**AI BASED DIABETES PREDICTION SYSTEM**

**Team member**

Name: PL.Veeramani

Regno:912421106019

Gmailid:palaniappanveeramani@gmail.com

NM id:au912421106019

**PHASE 2 Submission Document**

**Introduction**

* Diabetes is a chronic disease that directly affects the pancreas, and the body is incapable of producing insulin.It is mainly responsible for maintaining the blood glucose level.
* Many factors, such as excessive body weight, physical inactivity, high blood pressure, and abnormal cholesterol level, can cause a person get affected by diabetes.
* To use machine learning classification methods, that is, decision tree, SVM, Random Forest, Logistic Regression, KNN, and various ensemble techniques, to determine which algorithm produces the best prediction results.
* In this paper, we have employed machine learning and explainable AI techniques to detect diabetes.

**Content for Project Phase 2**

Consider exploring innovative techniques such as ensemble methods and deep learning architectures to improve the prediction system's accuracy and robustness.

**Data source**

AI-powered diabetes prediction system that leverages machine learning algorithms to analyze medical data and predict the likelihood of an individual developing diabetes, providing early risk assessment and personalized preventive measures.

**DatasetLink:**[**https://www.kaggle.com/datasets/mathchi/diabetes-data-set**](https://www.kaggle.com/datasets/mathchi/diabetes-data-set)



**Data Collection**

We need a dataset containing medical features such as glucose levels, blood pressure, BMI, etc., along with information about whether the individual has diabetes or not.

**Data Preprocessing:**

The medical data needs to be cleaned, normalized, and prepared for training machine learning models.

**Feature Selection:**

We will select relevant features that can impact diabetes risk prediction.

**Model Selection:**

We can experiment with various machine learning algorithms like Logistic Regression, Random Forest, and Gradient Boosting.

**Evaluation:**

We will evaluate the model's performance using metrics like accuracy, precision, recall, F1-score, and ROC-AUC.

**Iterative Improvement:**

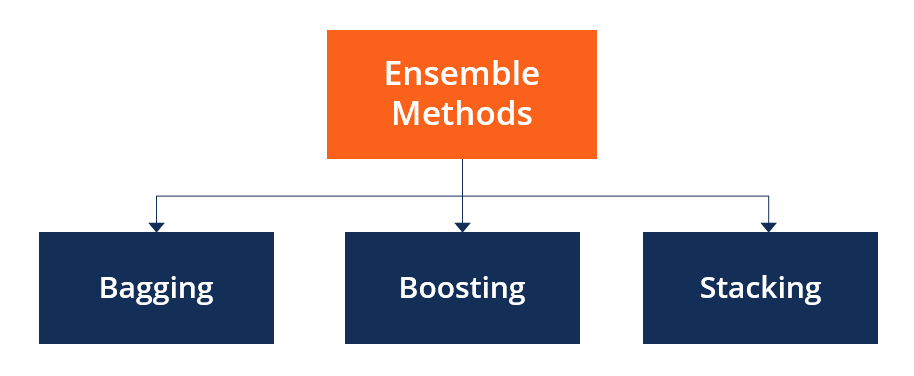
We will fine-tune the model parameters and explore techniques like feature engineering to enhance prediction accuracy.



**Ensemble methods to improve the prediction system's accuracy**

Ensemble methods are a type of machine learning technique that involve combining multiple models to improve the accuracy and robustness of predictions. Ensemble methods are typically used in situations where a single model may not be sufficient or where different models may have complementary strengths.

**Main Types of Ensemble Methods**

****

**1. Bagging**

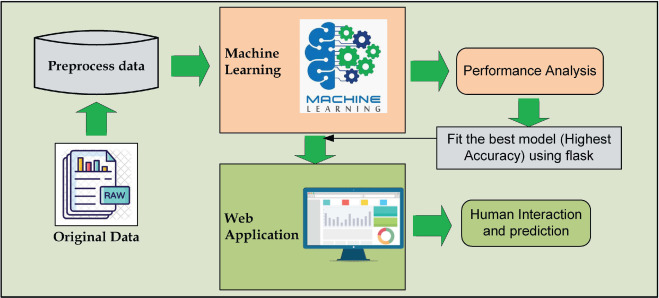
Bagging, the short form for bootstrap aggregating, is mainly applied in classification and [regression](https://corporatefinanceinstitute.com/resources/data-science/regression-analysis/). It increases the accuracy of models through decision trees, which reduces variance to a large extent. The reduction of variance increases accuracy, eliminating overfitting, which is a challenge to many predictive models.

**2. Boosting**

Boosting is an ensemble technique that learns from previous predictor mistakes to make better predictions in the future. The technique combines several weak base learners to form one strong learner, thus significantly improving the predictability of models. Boosting works by arranging weak learners in a sequence, such that weak learners learn from the next learner in the sequence to create better predictive models.

**3. Stacking**

Stacking, another ensemble method, is often referred to as stacked generalization. This technique works by allowing a training algorithm to ensemble several other similar learning algorithm predictions. Stacking has been successfully implemented in regression, density estimations, distance learning, and classifications.



**Program**

AI Based Diabetes Prediction System

# Import libraries

**import** numpy as np # for linear algebra

**import** pandas as pd # for data processing, CSV file I/O (e.g. pd.read\_csv)

**import** seaborn as sns # for data visualization

**import** matplotlib.pyplot as plt # to plot data visualization charts

**from** collections **import** Counter

**import** os

# Modeling Libraries

**from** sklearn.metrics **import** confusion\_matrix, accuracy\_score, precision\_score

**from** sklearn.preprocessing **import** QuantileTransformer

**from** sklearn.linear\_model **import** LogisticRegression

**from** sklearn.neighbors **import** KNeighborsClassifier

**from** sklearn.tree **import** DecisionTreeClassifier

**from** sklearn.ensemble **import** RandomForestClassifier, AdaBoostClassifier, GradientBoostingClassifier

**from** sklearn.model\_selection **import** GridSearchCV, cross\_val\_score, StratifiedKFold, learning\_curve, train\_test\_split

**from** sklearn.svm **import** SVC

**Importing the Dataset**

# Importing the dataset from Kaggle

data = pd.read\_csv("../input/pima-indians-diabetes-database/diabetes.csv")

# First step is getting familiar with the structure of the dataset

data.info()

**Output**

<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 BloodPressure 768 non-null int64

3 SkinThickness 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

**Code**

# Showing the top 5 rows of the dataset

data.head()

**Output**

****

**Filling the Missing Values** **Code**

# Exploring the missing values in the diabetes dataset

data.isnull().sum()

**Output**

Pregnancies 0

Glucose 0

BloodPressure 0

SkinThickness 0

Insulin 0

BMI 0

DiabetesPedigreeFunction 0

Age 0

Outcome 0

dtype: int64

**Code**

# Replacing 0 values with the mean of that column

# Replacing 0 values of Glucose

data['Glucose'] = data['Glucose'].replace(0, data['Glucose'].median())

# Filling 0 values of Blood Pressure

data['BloodPressure'] = data['BloodPressure'].replace(0, data['BloodPressure'].median())

# Replacing 0 values in BMI

data['BMI'] = data['BMI'].replace(0, data['BMI'].mean())

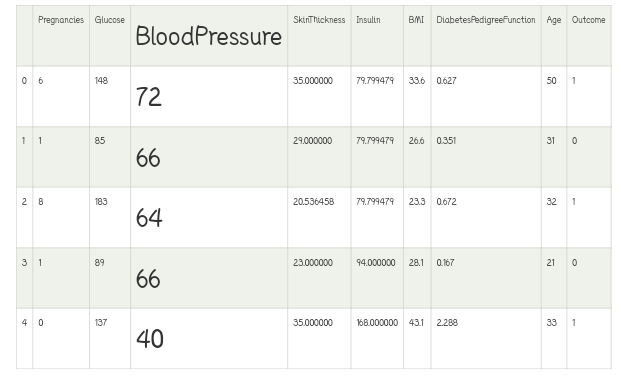
# Replacing the missing values of Insulin and SkinThickness

data['SkinThickness'] = data['SkinThickness'].replace(0, data['SkinThickness'].mean())

data['Insulin'] = data['Insulin'].replace(0, data['Insulin'].mean())

data.head()

**Output**

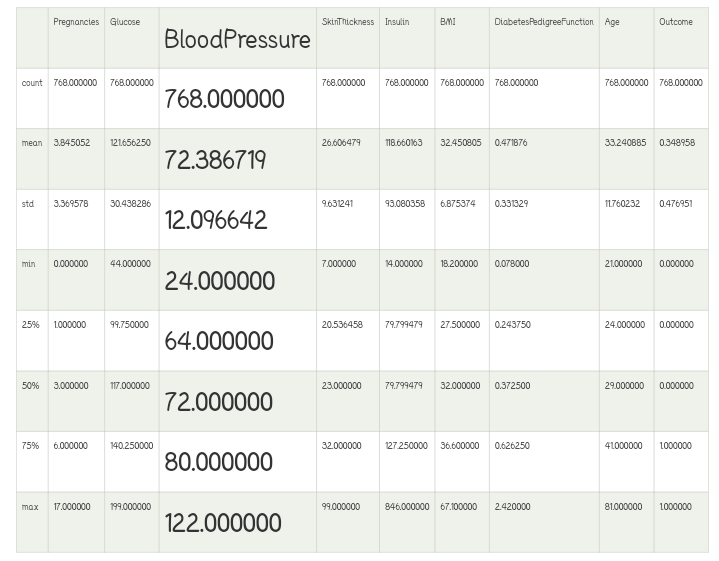
****

**Code**

# Reviewing the dataset statistics

data.describe()

**Output**

****

**Pregnancy** **Code**

# Exploring Pregnancy and target variables together

plt.figure(figsize = (10, 8))

# Plotting density function graph of the pregnancies and the target variable

kde = sns.kdeplot(data["Pregnancies"][data["Outcome"] == 1], color = "Red", shade = True)

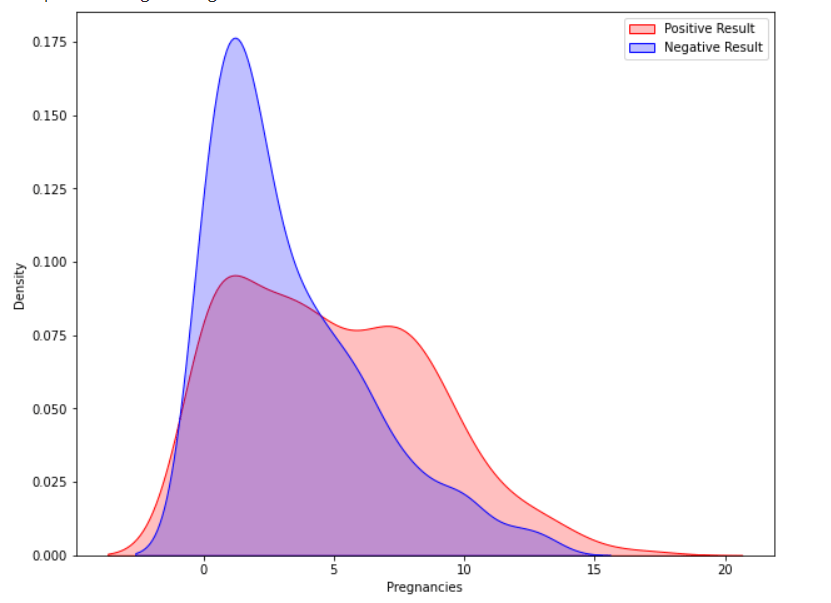
kde = sns.kdeplot(data["Pregnancies"][data["Outcome"] == 0], ax = kde, color = "Blue", shade= True)

kde.set\_xlabel("Pregnancies")

kde.set\_ylabel("Density")

kde.legend(["Positive Result", "Negative Result"])

**Output**



**Glucose code**

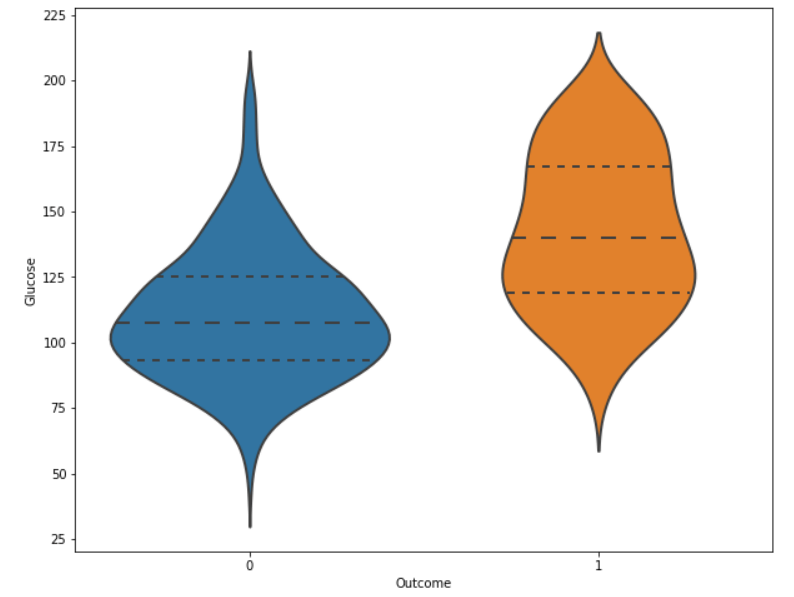
# Exploring the Glucose and the Target variables together

plt.figure(figsize = (10, 8))

sns.violinplot(data = data, x = "Outcome", y = "Glucose",

               split = True, inner = "quart", linewidth = 2)

**Output**



**Code**

# Exploring the density function plot of the Glucose levels

plt.figure(figsize = (10, 8))

kde = sns.kdeplot(data["Glucose"][data["Outcome"] == 1], color = "Red", shade = True)

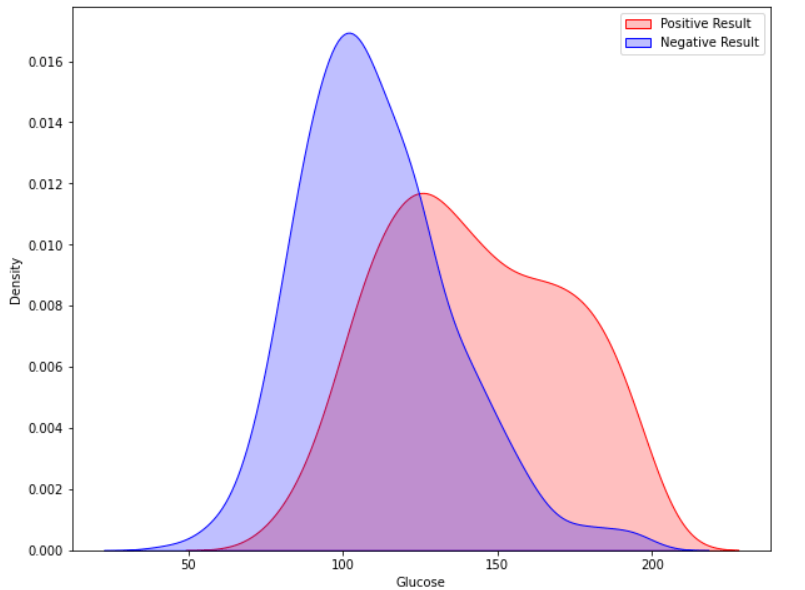
kde = sns.kdeplot(data["Glucose"][data["Outcome"] == 0], ax = kde, color = "Blue", shade= True)

kde.set\_xlabel("Glucose")

kde.set\_ylabel("Density")

kde.legend(["Positive Result","Negative Result"])

**Output**



**Future for diabetes prediction system**

* Researchers are motivated to create a Machine Learning methodology that can predict diabetes in the future.
* Exploiting Machine Learning Algorithms (MLA) is essential if healthcare professionals are able to identify diseases more effectively.

**Conclusions**

* One of the risks during pregnancy is diabetes. It will have to be diagnosed to avoid problems.
* An increase in glucose levels is strongly correlated to a rise in diabetes.
* In this paper, an automatic diabetes prediction system using various machine learning approaches has been proposed. The open‐source Pima Indian and a private dataset of female Bangladeshi patients have been used in this work.
* This research paper reported different performance metrics, that is, precision, recall, accuracy, F1 score, and AUC for various machine learning and ensemble techniques.
* There are some future scopes of this work, for example, we recommend getting additional private data with a larger cohort of patients to get better results.