

# Dia.B

Prepared By:  
Omar Alhadi





# Agenda

**Part 1:** Introduction



**Part 2:** Methodology



**Part 3:** Results



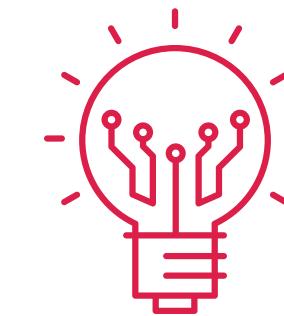
**Part 4:** Conclusion

# Introduction



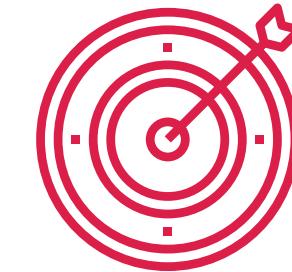
## Problem ?

- About one in seven U.S. adults has diabetes now, according to the Centers for Disease Control and Prevention. But by 2050, that rate could skyrocket to as many as one in three.



## Solutuation

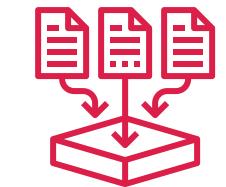
- In this project, I have built a classifier to predict Diabetes disease.
- I have implemented different classification models on the dataset and evaluated the performance of the models.



## Objectives

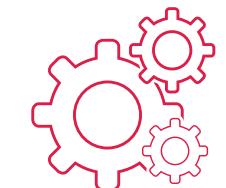
Build a classifier to predict Diabetes disease

# Methodology :



## Data Sources:

- The diabetes data set was originated from UCI Machine Learning.



## Tools



NumPy



pandas



matplotlib



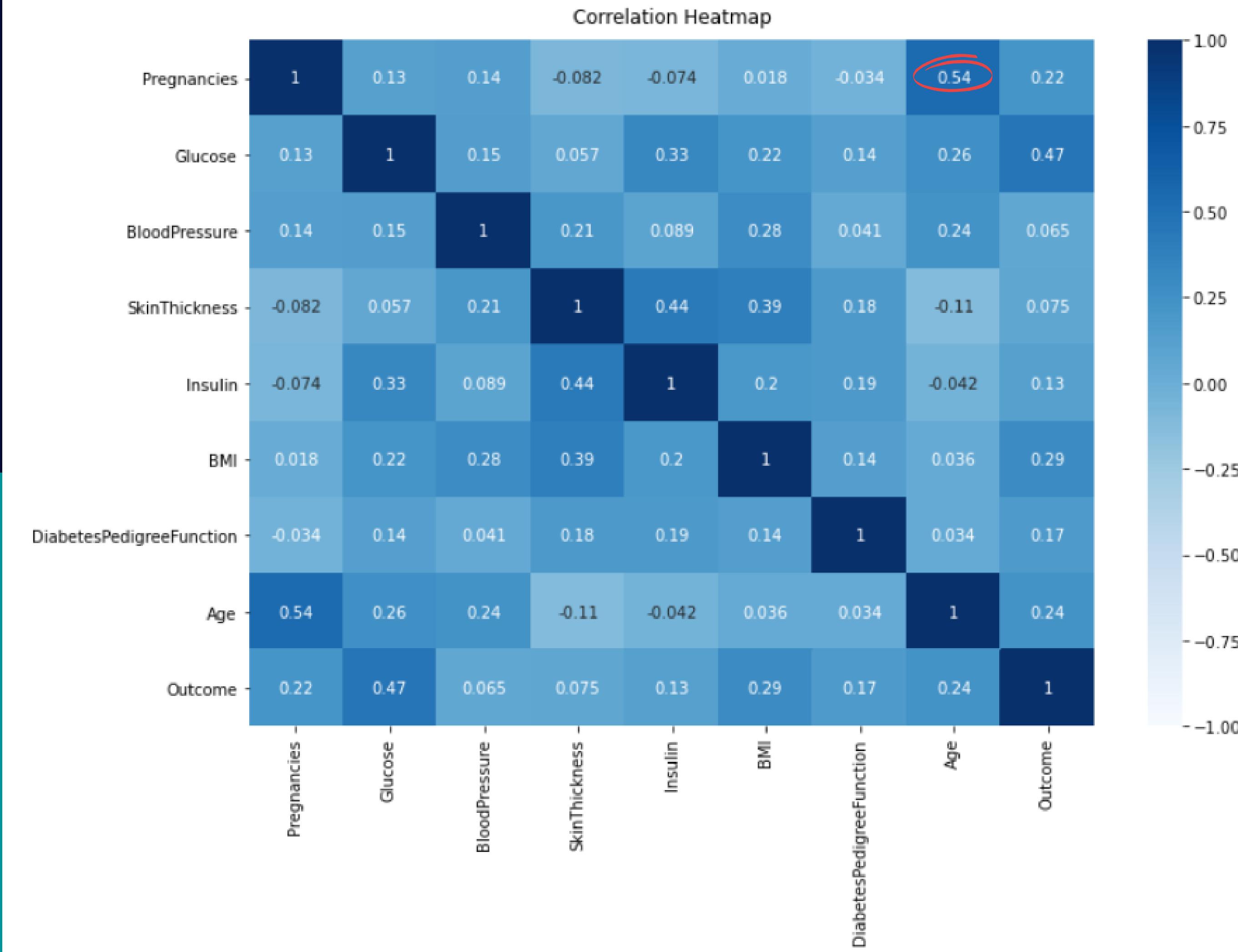
seaborn

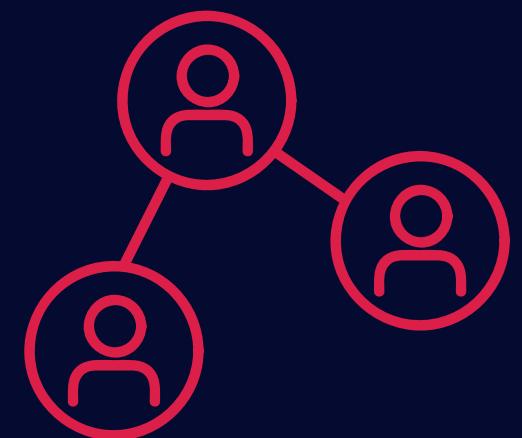


scikit  
learn  
machine learning in Python

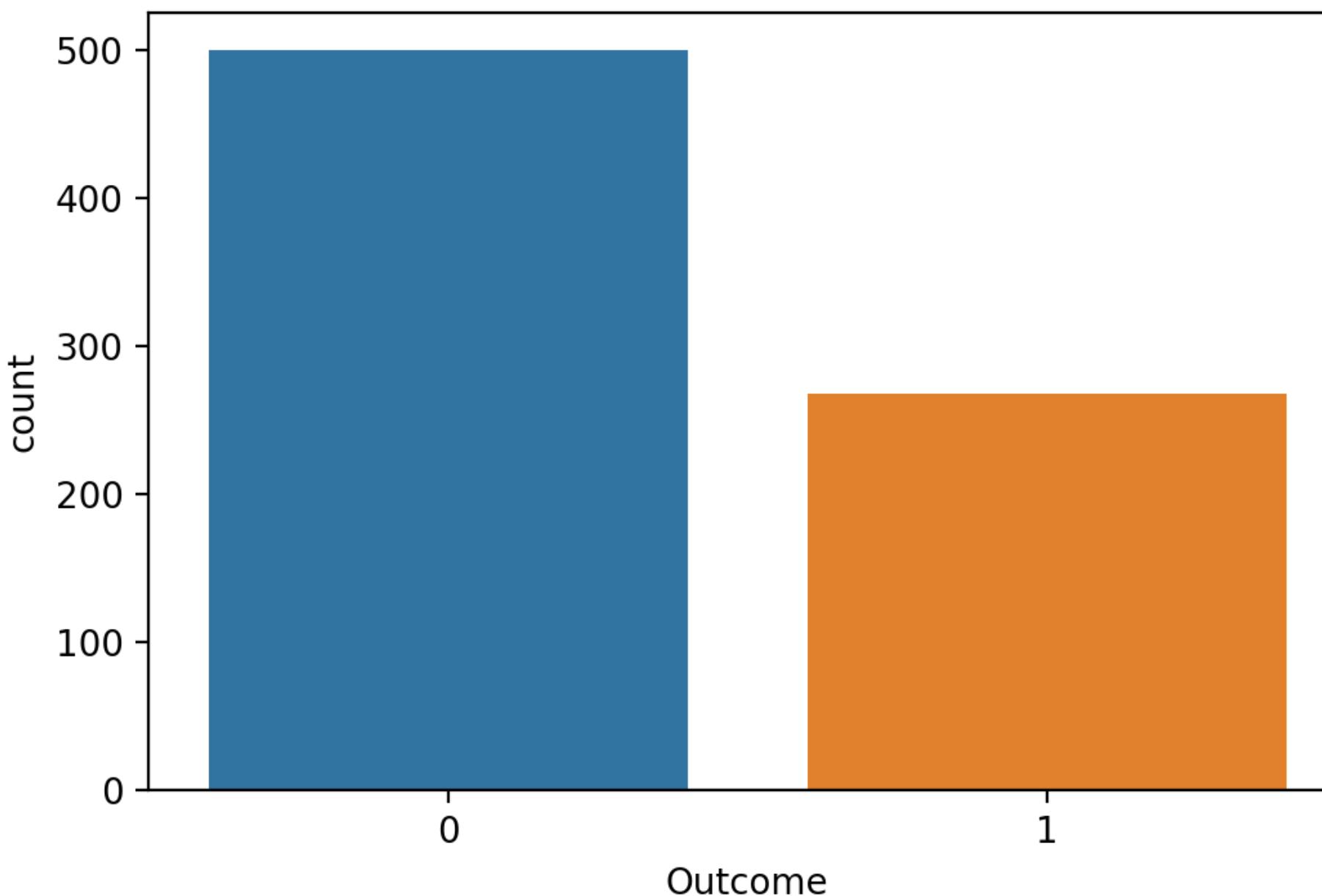
# Results :

## Brief Insight From Data





# Number of people suffering by diabetes



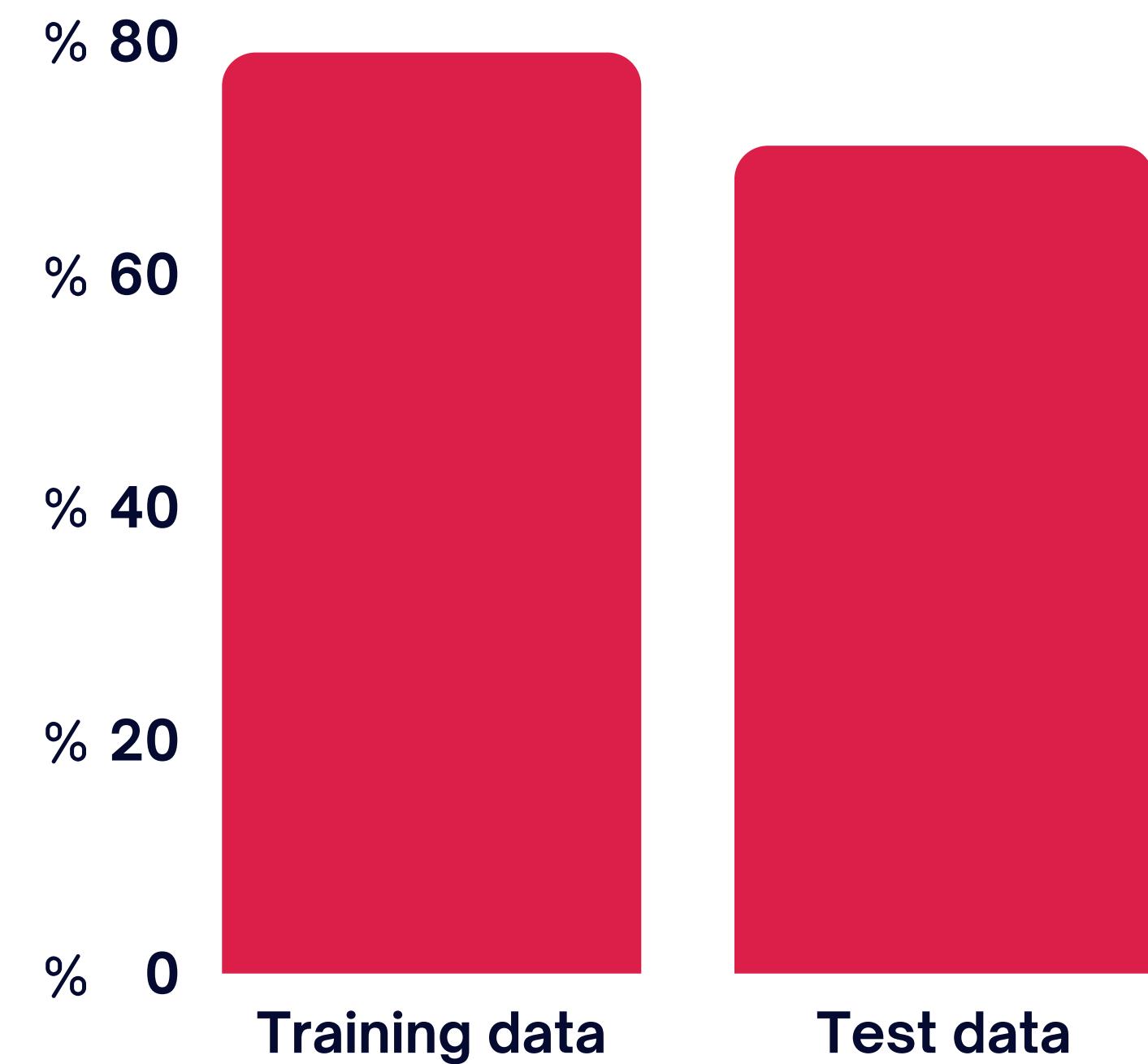
# K Nearest Neighbors Classifier Model

0.7914

Accuracy score of Training data

0.7186

Accuracy score of Test data



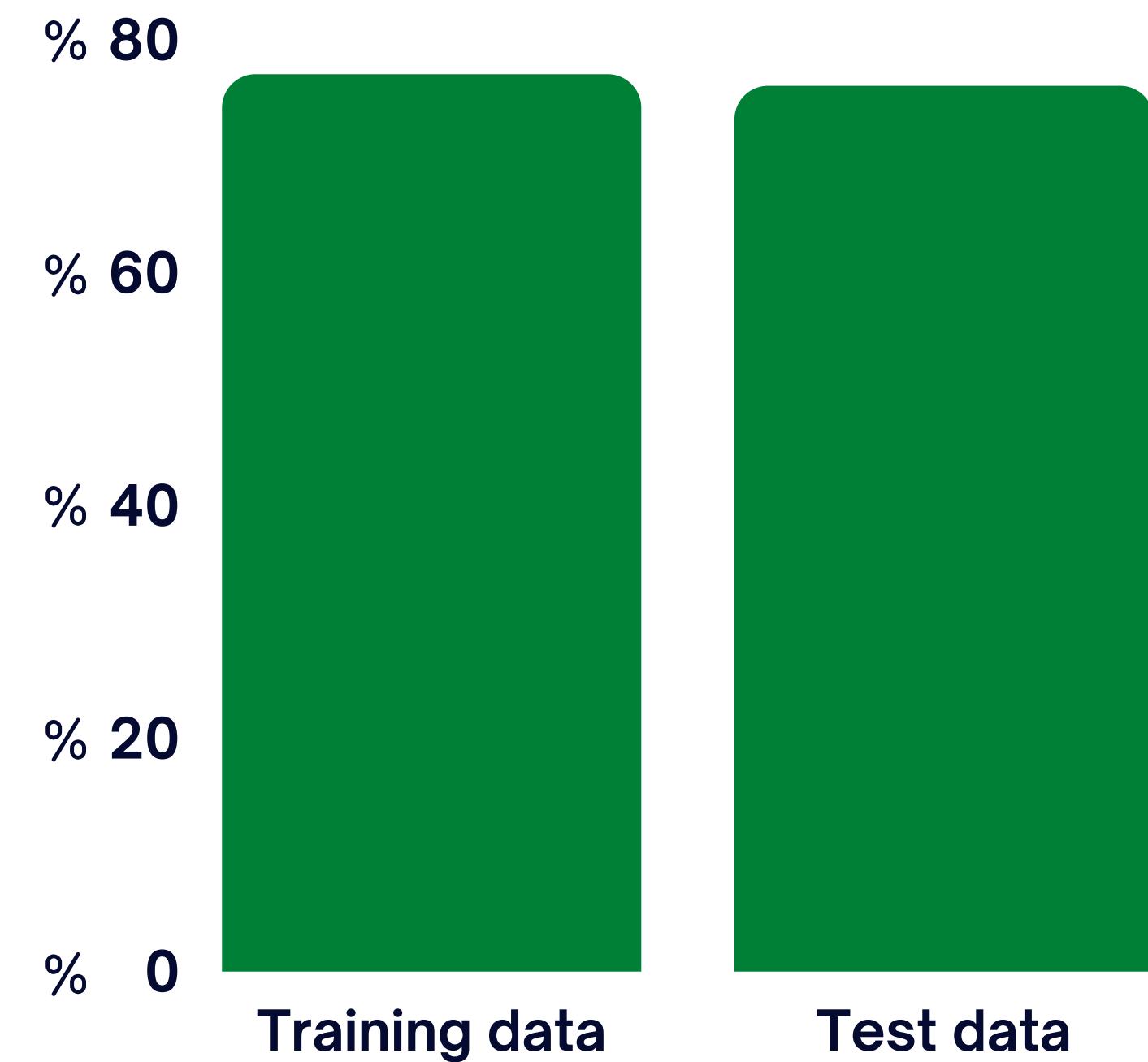
# Logistic Regression Model

0.7765

Accuracy score of Training data

0.7662

Accuracy score of Test data



# Decision Tree Classifier Model

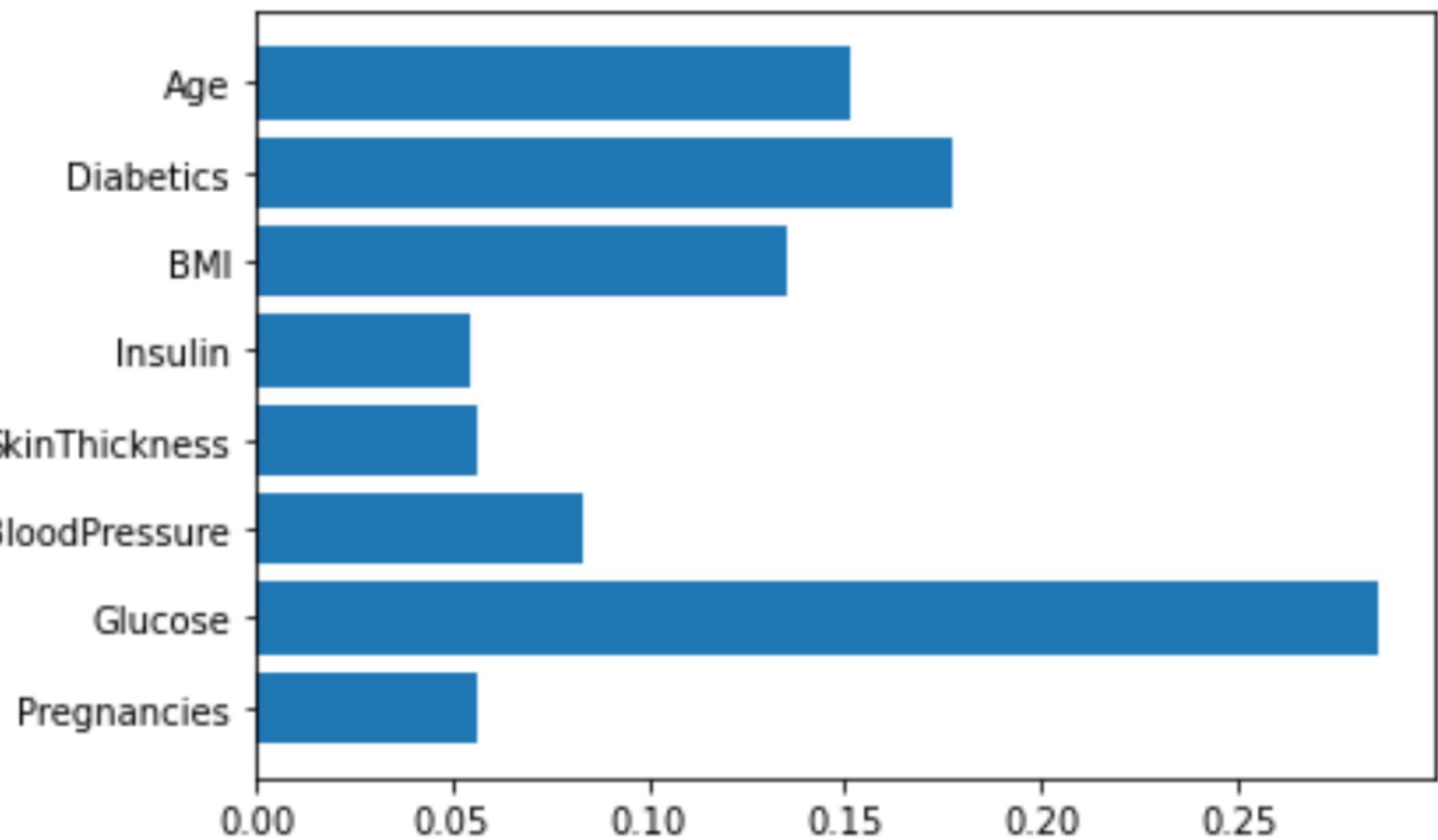
1.0

Accuracy score of Training data

0.6536

Accuracy score of Test data

Features Importance Bar Plot



# Random Forest Classifier Model

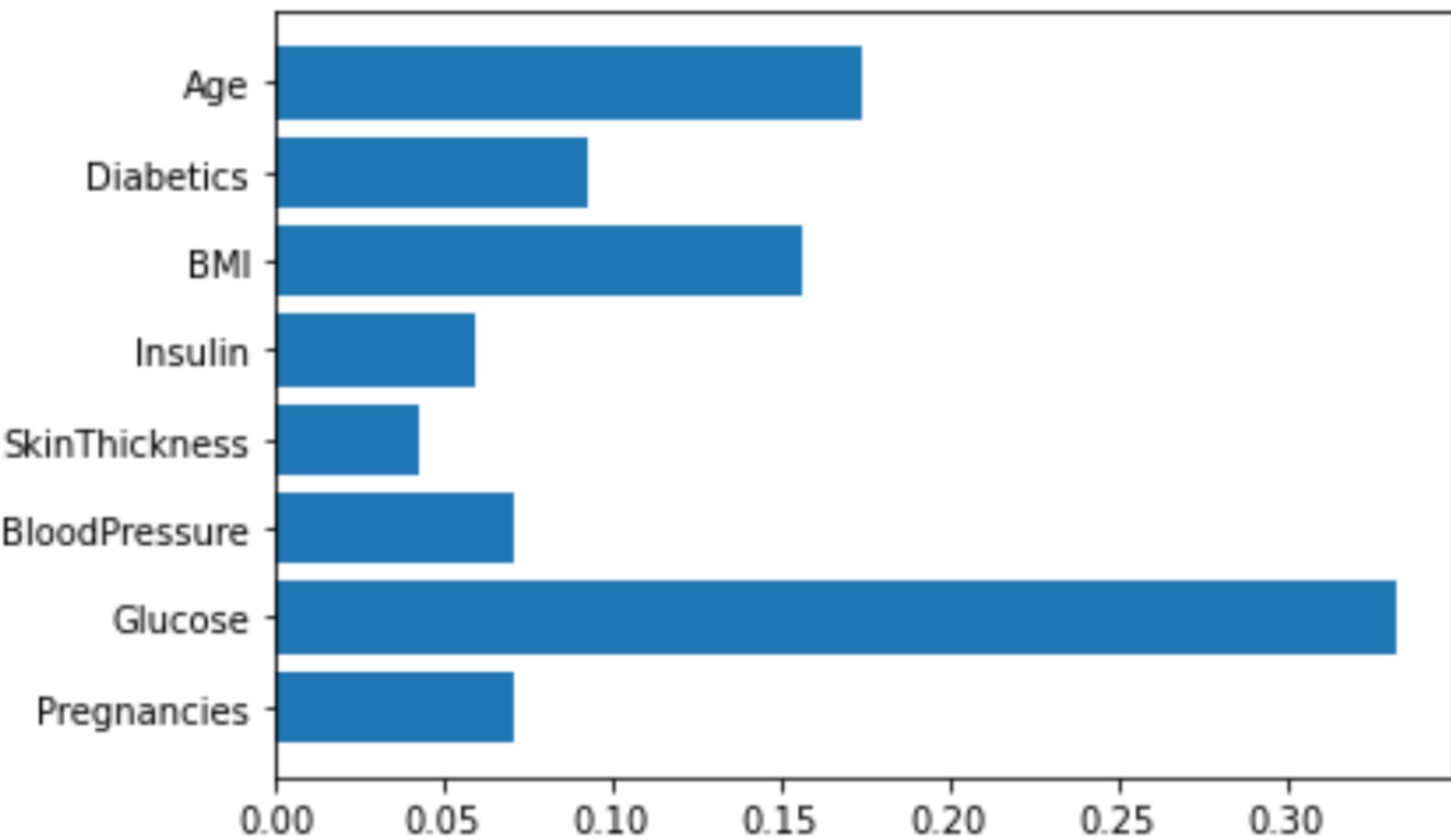
0.8566

Accuracy score of Training data

0.7748

Accuracy score of Test data

Features Importance Bar Plot



# Gradient Boosting Classifier Model

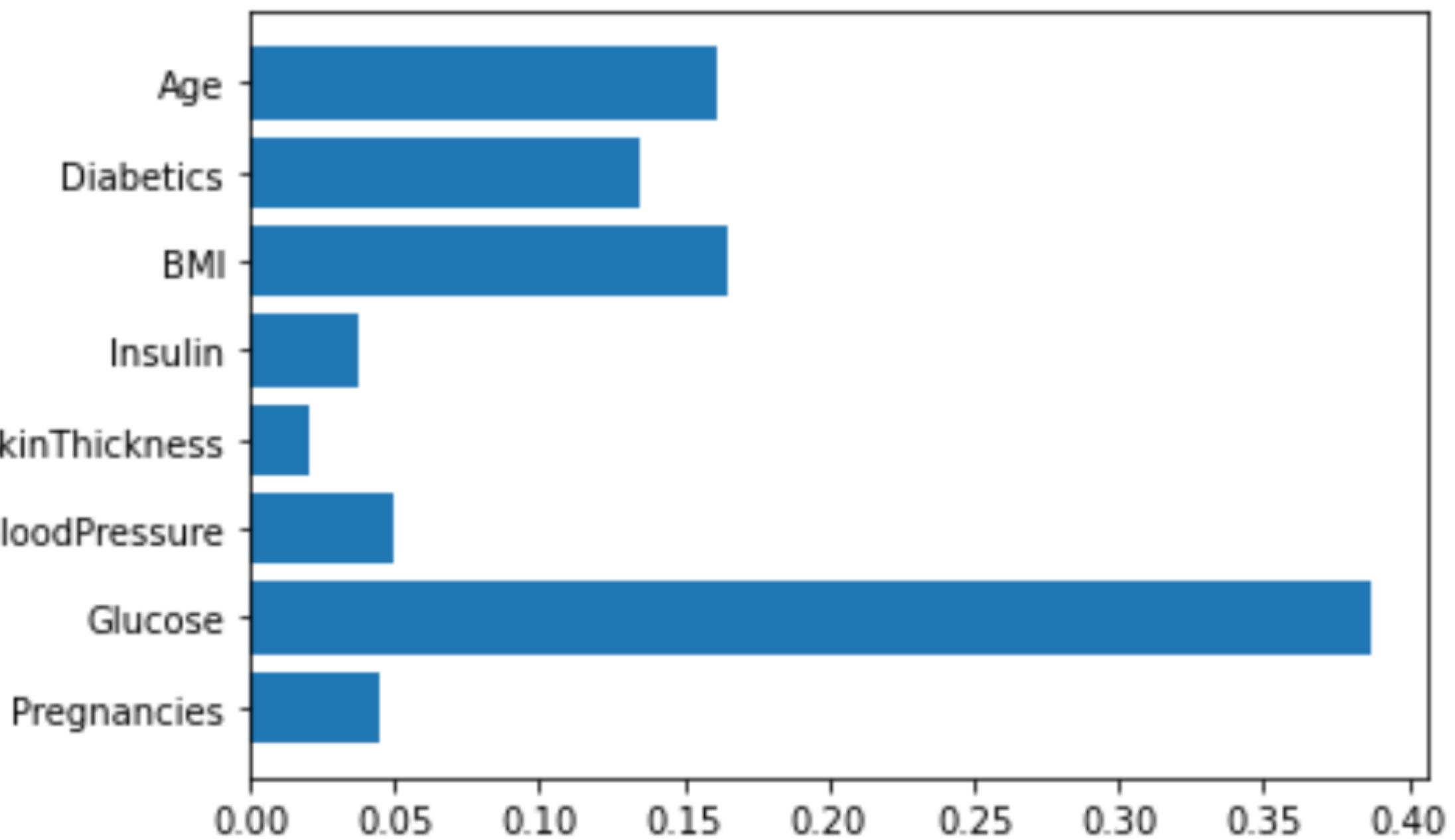
0.9199

Accuracy score of Training data

0.7316

Accuracy score of Test data

Features Importance Bar Plot



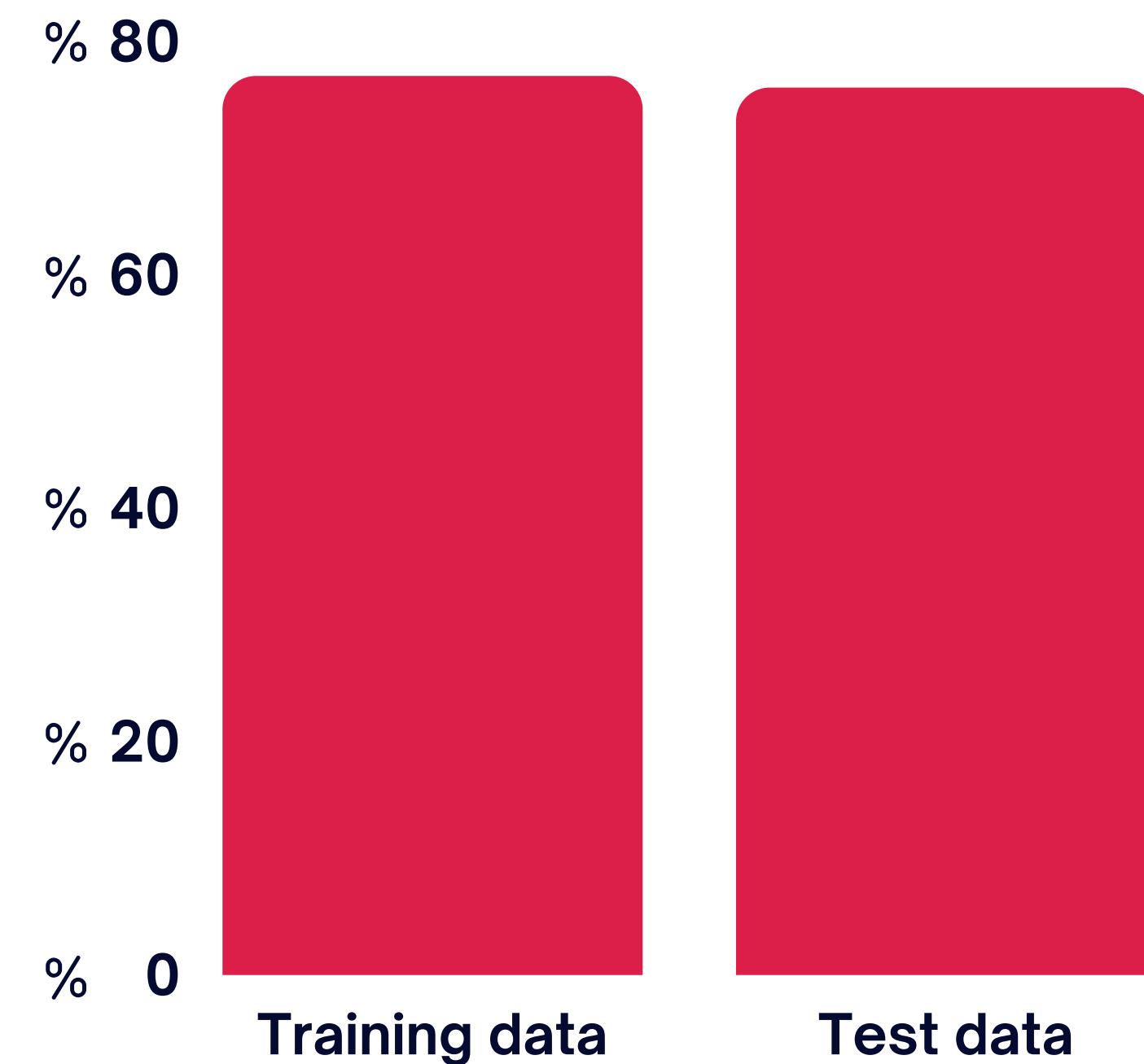
# Support Vector Machines (SVM) Model

0.7746

Accuracy score of Training data

0.7619

Accuracy score of Test data



# Conclusion:

Voting Classifier Model shows :

**0.72**

K Nearest Neighbors

**0.77**

Logistic Regression

**0.71**

Decision Tree

**0.76**

Random Forest

**0.76**

Gradient Boosting

**0.76**

Support Vector Machines (SVM)

# Thank you!

Dia.B