhances the accessibility and convenience of healthcare services.

Population Health Management: Aggregated and anonymized data from diabetes prediction applications can be utilized for population health management. Public health authorities can leverage this data to identify high-risk populations, design preventive programs, and allocate resources more efficiently.

Research and Development: Diabetes prediction applications serve as valuable tools for researchers to gather extensive data on diabetes risk factors, disease progression, and treatment outcomes. This data contributes to medical research, leading to a deeper understanding of the disease and the development of more effective interventions.

11.Bibilography

- https://techvidvan.com/tutorials/diabetes-prediction-using-machine-learning/
- https://www.researchgate.net/publication/347091823_Diabetes_Prediction_Using_Machine_Learning
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9553420/
- https://medium.com/geekculture/diabetes-prediction-using-machine-learning-python-23fc98125d8
- https://www.sciencedirect.com/science/article/pii/S2214860422000963
- https://www.frontiersin.org/articles/10.3389/fpsyg.2021.631179/full
- https://www.mdpi.com/2076-3417/11/21/9797

Appendix

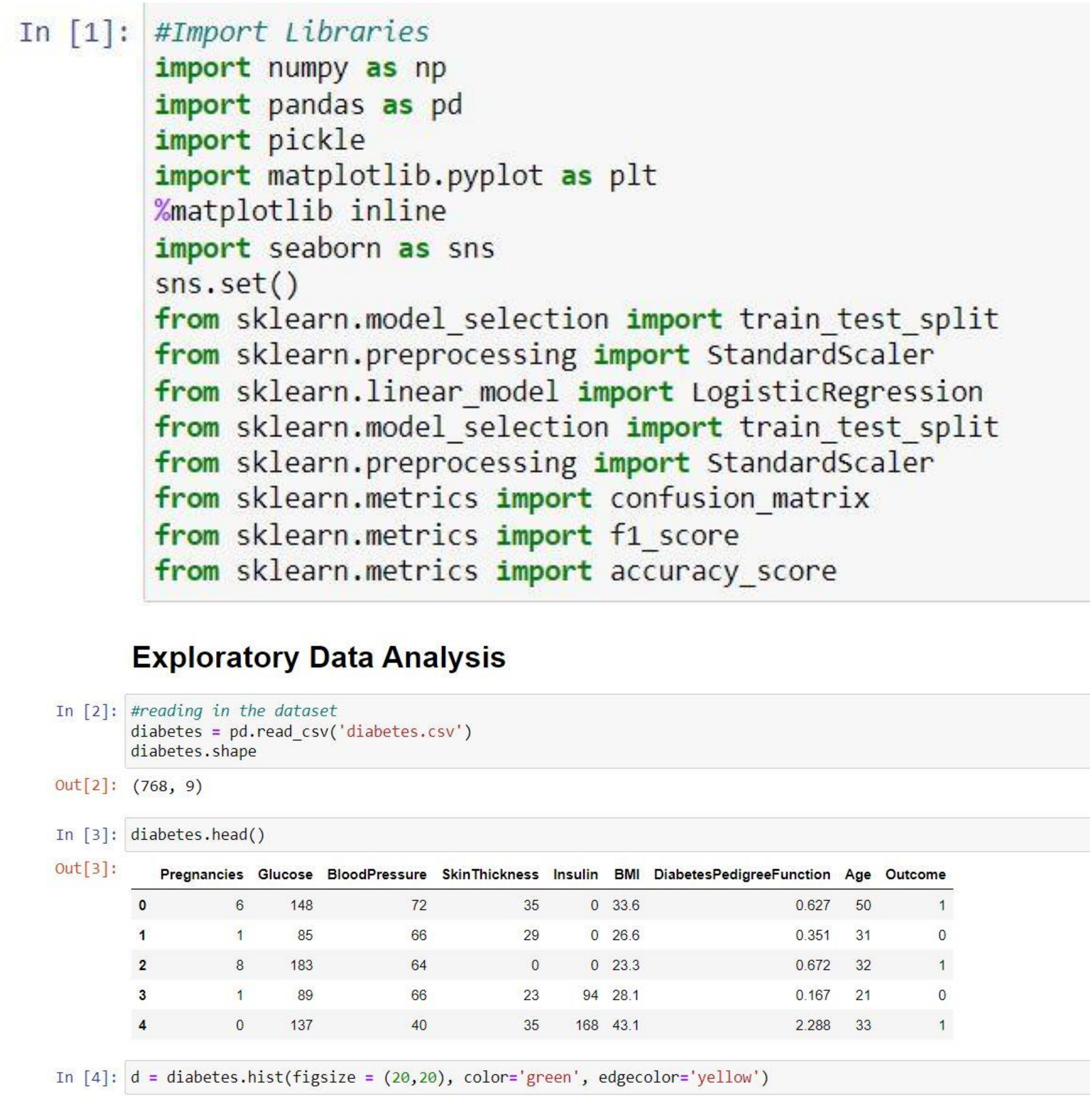
Code:-

The different segments of the code are displayed below:

Dataset:-

https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-databaseMODELS

COMPARISON CODE:



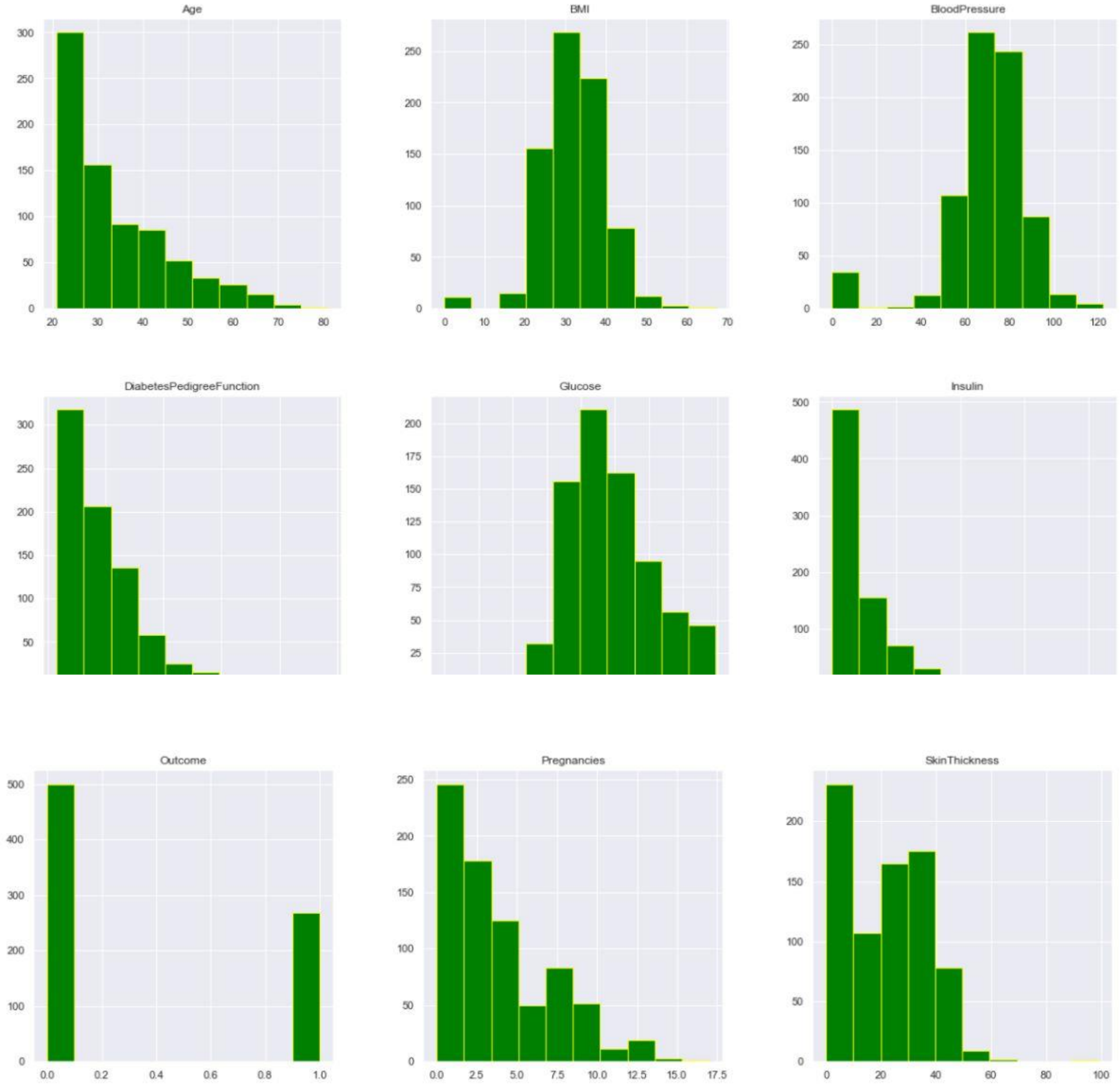
```
In [1]: #Import Libraries
        import numpy as np
        import pandas as pd
        import pickle
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns
        sns.set()
        from sklearn.model_selection import train_test_split
        from sklearn.preprocessing import StandardScaler
        from sklearn.linear_model import LogisticRegression
        from sklearn.model_selection import train_test_split
        from sklearn.preprocessing import StandardScaler
        from sklearn.metrics import confusion_matrix
        from sklearn.metrics import f1_score
        from sklearn.metrics import accuracy_score
```

Exploratory Data Analysis

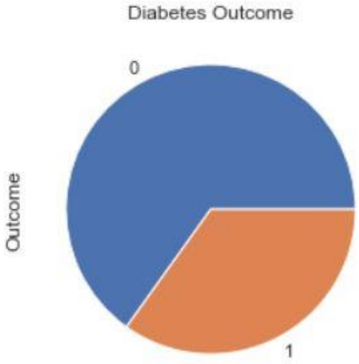
```
In [2]: #reading in the dataset
        diabetes = pd.read_csv('diabetes.csv')
        diabetes.shape

Out[2]: (768, 9)

In [3]: diabetes.head()

Out[3]:
```

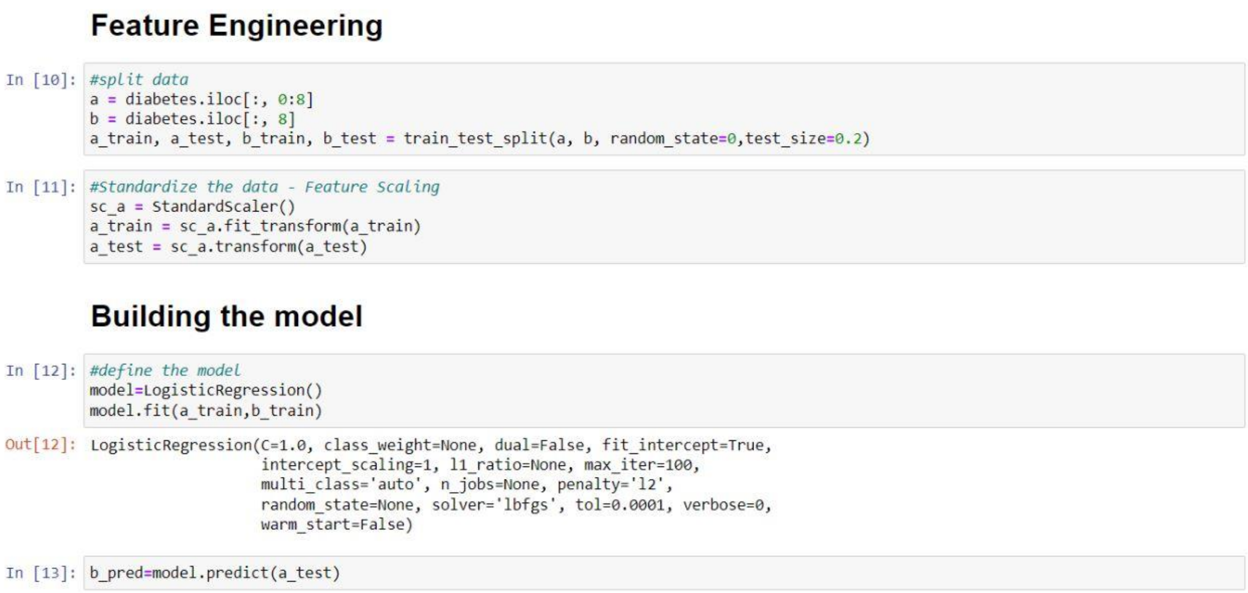
	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

```
In [4]: d = diabetes.hist(figsize = (20,20), color='green', edgecolor='yellow')
In [4]: d = diabetes.hist(figsize = (20,20), color='green', edgecolor='yellow')
```



```
In [6]: print(diabetes.Outcome.value_counts())
        diabetes['Outcome'].value_counts().plot(kind='pie').set_title('Diabetes Outcome')

        0    500
        1    268
        Name: Outcome, dtype: int64

Out[6]: Text(0.5, 1.0, 'Diabetes Outcome')
```





Feature Engineering

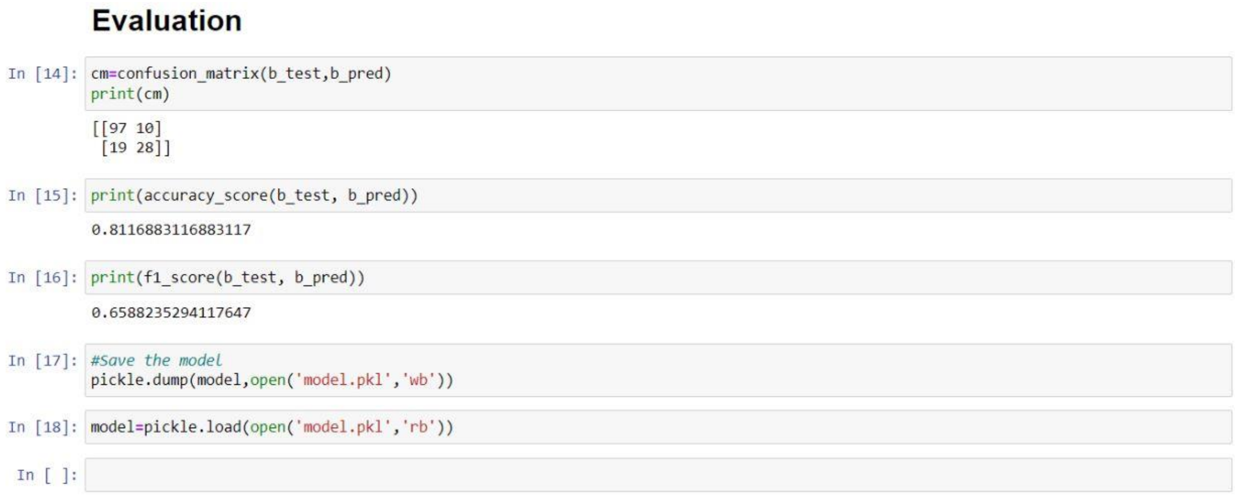
```
In [10]: #split data
         a = diabetes.iloc[:, 0:8]
         b = diabetes.iloc[:, 8]
         a_train, a_test, b_train, b_test = train_test_split(a, b, random_state=0,test_size=0.2)
```

```
In [11]: #Standardize the data - Feature Scaling
         sc_a = StandardScaler()
         a_train = sc_a.fit_transform(a_train)
         a_test = sc_a.transform(a_test)
```

Building the model

```
In [12]: #define the model
         model=LogisticRegression()
         model.fit(a_train,b_train)

Out[12]: LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                            intercept_scaling=1, l1_ratio=None, max_iter=100,
                            multi_class='auto', n_jobs=None, penalty='l2',
                            random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                            warm_start=False)
```

```
In [13]: b_pred=model.predict(a_test)
```



Evaluation

```
In [14]: cm=confusion_matrix(b_test,b_pred)
         print(cm)

         [[97 10]
          [19 28]]
```

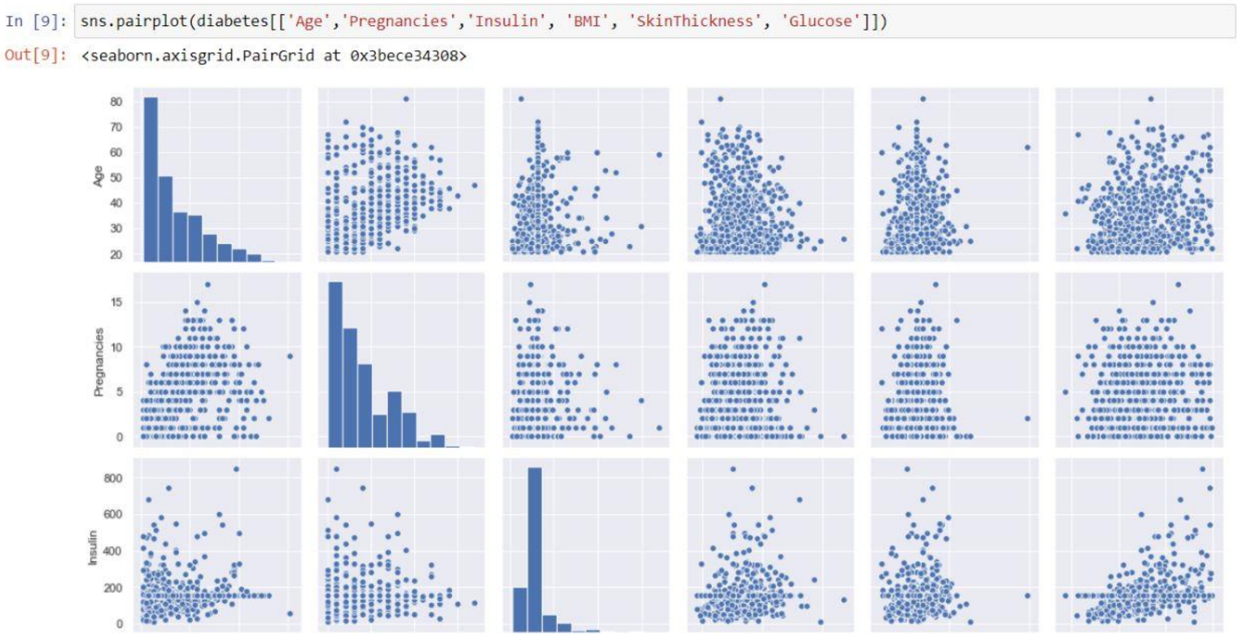
```
In [15]: print(accuracy_score(b_test, b_pred))

         0.8116883116883117
```

```
In [16]: print(f1_score(b_test, b_pred))

         0.6588235294117647
```

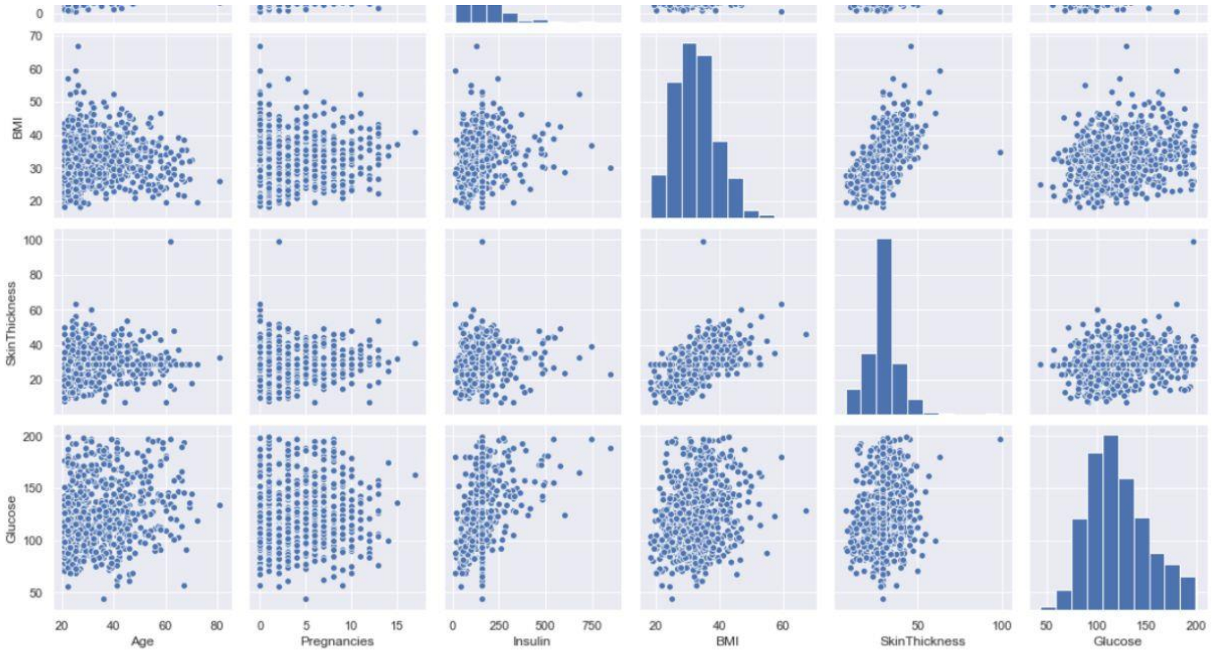
```
In [17]: #Save the model
         pickle.dump(model,open('model.pkl','wb'))
```

```
In [18]: model=pickle.load(open('model.pkl','rb'))
```

```
In [ ]:
```



```
In [9]: sns.pairplot(diabetes[['Age','Pregnancies','Insulin', 'BMI', 'SkinThickness', 'Glucose']])

Out[9]: <seaborn.axisgrid.PairGrid at 0x3bece34308>
```


0
70
60
50
40
30
20
BMI
100
80
60
40
20
SkinThickness
200
150
100
50
Glucose
20
40
60
80
Age
0
5
10
15
Pregnancies
0
250
500
750
Insulin
20
40
60
BMI
50
100
SkinThickness
50
100
150
200
Glucose