

Name: Muhammed SalihK

Email : [msalikh1@gmail.com](mailto:msalikh1@gmail.com)

Phone : 9846852615

## **Title of the project : DIABETES PREDICTION**

### **Description:**

Predict whether a patient had diabetes

### **Introduction:**

Diabetes, is a group of metabolic disorders in which there are high blood sugar levels over a prolonged period. Symptoms of high blood sugar include frequent urination, increased thirst, and increased hunger. If left untreated, diabetes can cause many complications. Acute complications can include diabetic ketoacidosis, hyperosmolar hyperglycemic state, or death. Serious long-term complications include cardiovascular disease, stroke, chronic kidney disease, foot ulcers, and damage to the eyes.

The objective of dataset is to predict whether the patient has diabetes or not on the basis of the features we will provide to our machine learning model, and for that, we will be using the famous Pima Indians Diabetes Database. The dataset consists of several independent variables and one dependent variable

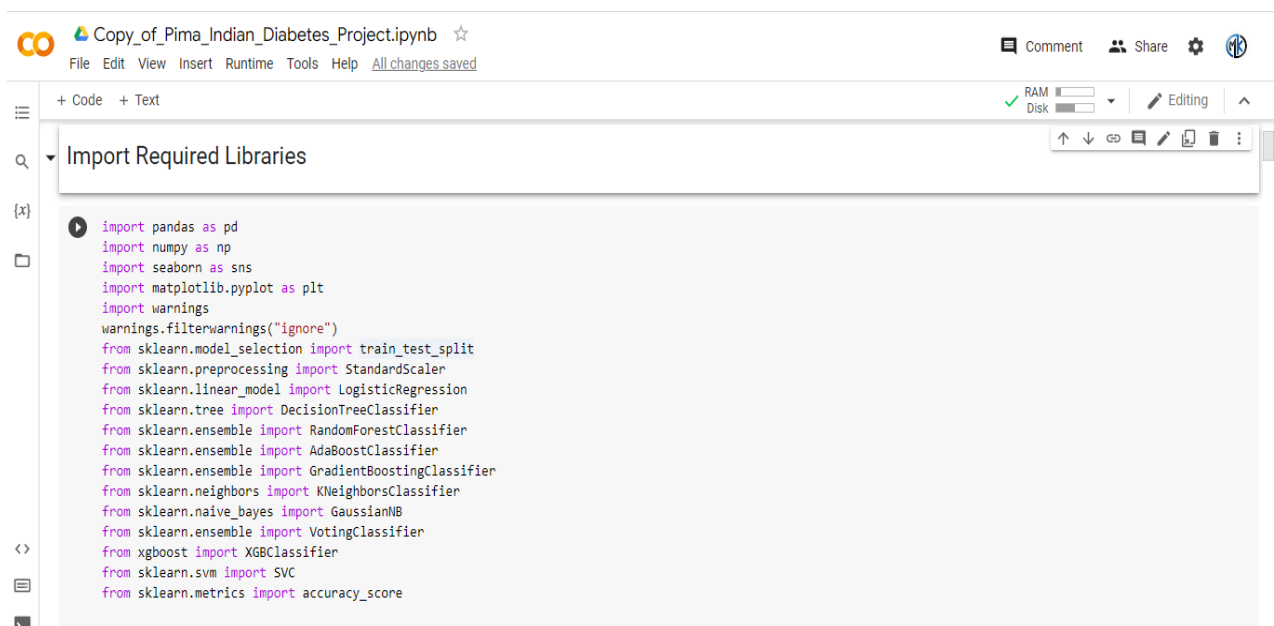
### **Details about the dataset:**

The datasets consists of several medical predictor variables and one target variable, Outcome. Predictor variables includes the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.

- Pregnancies: Number of times pregnant
- Glucose: Plasma glucose concentration a 2 hours in an oral glucose tolerance test

- BloodPressure: Diastolic blood pressure (mm Hg)
- SkinThickness: Triceps skin fold thickness (mm)
- Insulin: 2-Hour serum insulin (mu U/ml)
- BMI: Body mass index (weight in kg/(height in m)^2)
- DiabetesPedigreeFunction: Diabetes pedigree function
- Age: Age (years)
- Outcome: Class variable (0 or 1)

## Importing Libraries:



The screenshot shows a Jupyter Notebook interface with a code cell titled "Import Required Libraries". The code cell contains the following Python code:

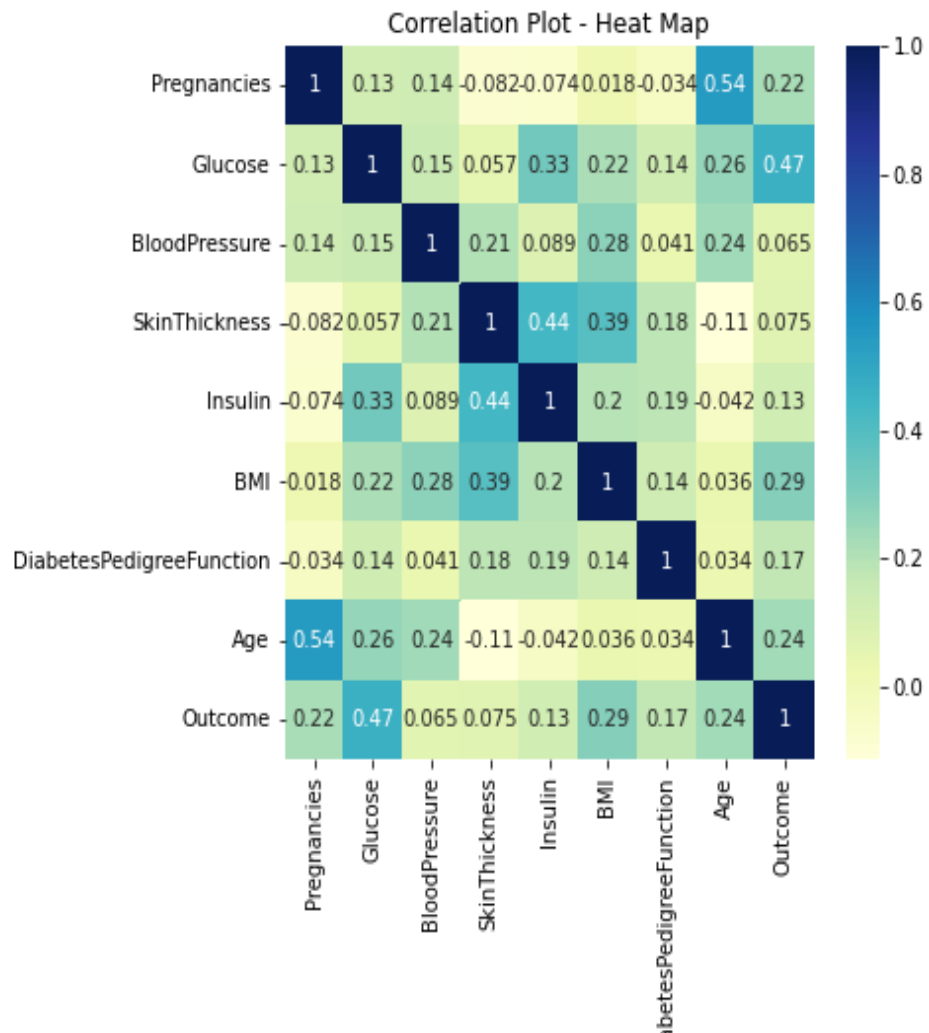
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import VotingClassifier
from xgboost import XGBClassifier
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
```

Load the dataset

## Exploratory Data Analysis(EDA):

- Analyze the target variable (outcome).
- Check the unique values,count and correlation in target variable and also plot it.
- Check the dataset information such as shape,info,describe,identify missing values.
- Plot the correlation between the features of input dataset.

## Correlation between the features :



# Pairplot:

`seaborn.axisgrid.PairGrid at 0x7f8a14d9ffdb`



- There are no missing values but that is actually not a true story as in this particular dataset all the missing values were given the 0 as a value which is not good for the authenticity of the dataset. Hence we will replace the zero's using the median values.
- Outlier treatment : Check the continuous columns in dataset and cap it with standard deviation.

## **Train Test Split:**

- Done train\_test\_split as X and y

## **Scale Down:**

- Done using StandardScaler function.
- X\_train scaled using fit\_transform and X\_test using transform.
- Create pickle file of scale down using joblib.

## **Applying Algorithms:**

- Apply all classification algorithms such as LogisticRegression, DecisionTreeClassifier, RandomForestClassifier, AdaBoostClassifier, GradientBoostingClassifier, SVC, KNeighborsClassifier, GaussianNB.
- Check the score of train and test accuracy of X and y.

## **Hyperparameter Tuning:**

- Done hyperparameter tuning in all algorithms for finding the best model.
- Print train accuracy, test accuracy and best parameter of each algorithms.
- I choose RandomForestClassifier as best model on the basis of accuracy values.
- Create pickle file of model using joblib.



```
File Edit Selection View Go Run Terminal Help index.html - Visual Studio Code

Copy of Pima Indian Diabetes Project.ipynb M app.py M index.html x
C:\Users\MUHAMMED SALIH K\DATA SCIENCE NOTES\DATA SCIENCE\Project\Project_1\templates\index.html > ...

59     <input placeholder="SkinThickness" name="4" id="first_name" type="text" class="validate">
60 </div>
61 <div class="input-field col s4">
62     <label for="last_name"><b>Insulin (0-1000) </b></label>
63     <br>
64     <input id="last_name" name="5" placeholder="Insulin level" type="text" class="validate">
65 </div>
66 <div class="input-field col s4">
67     <label ><b>BMI(10-100)</b></label>
68     <br>
69     <input name = "6" placeholder="Body Mass Index">
70
71 </div>
72 <div class="input-field col s4">
73     <label for="last_name"><b>DiabetesPedigreeFunction (0-2.5)</b></label>
74     <br>
75     <input id="last_name" name="7" placeholder="DiabetesPedigreeFunction" type="text" class="validate">
76 </div>
77 <div class="input-field col s4">
78     <label for="_name"><b>Age (10-120)</b></label>
79     <br>
80     <input id="_name" name="8" placeholder="Age" type="text" class="validate">
81 </div>
82
83 </div>
84
85 <div class="row center">
86
87     <button type="submit" class="btn-large waves-effect waves-light light-green">Submit and predict probability</button>
88
89 </div>
90 </form>
91 </div>
92
93 <br>
94
95
96 <br><br>
97 </div>
98 </div>
99
100 <footer class="page-footer light-green lighten-1" style="position: fixed;left: 0;bottom: 0; width: 100%;">
101 <div class="footer-copyright light-green lighten-1">
102     <div class="container" style="padding-bottom: 1rem; position: fixed; font-size: 1.2rem; margin-left: 5rem;">
103         <p>&copy; Muhammed Salih</p>
104         <p>msalihki@gmail.com</p>
105     </div>
106 </div>
107 </footer>
108
109
110 <!-- Scripts-->
111 <script src="https://code.jquery.com/jquery-2.1.1.min.js"></script>
112 <script src="js/materialize.js"></script>
113 <script src="js/init.js"></script>
114
115 </body>
116 </html>
117
```

File Edit Selection View Go Run Terminal Helpresult.html - Visual Studio Code

Copy\_of\_Pima\_Indian\_Diabetes\_Project.ipynb Mapp.py Mindex.htmlresult.html X

C:\Users\> MUHAMMED SALIH K > DATASCIENCE NOTES > DATA SCIENCE > Project > Project\_1 > templates > result.html > ...

```
1  |<DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
5  <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0"/>
6  <title>Diabetes prediction</title>
7
8  <!-- CSS -->
9  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
10 <link href="/static/css/materialize.css" type="text/css" rel="stylesheet" media="screen,projection"/>
11 <link href="/static/css2/style.css" type="text/css" rel="stylesheet" media="screen,projection"/>
12 </head>
13
14 <body>
15 <nav class="light-green lighten-1" role="navigation">
16   <div class="nav-wrapper container"><a id="logo-container" href="/" class="brand-logo">Diabetes Prediction</a>
17   <ul class="right hide-on-med-and-down">
18     <li><a href="/">Home</a></li>
19   </ul>
20 </nav>
21
22
23
24
25
26
27   <div class="row" style="margin:15% 0% 0% 10%">
28     <h3 >{{pred}}</h3>
29   </div>
30
31   <br>
32
33   <br><br>
34 </div>
35 </div>>
36
37
38 <footer class="page-footer light-green lighten-1" style="position: fixed;left: 0;bottom: 0; width: 100%;>
39   <div class="footer-copyright light-green lighten-1">
40     <div class="" style="padding-bottom: 1rem; position: fixed; font-size: 1.2rem; margin-left: 5rem;">
41       <p>&copy; Muhammed Salih</p>
42       <p>msalihk1@gmail.com</p>
43     </div>
44   </div>
45 </footer>
46
47
48 <!-- Scripts-->
49 <script src="https://code.jquery.com/jquery-2.1.1.min.js"></script>
50 <script src="/js/materialize.js"></script>
51 <script src="/js/init.js"></script>
52
53 </body>
54 </html>
55
56
```

main 0 1 tabnine starter



## User Interface:

Are you worried that you might be Diabetic?

Return to home

# Predict diabetes (using AI in the background)

Predict the probability that you may be diabetic

<div>Pregnancies (0-15) No. of Pregnancies</div> <div></div>	<div>Glucose (40-250) Glucose level in sugar</div> <div></div>	<div>BloodPressure (20-140) BloodPressure</div> <div></div>
<div>SkinThickness (5-80) SkinThickness</div> <div></div>	<div>Insulin (0-1000) Insulin level</div> <div></div>	<div>BMI(10-100) Body Mass Index</div> <div></div>
<div>DiabetesPedigreeFunction (0-2.5) DiabetesPedigreeFunction</div> <div></div>	<div>Age (10-120) Age</div> <div></div>	

SUBMIT AND PREDICT PROBABILITY

© Muhammed Salih

msalihk1@gmail.com

Put the values and click submit and predict probability.

## Model Deployment:

- I have done the project in my local system and push all the files to my github ([https://github.com/msalihk1git/Project\\_1.git](https://github.com/msalihk1git/Project_1.git)).
- Deployed in heroku(<https://git.heroku.com/aipredictiondiabetes.git>)