Table of Contents

1 Introduction

This study uses the Pima Indians Diabetes Dataset to test Explainable Machine Learning (ML) approaches for diabetes prediction. Diabetes is a widespread health issue that requires early detection and treatment to avoid major complications. This field requires model explainability to give healthcare practitioners trust and understanding in machine learning model predictions.

2 Problem Definition and Dataset Selection

Healthcare diagnostics, namely diabetes prediction utilising health factors, is the chosen problem domain.

2.1 Problem Definition

Diabetes affects many people worldwide and can have catastrophic implications if not diagnosed and treated. This project aims to construct an interpretable machine learning model that can accurately predict diabetes using patient data. Medical practitioners must trust and comprehend model predictions, hence model interpretability is crucial.

2.2 Significance of the Problem

Diabetes can cause cardiovascular disease, renal failure, and blindness. Due to its widespread impact on public health, diabetes must be accurately diagnosed. Accurate prognoses can improve patient management and treatment, improving health outcomes and lowering healthcare costs. Healthcare providers can make informed decisions by understanding the model's predictions. Controlling the condition and avoiding complications requires early detection and treatment. In healthcare, predictive models must be interpretable to build trust and give practitioners with practical insights.

2.3 Dataset

Visit https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database to access the UCI Machine Learning Repository's "Pima Indians Diabetes Database". The dataset has 768, 8-variable occurrences. Number of pregnancies, glucose concentration, blood pressure, skin thickness, insulin levels, BMI, diabetes pedigree function, and age. "Outcome" indicates whether the patient has diabetes.

2.4 Importance of Model Explainability

Model interpretability is important in healthcare because machine learning model results might affect patient well-being. Healthcare practitioners must comprehend a model's predictions to trust its advice. Explainable machine learning (ML) methods can reveal the most important diabetes predictors. Additionally, this helps healthcare providers make more informed therapy decisions.

3 Methodology

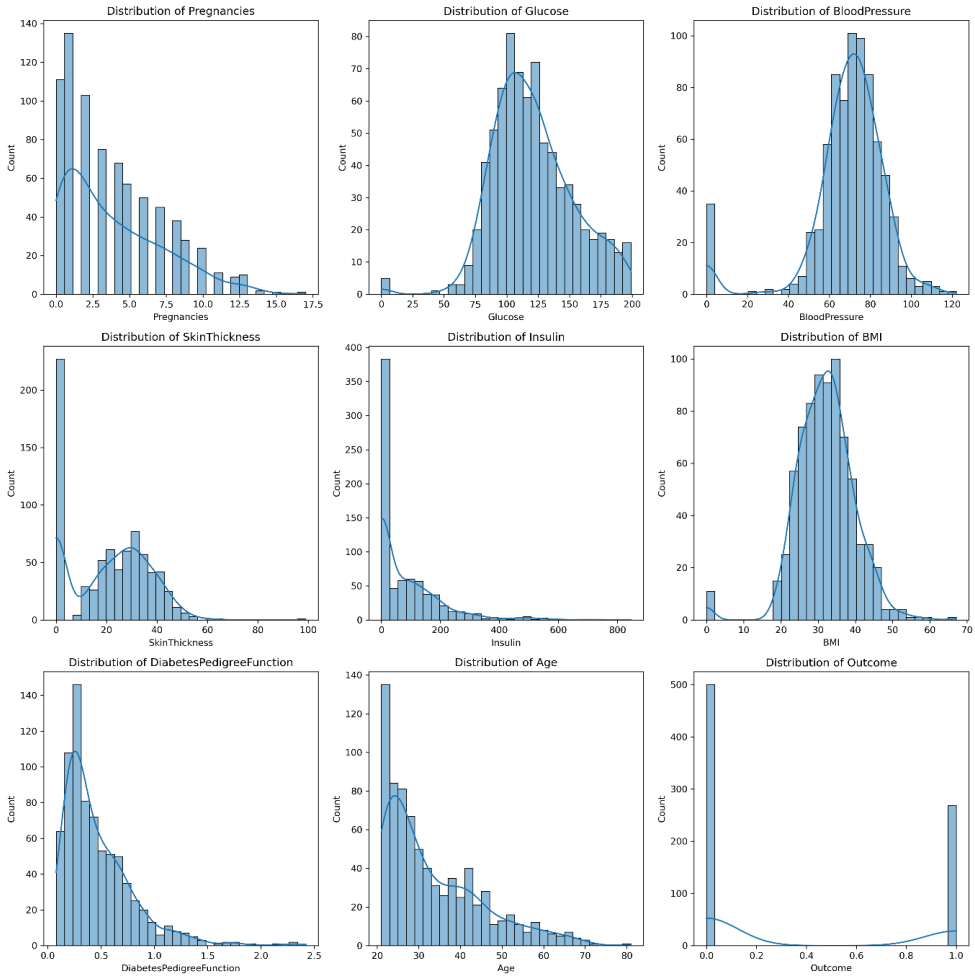
It involves building a machine learning model and improving its performance with Bayesian hyperparameter tuning. The implementation will maintain a clean Python notebook with detailed comments and explanations for clarity and repeatability.

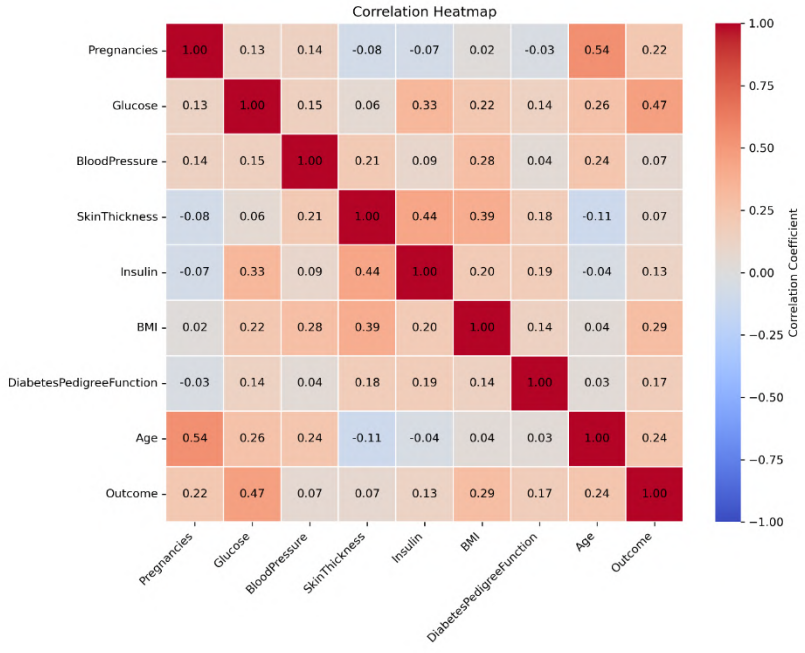
1. Understanding and preprocessing of data

#	Column	Description	Data type	Count	Null values
0	Pregnancies	Number of times pregnant	int	768	0
1	Glucose	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	int	768	0
2	BloodPressure	Diastolic blood pressure (mm Hg)	int	768	0
3	SkinThickness	Triceps skin fold thickness (mm)	int	768	0
4	Insulin	2-Hour serum insulin (mu U/ml)	int	768	0
5	BMI	Body mass index (weight in kg/(height in m)^2)	float	768	0
6	DiabetesPedigreeFunction	Diabetes pedigree function	float	768	0
7	Age	Age (years)	int	768	0
8	Outcome	Class variable (0 or 1) 268 of 768 are 1(Diabetes), the others are 0(No diabetes)	int	768	0

```
Summary of each numeric variable:
                            min     max    range
Pregnancies               0.000   17.00   17.000
Glucose                   0.000  199.00  199.000
BloodPressure             0.000  122.00  122.000
SkinThickness             0.000   99.00   99.000
Insulin                   0.000  846.00  846.000
BMI                       0.000   67.10   67.100
DiabetesPedigreeFunction  0.078    2.42    2.342
Age                      21.000   81.00   60.000
Outcome                   0.000    1.00    1.000

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


Distribution of Pregnancies
Pregnancies
Distribution of Glucose
Glucose
Distribution of BloodPressure
BloodPressure
Distribution of SkinThickness
SkinThickness
Distribution of Insulin
Insulin
Distribution of BMI
BMI
Distribution of DiabetesPedigreeFunction
DiabetesPedigreeFunction
Distribution of Age
Age
Distribution of Outcome
Outcome
Count


Correlation Heatmap
Pregnancies 1.00 0.13 0.14 -0.08 -0.07 0.02 -0.03 0.54 0.22
Glucose 0.13 1.00 0.15 0.06 0.33 0.22 0.14 0.26 0.47
BloodPressure 0.14 0.15 1.00 0.21 0.09 0.28 0.04 0.24 0.07
SkinThickness -0.08 0.06 0.21 1.00 0.44 0.39 0.18 -0.11 0.07
Insulin -0.07 0.33 0.09 0.44 1.00 0.20 0.19 -0.04 0.13
BMI 0.02 0.22 0.28 0.39 0.20 1.00 0.14 0.04 0.29
DiabetesPedigreeFunction -0.03 0.14 0.04 0.18 0.19 0.14 1.00 0.03 0.17
Age 0.54 0.26 0.24 -0.11 -0.04 0.04 0.03 1.00 0.24
Outcome 0.22 0.47 0.07 0.07 0.13 0.29 0.17 0.24 1.00
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome
Correlation Coefficient
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

2. Splitting the data, re-sampling and scaling

X-train, y_train, X-test, and y_test are the training and testing sets. SMOTE is used to balance the training data (X_train and y_train) because class 0 and class 1 are imbalanced. Additionally, data numerical values differ. Min-Max scaling was performed to X_train and X_test after SMOTE.

```
# Feature and target split
X = data.drop("Outcome", axis=1)
y = data["Outcome"]
```

```
# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

```
# Scaling features
scaler = MinMaxScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

```
# Initialize the SMOTE object
smote = SMOTE(random_state=42)

# Fit and transform the training data
X_train_scaled_res, y_train_res = smote.fit_resample(X_train_scaled, y_train)

#Check balancing
y_train_res.value_counts()
```

Data	**X(Features)**		**y (Outcome)**			
	768		768 (Class 0 -500, Class 1 – 268)			
Split	X_train	X_test	y_train(70%)		y_test(30%)	
			Class 0	Class 1	Class 0	Class 1
	537(70%)	231(30%)	349	188	151	80
SMOTE	698	Not applicable	349 (50%)	349 (50%)	Not applicable	
Min-Max Scaling	Applied		Not applicable			

3. Model training and Bayesian optimization

I selected XGBoost classifier because its decision tree ensemble structure makes it SHAP-compatible and provides rapid, accurate, and interpretable model prediction explanations. XGBoost is one of the best SHAP models for explaining predictions in a meaningful and computationally efficient way due to its high performance and tree-specific optimisations.

```
# Define the model as XGBClassifier
model = XGBClassifier(use_label_encoder=False, eval_metric='mlogloss', random_state=42)
```

```
# Bayesian hyperparameter tuning
bayes_search = BayesSearchCV(model, param_space, n_iter=32, cv=3, random_state=42, n_jobs=-1)
bayes_search.fit(X_train_scaled_res, y_train_res)

# Best model after tuning
best_model_bayes = bayes_search.best_estimator_
```

4. Model Evaluation

Evaluate the model's performance using appropriate metrics (e.g., accuracy, precision, recall, F1-score).

```
# Predictions
y_pred = best_model_bayes.predict(X_test_scaled)
```

```
# Classification Report
print(classification_report(y_pred, y_test))
```

```
# Calculate the metrics
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted', zero_division=0)
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')
roc_auc = roc_auc_score(y_test, y_pred, average='weighted', multi_class='ovr')

# Print the metrics
print(f"Accuracy: {accuracy:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")
print(f"F1 Score: {f1:.2f}")
print(f"roc_auc: {f1:.2f}")
```

5. Model Explainability

a. Explainability techniques

SHAP (SHapley Additive exPlanations) is a method used to explain the output of a model by calculating the contribution of each feature to the prediction. They assist in comprehending the influence of each feature on the model's predictions.

```
# Calculate SHAP values
explainer = shap.TreeExplainer(best_model_bayes)
shap_values = explainer.shap_values(X_test)
```

LIME (Local Interpretable Model-agnostic Explanations) is a method used to produce explanations at a local level for individual predictions.

```
# Index of the sample to explain
i = 10  # You can change this index to explain other predictions

# Explain a single prediction
lime_exp = lime_explainer.explain_instance(
    data_row=X_test.iloc[i].values,             # Instance to explain (from test set)
    predict_fn=best_model_bayes.predict_proba,  # Prediction function (probabilities)
    num_features=7                              # Number of features to display
)
```

Feature importance: This refers to the process of determining the relative relevance of different features in influencing the predictions made by the model.

```
# Get feature importances from the model
importances = best_model_bayes.feature_importances_
```

Partial Dependence Plots (PDP) are used to illustrate the link between different characteristics and the projected outcome.

```
# Plot the Partial Dependence Plots with adjusted layout
display = PartialDependenceDisplay.from_estimator(
    best_model_bayes,           # Your trained model
    X_train_scaled_res,         # Scaled training data
    features,                   # Features to plot
    feature_names=X.columns,    # Feature names
    grid_resolution=10,         # Resolution of the grid (more points = more detail)
    ax=ax                       # Use the ax to control layout
)
```

6. Importance of Model Explainability in Healthcare Diagnostic

- Trust: Clinicians must possess a strong belief in the precision and dependability of the model's predictions in order to seamlessly incorporate them into their decision-making process.

- Actionability: The capacity to identify the specific characteristics that contribute to making forecasts can aid in identifying risk factors and provide suggestions for preventive measures.

- Ethical Compliance: Ensuring that the model follows ethical norms by avoiding the continuation of biases that exist in the data.

4 Findings

4.1 Model Performance

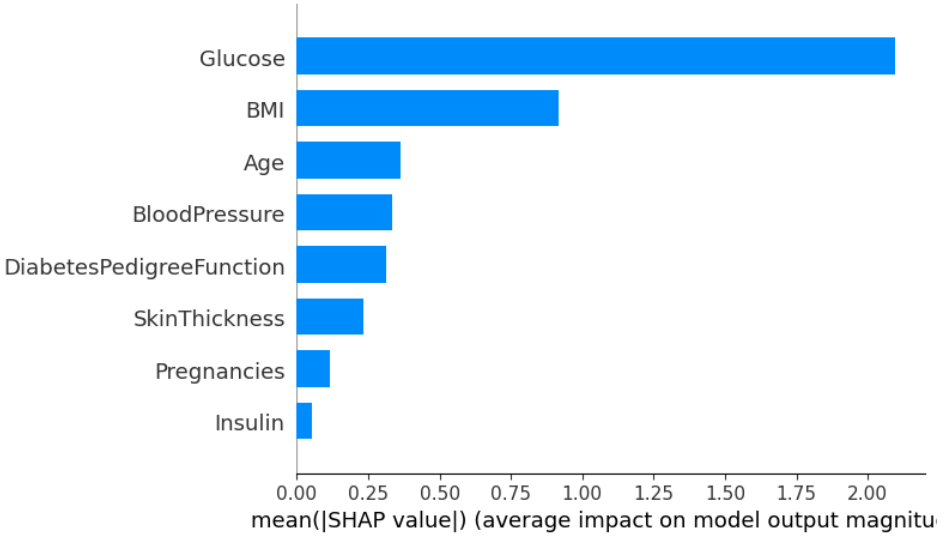
	precision	recall	f1-score	support
0	0.73	0.83	0.77	133
1	0.71	0.58	0.64	98
accuracy			0.72	231
macro avg	0.72	0.70	0.71	231
weighted avg	0.72	0.72	0.72	231

```
Accuracy: 0.72
Precision: 0.74
Recall: 0.72
F1 Score: 0.73
roc_auc: 0.73
```

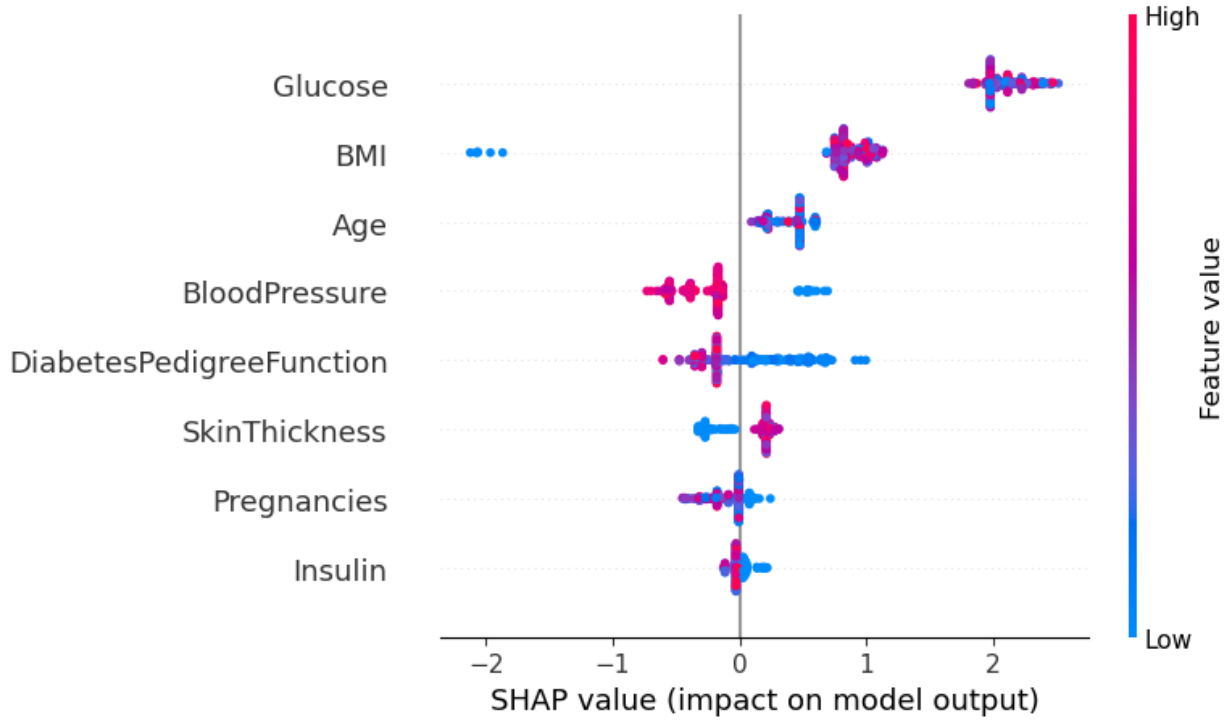
4.2 SHAP

4.2.1 SHAP Summary plot

According to the SHAP summary plots, Glucose, BMI, and Age have a significant impact on the development of Diabetes.

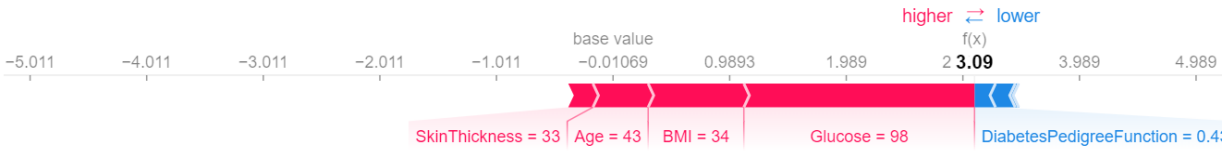


According to the SHAP summary plots, Glucose, BMI, and Age have a significant impact on the model output, especially when their feature values are high. The blood pressure exhibits high feature values but has a limited impact on the model's predictions.



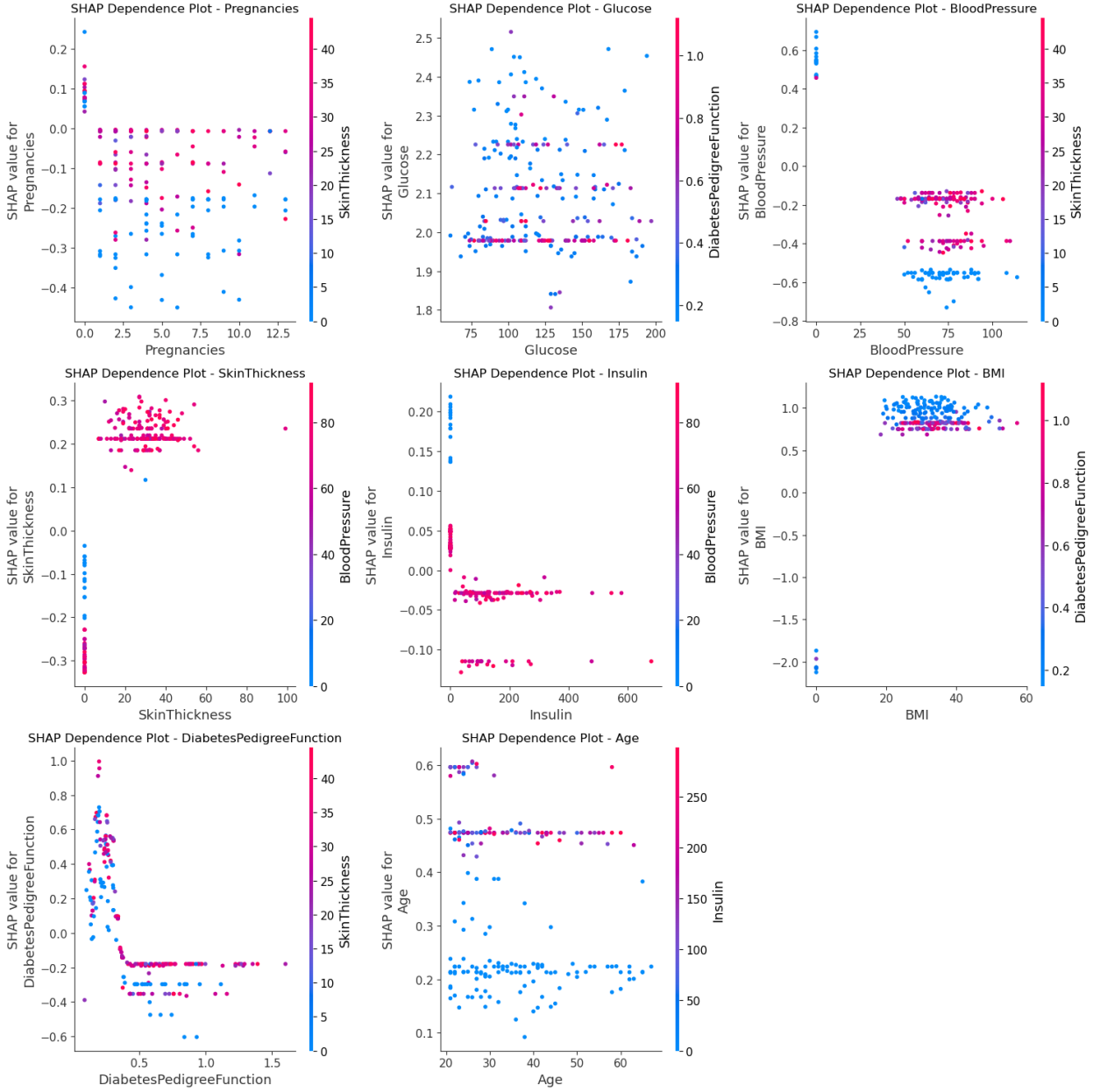
4.2.2 SHAP Force plot

A force plot is employed to elucidate a specific occurrence within a dataset.



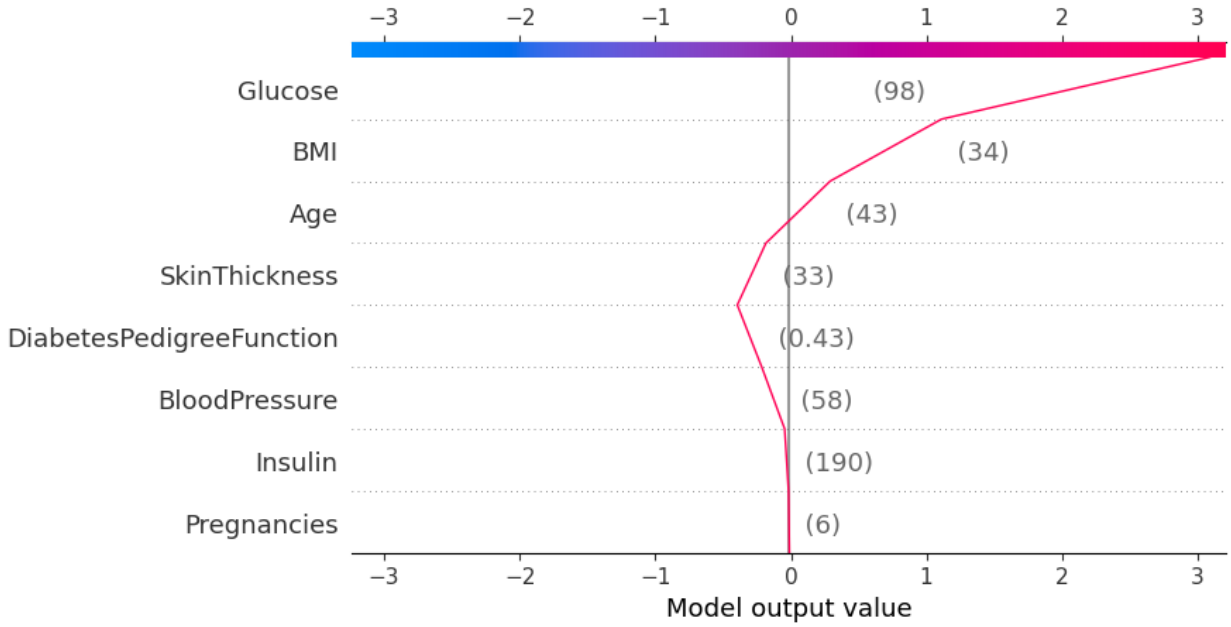
4.2.3 SHAP Dependance plot

A SHAP dependence plot illustrates the correlation between a specific property and its corresponding SHAP values. It can also include the influence of another feature if desired.



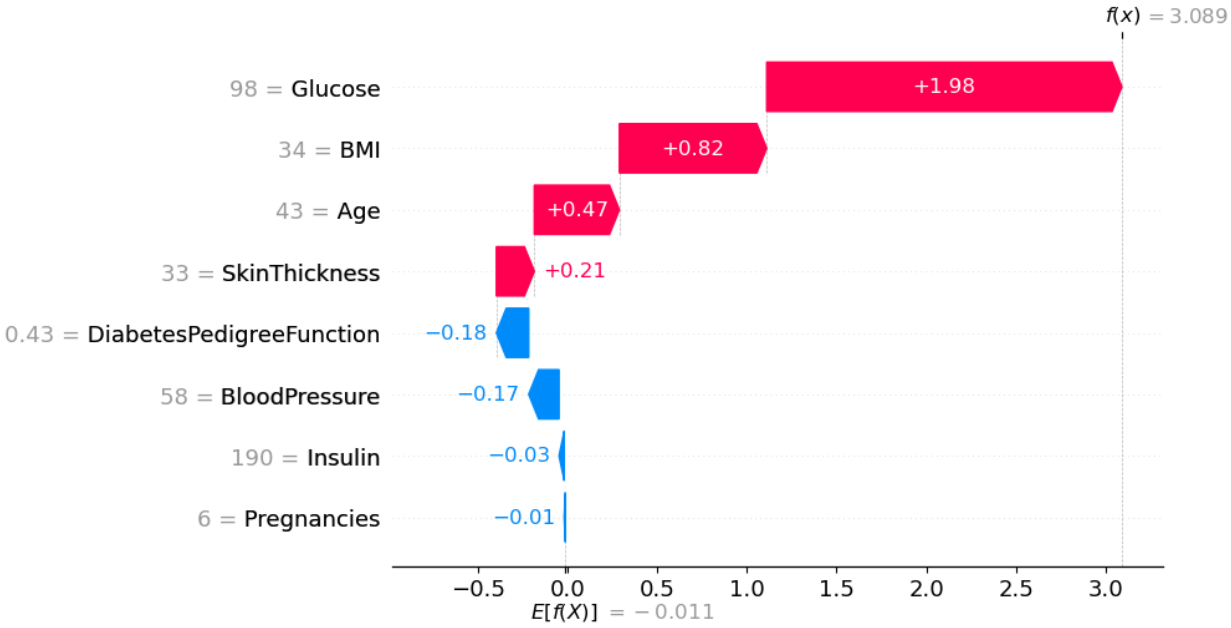
4.2.4 SHAP Decision plot

This plot illustrates the decision-making process of the model by aggregating the contributions of each feature for individual data.



4.2.5 SHAP Waterfall plot

This plot illustrates the decomposition of a singular prediction based on the individual contribution of each feature. And here is the first value of X_test.

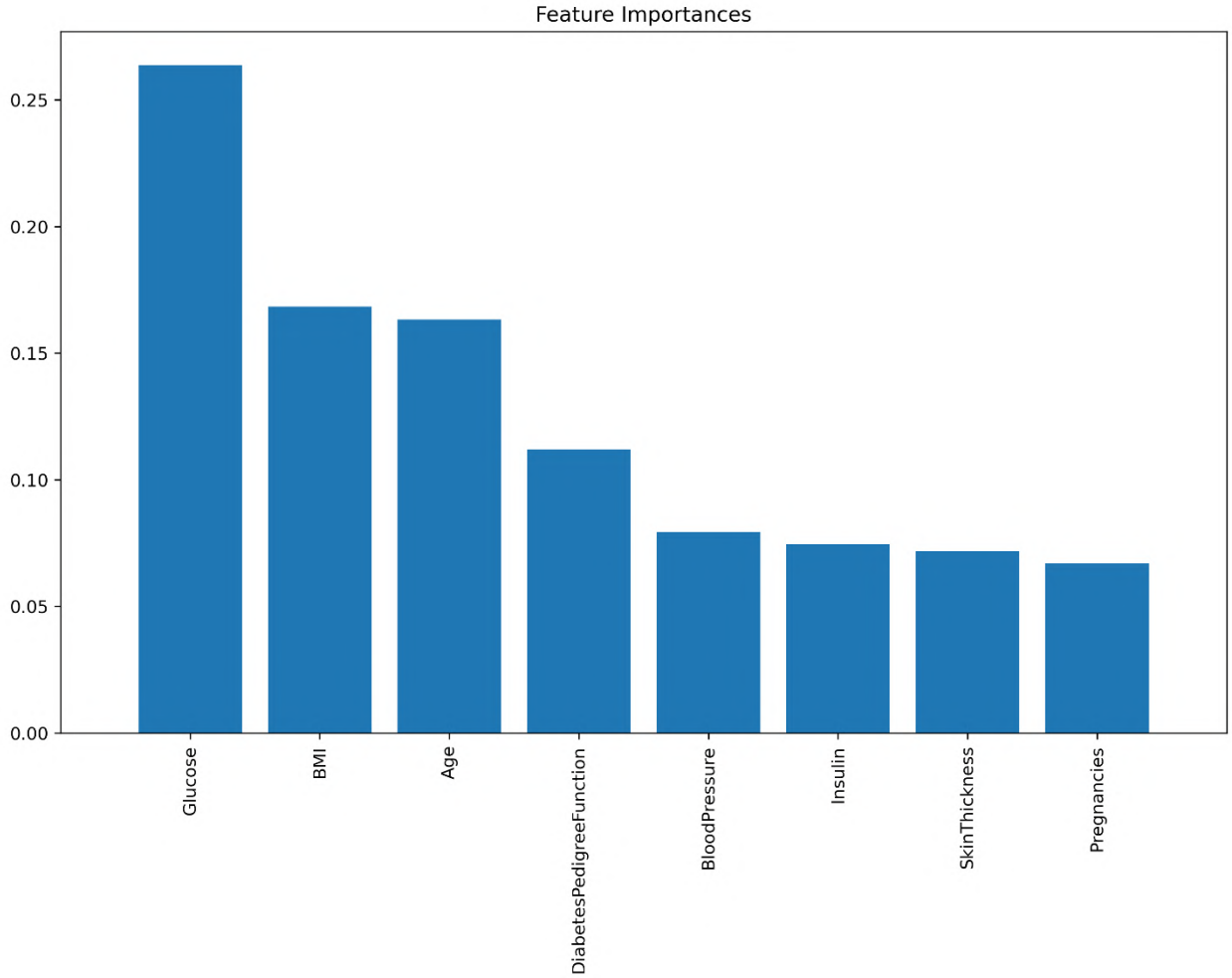


4.3 LIME

LIME's primary objective is to provide detailed explanations for specific predictions.

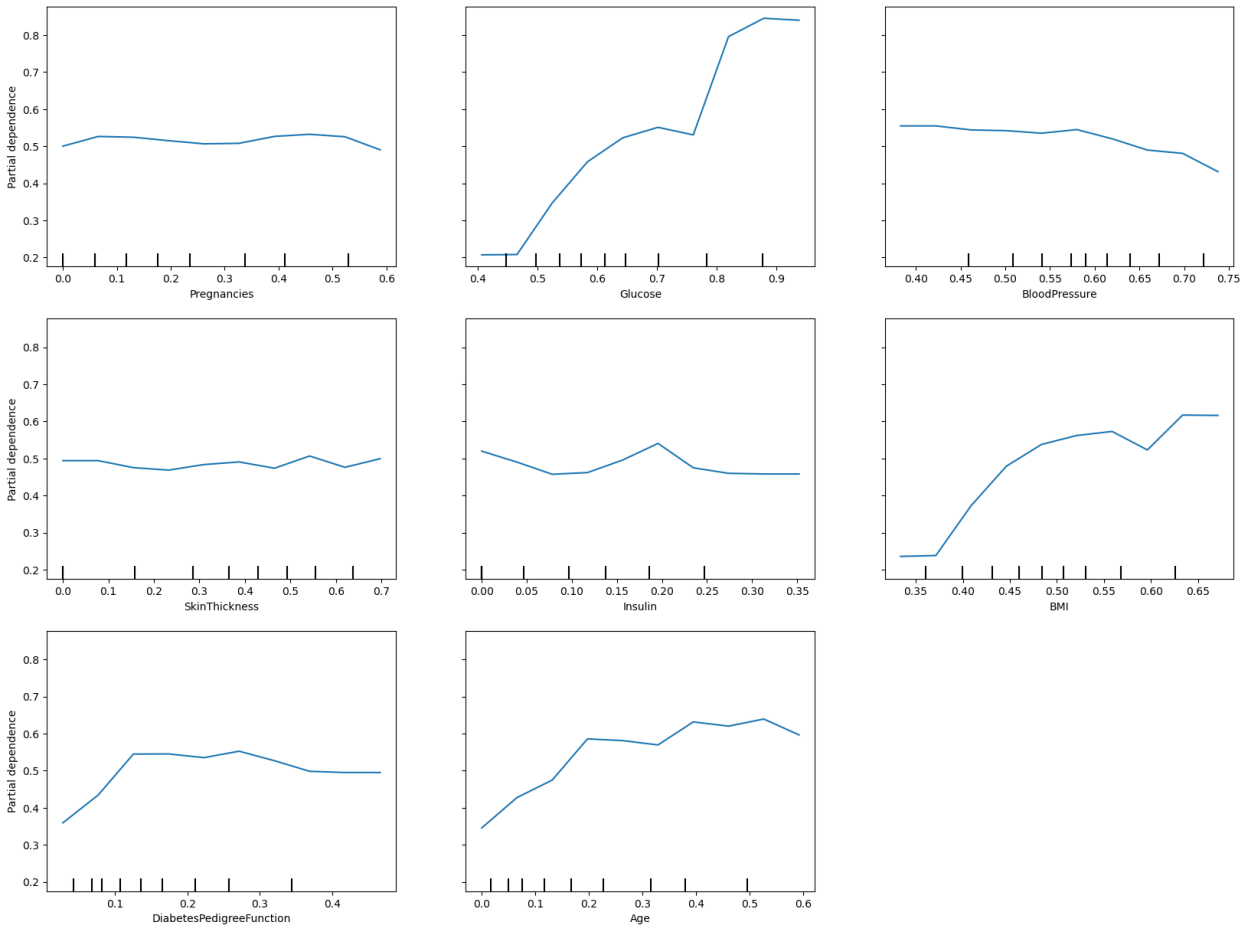
4.4 Feature importance

Feature importance approaches score input characteristics based on their prediction of a target variable. In this example, glucose, BMI, and age predict target variable better.



4.5 Partial dependency plot

This isolates the effect of feature(s) on model predictions, with glucose, BMI, and age exhibiting strongly positive contributions.



5 Comparison with Generative AI Methods

Although Generative AI can be effective for generating and enhancing data, Explainable ML (SHAP, LIME, Feature Importance, and Partial Dependency Plots) provides superior performance in terms of cost, interpretability, and application for predicting diabetes. Within the healthcare sector, where the need for transparent and justified conclusions is paramount, it is more suitable to Explainable ML methods that incorporate explainability methodologies.

- Performance - Explainable ML are often more provide better performance at no cost, Generative AI methods may provide superior performance but at a greater expense.
- Interpretability - Generative AI techniques, such as GANs or VAEs, exhibit lower interpretability compared to Explainable ML approaches.

- Applicability - The explainable ML approach is more suitable for situations that demand transparency and justified conclusions.

6 Ethical Considerations

- Ensuring the model does not exhibit unjust discrimination against particular demographics. It is necessary to examine the dataset for any biases and assess the model's fairness.
- Data privacy refers to the protection and control of personal information, ensuring that it is kept confidential and used only for authorised purposes. Strict privacy measures must be implemented to safeguard sensitive health information of patients, in order to comply with requirements such as HIPAA.
- Data security is of utmost importance. It is necessary to ensure that the information is anonymised and that robust measures are implemented to handle and preserve patient data securely.

7 Conclusion

Explainable machine learning algorithms provide insights into model predictions, making them suitable for essential applications like healthcare diagnosis. We can construct trustworthy models to help healthcare practitioners make informed decisions by optimising model performance and using reliable explainability methodologies.

This effort used XGBoost classifier and Bayesian hyperparameter tuning to improve an interpretable machine learning model for diabetes prediction. Explainable ML approaches reveal that Glucose, BIM, and Age strongly influence model prediction and can interpret each datapoint, which is significant for healthcare. This study shows that explainable machine learning methods are essential for building reliable and trustworthy models.

References

Smith, J.W., Everhart, J.E., Dickson, W.C., Knowler, W.C., & Johannes, R.S. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. *In Proceedings of the Symposium on Computer Applications and Medical Care* (pp. 261--265). IEEE Computer Society Press.