Project: Data Mining on diabetes dataset

Table of Contents

[I. Introduction 4](#__RefHeading___Toc654_897734607)

[Questions to be answered by the following implementations and analysis 4](#__RefHeading___Toc656_897734607)

[The Data 4](#__RefHeading___Toc658_897734607)

[Relevance and Importance: 5](#__RefHeading___Toc660_897734607)

[Rich Feature Set: 5](#__RefHeading___Toc662_897734607)

[II. Statistical Measures 5](#__RefHeading___Toc664_897734607)

[Descriptive statistics 5](#__RefHeading___Toc666_897734607)

[Data Distribution 6](#__RefHeading___Toc668_897734607)

[Interpretation of the values: 6](#__RefHeading___Toc670_897734607)

[III. Regression 7](#__RefHeading___Toc672_897734607)

[Logistic Regression Model 8](#__RefHeading___Toc674_897734607)

[Regression Coefficients 9](#__RefHeading___Toc676_897734607)

[Model Evaluation 9](#__RefHeading___Toc678_897734607)

[Training Confusion Matrix Results: 9](#__RefHeading___Toc680_897734607)

[Testing Confusion Matrix Results: 10](#__RefHeading___Toc682_897734607)

[Precision and Recall 10](#__RefHeading___Toc684_897734607)

[Classification Report 11](#__RefHeading___Toc686_897734607)

[Linear Regression Performance Analysis 11](#__RefHeading___Toc688_897734607)

[1. Model Accuracy: 11](#__RefHeading___Toc690_897734607)

[2. Confusion Matrix: 11](#__RefHeading___Toc692_897734607)

[3. Precision and Recall: 11](#__RefHeading___Toc694_897734607)

[4. Classification Report: 11](#__RefHeading___Toc696_897734607)

[Feature Importance: 11](#__RefHeading___Toc698_897734607)

[IV. Decision Trees 12](#__RefHeading___Toc700_897734607)

[V. Model Training 12](#__RefHeading___Toc702_897734607)

[Model Evaluation 12](#__RefHeading___Toc704_897734607)

[Hyperparameter Tuning 12](#__RefHeading___Toc706_897734607)

[Feature Importance 13](#__RefHeading___Toc708_897734607)

[Permutation Feature Importance 13](#__RefHeading___Toc710_897734607)

[VI. Data Analysis and Explanation of Results 14](#__RefHeading___Toc712_897734607)

[Basic Decision Tree Model 14](#__RefHeading___Toc714_897734607)

[Confusion Matrix Analysis 14](#__RefHeading___Toc716_897734607)

[Cross-Validation 14](#__RefHeading___Toc718_897734607)

[Optimized Model 14](#__RefHeading___Toc720_897734607)

[Hyperparameter Tuning 14](#__RefHeading___Toc722_897734607)

[Optimized Accuracy (0.6926): 14](#__RefHeading___Toc724_897734607)

[Slightly lower but more reliable than the initial model, indicating better generalization. 14](#__RefHeading___Toc726_897734607)

[Optimized Confusion Matrix: 14](#__RefHeading___Toc728_897734607)

[Feature Importance 14](#__RefHeading___Toc730_897734607)

[Most Important Features: 14](#__RefHeading___Toc732_897734607)

[Permutation Feature Importance 15](#__RefHeading___Toc734_897734607)

[VII. Assoctiation Rules 16](#__RefHeading___Toc736_897734607)

[The Apriori Algorithm 16](#__RefHeading___Toc738_897734607)

[Data preprocessing 16](#__RefHeading___Toc740_897734607)

[Binning Continuous Features 16](#__RefHeading___Toc742_897734607)

[Applying the Apriori Algorithm 17](#__RefHeading___Toc744_897734607)

[Results 18](#__RefHeading___Toc746_897734607)

[Top Rules: Sorted by lift, confidence, and support to identify the most significant patterns. 18](#__RefHeading___Toc748_897734607)

[Interpretation of Results 19](#__RefHeading___Toc750_897734607)

[Support vs. Confidence Colored by Lift: 19](#__RefHeading___Toc752_897734607)

[Heatmap 21](#__RefHeading___Toc754_897734607)

[Data Interpretation 21](#__RefHeading___Toc756_897734607)

[Lift: This relationship is much stronger than what would be expected by random chance. 21](#__RefHeading___Toc758_897734607)

[Clustering 22](#__RefHeading___Toc760_897734607)

[Scaling was necessary to ensure that features contribute equally to the distance calculations used in clustering algorithms. 22](#__RefHeading___Toc762_897734607)

[Partition Clustering 22](#__RefHeading___Toc764_897734607)

[K-Means Clustering: 22](#__RefHeading___Toc766_897734607)

[Convergence: Repeat the assignment and update steps until the cluster centers do not change significantly. 22](#__RefHeading___Toc768_897734607)

[K-Medoids Clustering 22](#__RefHeading___Toc770_897734607)

[This indicates an even lower agreement between the true labels and the clustering results compared to K-means. 22](#__RefHeading___Toc772_897734607)

[Hierarchical Clustering 23](#__RefHeading___Toc774_897734607)

[Interpretation of Results 24](#__RefHeading___Toc776_897734607)

[K-Means and K-Medoids 24](#__RefHeading___Toc778_897734607)

[Clusters identified do not correspond well with the natural groupings in the data, making it hard to derive meaningful insights. 24](#__RefHeading___Toc780_897734607)

[Hierarchical Clustering 24](#__RefHeading___Toc782_897734607)

[DBSCAN Clustering 24](#__RefHeading___Toc784_897734607)

[The choice of eps and min\_samples heavily influences the outcome, and in most cases, DBSCAN failed to identify meaningful clusters. 24](#__RefHeading___Toc786_897734607)

[Results interpretation: 25](#__RefHeading___Toc788_897734607)

[The clustering results suggest that the data does not naturally group into well-defined clusters using these methods. 25](#__RefHeading___Toc790_897734607)

[For meaningful insights, more sophisticated techniques or domain-specific feature engineering might be necessary. 25](#__RefHeading___Toc792_897734607)

[VIII. Bottom Line Results 26](#__RefHeading___Toc794_897734607)

[Feature Analysis and Influence on Outcome 26](#__RefHeading___Toc796_897734607)

[Implemented Models and Algorithms 26](#__RefHeading___Toc798_897734607)

[Logistic Regression 26](#__RefHeading___Toc800_897734607)

[Decision Trees 27](#__RefHeading___Toc802_897734607)

[Association Rules 27](#__RefHeading___Toc804_897734607)

[Clustering 27](#__RefHeading___Toc806_897734607)

[Summary of Findings 28](#__RefHeading___Toc808_897734607)

[Conclusion 28](#__RefHeading___Toc810_897734607)

[IX. Resources: 29](#__RefHeading___Toc812_897734607)

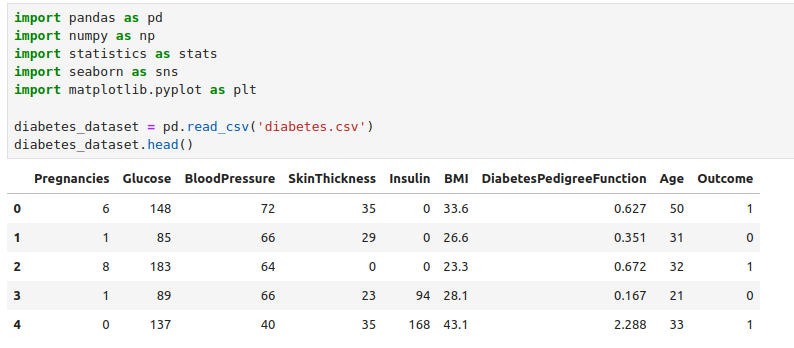
# Introduction

## Questions to be answered by the following implementations and analysis

1. Can we develop a reliable predictive model to identify individuals at risk of diabetes?
2. Can we develop a reliable predictive model to identify individuals at risk of diabetes?
3. How do different data mining techniques compare in terms of accuracy and interpretability
4. What are the common patterns and associations among different features in the dataset?

## The Data

The dataset I used in this analysis is the Pima Indians Diabetes Database, which includes 768 observations and 9 features: Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age, and Outcome (indicating the presence of diabetes).



**Pregnancies**: This feature represents the number of times the individual has been pregnant. It's a discrete numerical feature. **Scale of Measurement**: Ratio.

**Glucose**: This feature represents the plasma glucose concentration after a 2-hour oral glucose tolerance test. It's a continuous numerical feature. **Scale of Measurement**: Interval.

**Blood Pressure**: This feature represents the diastolic blood pressure (mm Hg) of the individual. It's a continuous numerical feature. **Scale of Measurement:** Ratio

**Skin Thickness**: This feature represents the thickness of the skin (triceps skin fold thickness) measured in mm. It's a continuous numerical feature. Scale of Measurement: Ratio.

**Insulin**: This feature represents the 2-hour serum insulin (mu U/ml) of the individual. It's a continuous numerical feature. Scale of Measurement: Ratio.

**BMI**: This feature represents the Body Mass Index (weight in kg / (height in m)^2) of the individual. It's a continuous numerical feature. Scale of Measurement: Ratio.

**Diabetes Pedigree Function**: This feature represents a function that scores the likelihood of diabetes based on family history. It's a continuous numerical feature. Scale of Measurement: Ratio

**Age**: This feature represents the age of the individual in years. It's a discrete numerical feature. Scale of Measurement: Ratio.

**Outcome**: This feature represents whether the individual has diabetes or not, with "1" indicating the presence of diabetes and "0" indicating the absence. It's a categorical feature, specifically binary. Scale of Measurement: Nominal.

### **Relevance and Importance**:

* ***Health Impact***: Diabetes is a major public health concern affecting millions of people worldwide. Understanding the factors that contribute to diabetes can help in early detection, prevention, and effective management.
* ***Predictive Modeling***: This dataset provides an excellent opportunity to develop predictive models that can help identify individuals at high risk of developing diabetes. Such models can be instrumental in medical research and clinical decision-making.

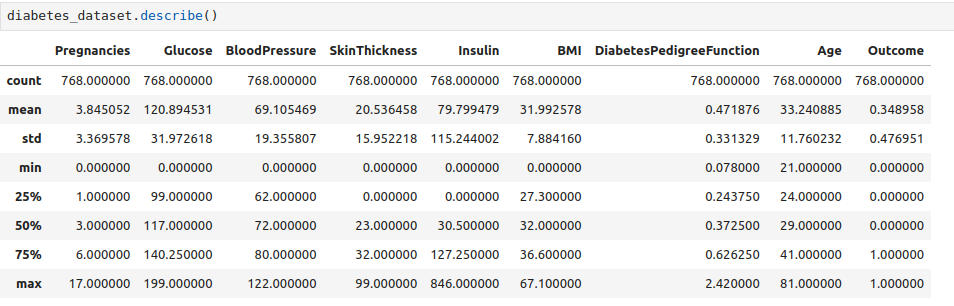
### **Rich Feature Set**:

* ***Diverse Attributes***: The dataset includes a variety of features that are clinically relevant to diabetes, such as glucose levels, blood pressure, BMI, insulin levels, and others. This diversity allows for comprehensive analysis and the exploration of different relationships and interactions among features.
* ***Demographic Data***: Features like age and pregnancies provide additional context that can enhance the understanding of diabetes risk factors.
* ***Publicly Available***: The diabetes dataset is readily available from sources like the UCI Machine Learning Repository, making it accessible for research and educational purposes.
* ***Manageable Size***: With 768 observations and 9 features, the dataset is large enough to provide meaningful insights but small enough to be processed and analyzed without requiring extensive computational resources.

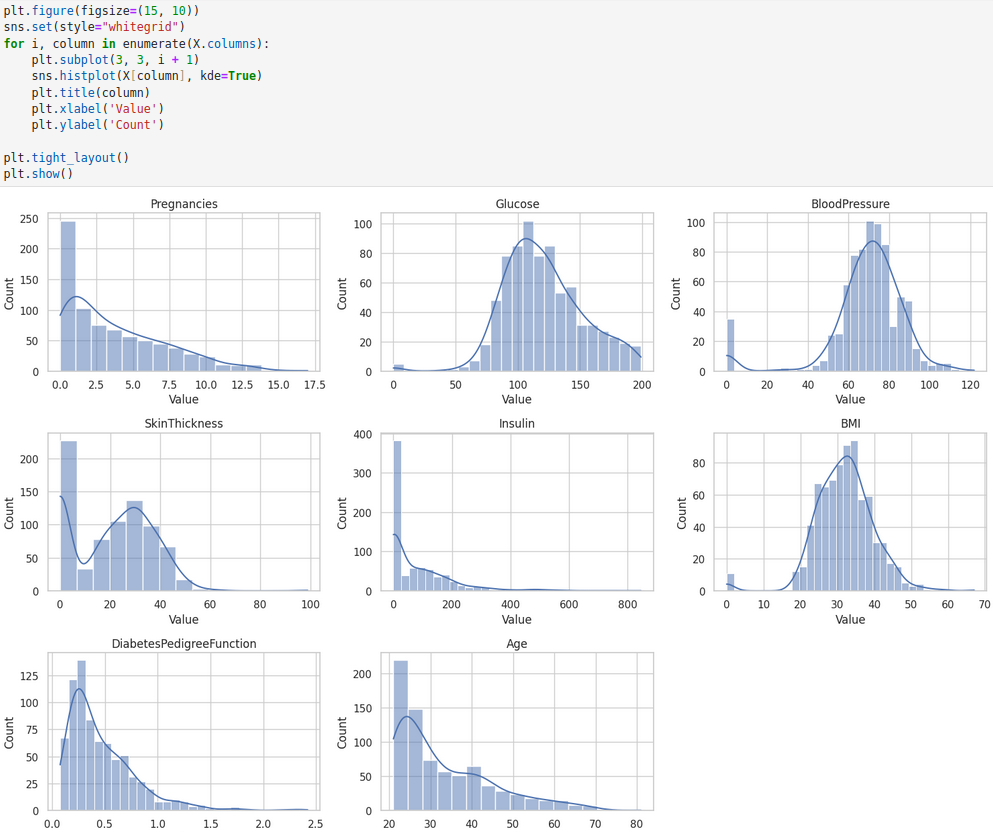
# Statistical Measures

## Descriptive statistics

* count, mean, std, min, 25%, 50%, 75%, max) provide a summary of the distribution of each feature.



## Data Distribution



## Interpretation of the values:

1. Pregnancies:
   * Mean: The average number of pregnancies in the dataset is approximately 3.85.
   * Standard Deviation (Std): The standard deviation is approximately 3.37, indicating a moderate spread of data around the mean.
   * Mode: The most frequent number of pregnancies is 1.
   * Interpretation: Higher numbers of pregnancies might indicate a higher risk of developing diabetes, as pregnancy can affect insulin sensitivity and glucose metabolism. However, it's essential to consider other factors such as age and lifestyle.
2. Glucose:
   * Mean: The average glucose level is approximately 120.89 mg/dL.
   * Std: The standard deviation is approximately 31.97, indicating variability in glucose levels.
   * Mode: The most frequent glucose level is 100 mg/dL.
   * Interpretation: High glucose levels (hyperglycemia) are a key indicator of diabetes. Higher glucose levels are associated with an increased risk of diabetes, as insulin resistance or insufficient insulin production can lead to elevated blood sugar levels.
3. Blood Pressure:
   * Mean: The average blood pressure is approximately 69.11 mm Hg (diastolic).
   * Std: The standard deviation is approximately 19.36, indicating variability in blood pressure measurements.
   * Mode: The most frequent blood pressure measurement is 70 mm Hg.
   * Interpretation: Elevated blood pressure (hypertension) is a common comorbidity of diabetes. High blood pressure can contribute to the progression of diabetes complications, such as cardiovascular disease and kidney disease.
4. Skin Thickness:
   * Mean: The average skin thickness is approximately 20.54 mm.
   * Std: The standard deviation is approximately 15.95, indicating variability in skin thickness measurements.
   * Mode: The most frequent skin thickness is 0 mm.
   * Interpretation: Skin thickness may not directly influence diabetes outcome but could be related to obesity or insulin resistance. However, it's essential to interpret this feature in conjunction with other clinical parameters.
5. Insulin:
   * Mean: The average insulin level is approximately 79.80 μU/ml.
   * Std: The standard deviation is approximately 115.24, indicating considerable variability in insulin levels.
   * Mode: The most frequent insulin level is 0 μU/ml.
   * Interpretation: Insulin levels are directly related to diabetes, as insulin resistance or insufficient insulin production characterizes diabetes. Higher insulin levels may indicate insulin resistance, a precursor to type 2 diabetes.
6. BMI (Body Mass Index):
   * Mean: The average BMI is approximately 31.99 kg/m^2.
   * Std: The standard deviation is approximately 7.88, indicating variability in BMI measurements.
   * Mode: The most frequent BMI is 32.0 kg/m^2.
   * Interpretation: High BMI is strongly associated with an increased risk of type 2 diabetes. Obesity is a major risk factor for diabetes, as it contributes to insulin resistance and metabolic dysfunction.
7. Diabetes Pedigree Function:
   * Mean: The average diabetes pedigree function is approximately 0.47.
   * Std: The standard deviation is approximately 0.33, indicating variability in diabetes pedigree function values.
   * Mode: The most frequent diabetes pedigree function value is 0.254.
   * Interpretation: The diabetes pedigree function assesses the genetic predisposition to diabetes. Higher values indicate a stronger family history of diabetes, which can increase the risk of developing the condition.
8. Age:
   * Mean: The average age is approximately 33.24 years.
   * Std: The standard deviation is approximately 11.76, indicating variability in ages.
   * Mode: The most frequent age number is 22.
   * Interpretation: Age is a significant risk factor for diabetes, as the prevalence of diabetes increases with age. Older individuals are more likely to develop diabetes due to factors such as reduced insulin sensitivity and lifestyle changes.

# Regression

When selecting a machine learning algorithm for a particular dataset, it's crucial to consider the nature of the data, the problem at hand, and the strengths of various algorithms.

1. Nature of the Problem

The primary task with the diabetes dataset is binary classification: determining whether an individual has diabetes (1) or does not have diabetes (0). Logistic regression is specifically designed for binary classification tasks. It directly estimates the probability of the default class, making it inherently suitable for this kind of problem.

2. Interpretability

Logistic regression is one of the most interpretable machine learning models. The model provides coefficients for each feature, which can be directly interpreted in terms of odds ratios. For instance, you can understand how a unit change in glucose level affects the odds of having diabetes. This interpretability is crucial in medical applications where understanding the influence of different factors is important for both practitioners and patients.

3. Performance

Logistic regression performs well when the relationship between the features and the outcome is approximately linear. Given the nature of the diabetes dataset, which includes features such as glucose level, BMI, and age, there is reason to believe that the relationship between these features and the likelihood of diabetes could be approximated linearly.

In the given results:

The training accuracy was 77.65%, and the test accuracy was 78.79%.

These are respectable performance metrics, indicating that the logistic regression model has learned the underlying patterns in the training data and generalizes reasonably well to the test data.

## Logistic Regression Model

Logistic Regression is a statistical model used for binary classification tasks. It estimates the probability that a given input belongs to a certain class. It’s commonly used in scenarios where the outcome is binary, such as predicting whether a person has a disease (yes/no).

First, I split the features (X) and targets (Y) of the dataset into training and testing data, choosing a ratio of 70% for training and 30% for testing.

I trained the Logistic Regression Model by importing the model from scikit-learn and used the training sets, both for the features (X) and target (Y) to fit the model.

I used ***regularization parameters*** and grid search cross-validation that improved the model’s performance.

***Regularization*** is a technique used to avoid overfitting in machine learning models. It does this by adding a penalty term to the objective function (also called the loss function or error function) that the model is trying to minimize.

***L1 (Lasso)*** regularization adds the absolute value of the coefficients to the cost function.

***L2 (Ridge)*** regularization adds the square of the coefficients to the cost function.

* adds a penalty term to the objective function equal to the square of the coefficients.
* it leads to a model with all coefficients close to zero, but not necessarily equal to zero.
* is less prone to overfitting than L1 regularization and is often used as a default choice.

***The “C” hyperparameter*** controls the strength of the regularization. A smaller value for “C” (e.g. C=0.01) leads to stronger regularization and a simpler model, while a larger value (e.g. C=1.0) leads to weaker regularization and a more complex model.

The best model was found with C=1, penalty='l2', and solver='liblinear'. This approach helped in finding the optimal balance between bias and variance, leading to a slightly better cross-validation score and consistent accuracy on both training and test datasets.

### Regression Coefficients

1. Intercept: The intercept is a constant term in the regression equation. It represents the log-odds of the outcome when all predictors are zero.
2. Coefficients: These are weights assigned to each feature in the model, indicating how much each feature contributes to the outcome.

For the training data upon fitting the model, I received the following intercept and regression coefficients:

* Intercept: - 7.76086305
* Coefficients: 0.11492368, 0.03372375, -0.01737678, 0.0047834, -0.00149846, 0.08449105, 1.00366767, 0.0131593

### Model Evaluation

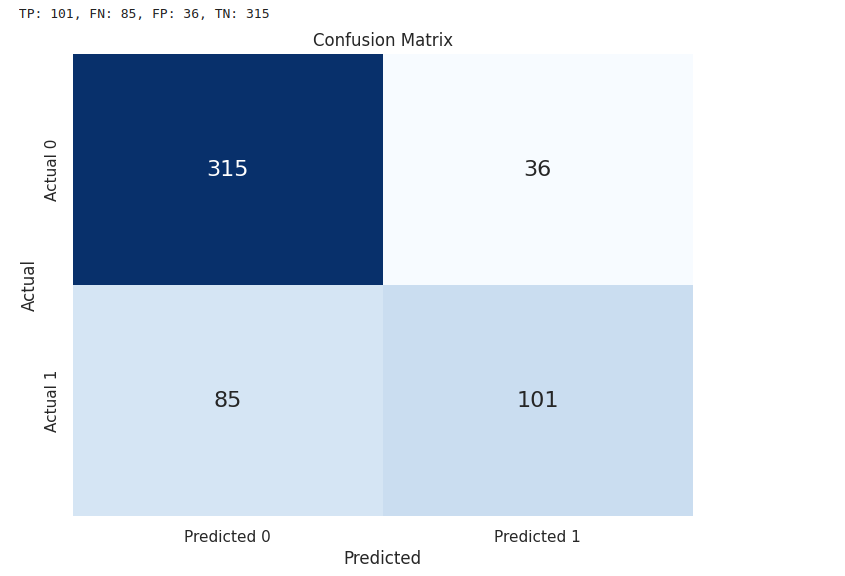
Using another scikit-learn module, I found out the accuracy score on the training data.

* Accuracy score of training data: 0.7746741154562383 (77%)

I also used a confusion matrix algorithm for evaluating the model’s performance. Confusion matrix, an essential tool in evaluating the performance of a classification model, encompasses elements representing True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. Furthermore, validation techniques ensure the model’s robustness, guard against overfitting, and enhance its generalization capabilities.

#### Training Confusion Matrix Results:

* TP: 101, FN: 85, FP: 36, TN: 315

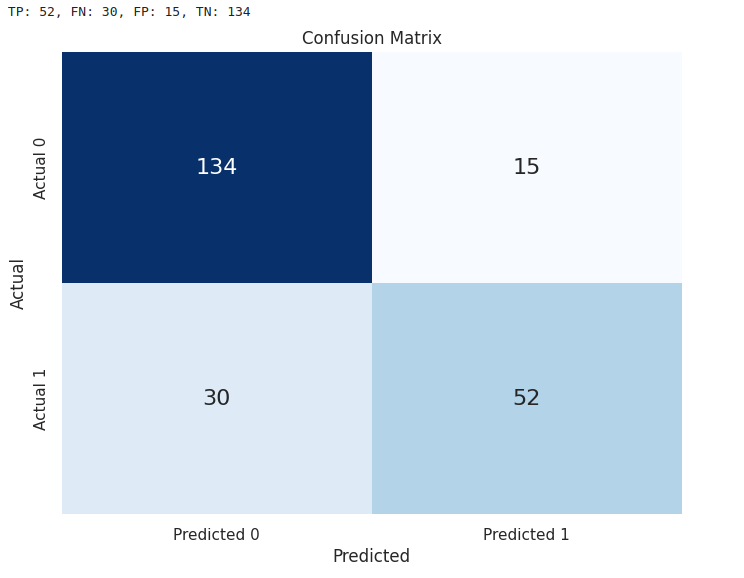


* Accuracy score of test data: 0.8051948051948052 (80%)

I used the confusion matrix algorithm for the model’s performance on the test data.

#### Testing Confusion Matrix Results:

* TP: 52, FN: 30, FP: 15, TN: 134

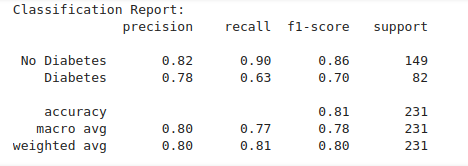


### Precision and Recall

* Precision: The ratio of true positives to the sum of true positives and false positives.
  + Formula: TP / ( TP + FP )
  + For my trained model: 0.78
* Recall: The ratio of true positives to the sum of true positives and false negatives.
  + Formula: TP / ( TP + FN )
  + For my trained model: 0.63

### Classification Report

The classification report provides a comprehensive view of the precision, recall, and F1-score for each class, as well as the overall accuracy, macro average, and weighted average metrics.



## Linear Regression Performance Analysis

#### 1. Model Accuracy:

The accuracy of around 77-80% on both the training and test sets indicates that the model is reasonably good at predicting whether an individual has diabetes. However, accuracy alone is not always the best metric, especially in imbalanced datasets.

#### 2. Confusion Matrix:

The confusion matrix provides a more detailed insight into the performance:

* Training Data: There are 101 true positives and 85 false negatives, indicating some cases of diabetes are missed. The number of false positives is 36, which is relatively low.
* Testing Data: There are 52 true positives and 30 false negatives, indicating the model misses more cases of diabetes in the test set. The number of false positives is 15, which is relatively lower.

#### 3. Precision and Recall:

* Precision: 78% suggests that when the model predicts diabetes, it is correct 70% of the time.
* Recall: 63% indicates that the model is able to identify 67% of the actual diabetes cases, which is somewhat lower and suggests that the model might be improved to better capture more of the actual positive cases.

#### 4. Classification Report:

* No Diabetes: The model performs better for the "No Diabetes" class with higher precision (0.82) and recall (0.90).
* Diabetes: For the "Diabetes" class, the model has a precision of 0.78 and recall of 0.63, indicating that it struggles more to correctly identify diabetic cases.

### Feature Importance:

The largest positive coefficient, 1.00366767 for the Diabetes Pedigree Function, suggests that this feature has a strong positive impact on the likelihood of diabetes. In contrast, the negative coefficients suggest a negative relationship with diabetes.

# Decision Trees

A non-linear model that splits the data into subsets based on the most significant features. Each split is determined by criteria that maximize the homogeneity of the resulting subsets.

## Model Training

The Decision Tree model is trained using the training data.

Training Accuracy: 1.0 (100% accuracy on the training set, indicating overfitting).Test Accuracy: 0.7056 (lower accuracy on the test set, indicating overfitting).

#### Model Evaluation

1. Confusion Matrix: Shows the performance of the model by comparing the predicted and actual labels.
   * True Positives (TP): Correctly predicted positives (192).
   * False Negatives (FN): Actual positives incorrectly predicted as negatives (0).
   * False Positives (FP): Actual negatives incorrectly predicted as positives (0).
   * True Negatives (TN): Correctly predicted negatives (345).
2. Cross-Validation Scores: Evaluates the model's performance across different subsets of the data.
3. Mean CV Accuracy: 0.7004 (indicating the model's general performance).

### Hyperparameter Tuning

Grid Search: Optimizes the model by searching for the best combination of hyperparameters.

* Best Parameters: {'criterion': 'gini', 'max\_depth': 10, 'min\_samples\_leaf': 2, 'min\_samples\_split': 2}

criterion: Measure to split nodes (Gini impurity or entropy).

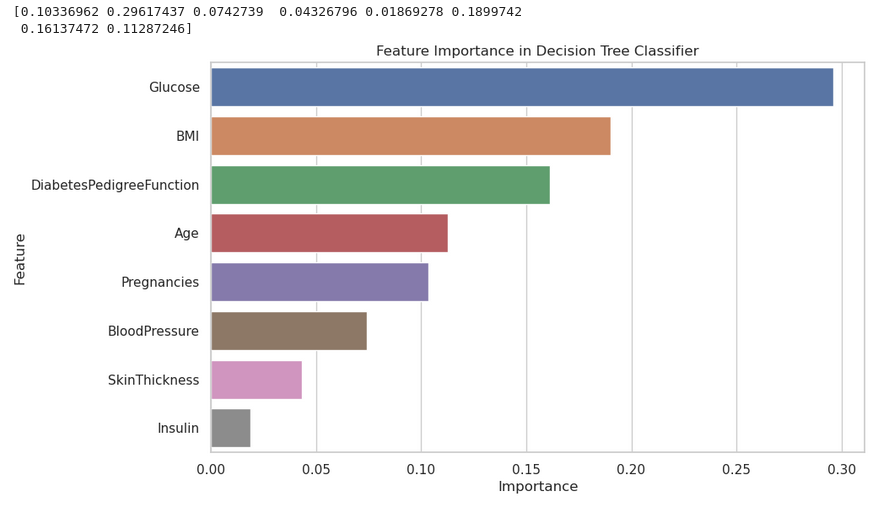
max\_depth: Maximum depth of the tree to prevent overfitting.

min\_samples\_split: Minimum number of samples required to split a node.

min\_samples\_leaf: Minimum number of samples required at a leaf node.

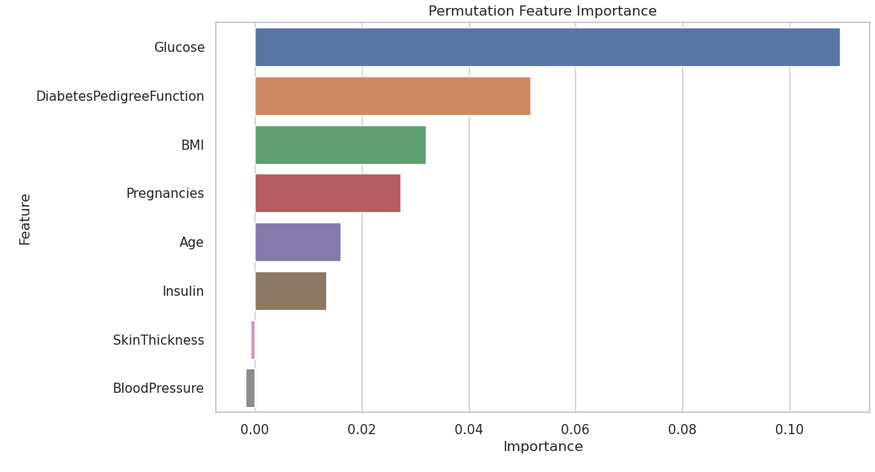
* Best Cross-Validation Score: 0.7262

### Feature Importance

* Importance Scores: [0.103, 0.296, 0.074, 0.043, 0.018, 0.189, 0.161, 0.112]
* Visualization: Bar plots of feature importance.

### Permutation Feature Importance

Measures the change in the model's performance when the values of a feature are randomly shuffled.



## Data Analysis and Explanation of Results

## Basic Decision Tree Model

* Training Accuracy (1.0): The model perfectly fits the training data, indicating it may have memorized the data rather than learning general patterns.
* Test Accuracy (0.7056): The model performs significantly worse on unseen data, confirming overfitting.

### Confusion Matrix Analysis

* True Positives (192): The model correctly identified 192 cases of diabetes.
* False Negatives (0): The model did not miss any diabetes cases in the training set.
* False Positives (0): The model did not incorrectly classify any non-diabetes cases as diabetes.
* True Negatives (345): The model correctly identified 345 non-diabetes cases.

### Cross-Validation

* Mean CV Accuracy (0.7004): Indicates the model's performance across multiple folds, providing a more robust measure of its generalization capability.

## Optimized Model

### Hyperparameter Tuning

* Best Parameters: Found using Grid Search, these parameters help balance the model's complexity and performance:
  + criterion='gini': Uses Gini impurity to measure the quality of a split.
  + max\_depth=10: Limits the depth of the tree to prevent overfitting.
  + min\_samples\_leaf=2: Ensures each leaf node has at least 2 samples.
  + min\_samples\_split=2: Ensures a node must have at least 2 samples to be split.
* Best Cross-Validation Score (0.7262): Indicates improved performance with the optimized model.

### **Optimized Accuracy (0.6926)**:

#### Slightly lower but more reliable than the initial model, indicating better generalization.

### **Optimized Confusion Matrix**:

* True Positives (118): Correctly predicted diabetes cases.
* False Negatives (37): Missed diabetes cases.
* False Positives (34): Incorrectly predicted diabetes cases.
* True Negatives (42): Correctly identified non-diabetes cases.

### Feature Importance

#### Most Important Features:

* Glucose (0.296): Most significant feature, indicating higher glucose levels strongly predict diabetes.
* BMI (0.189): High BMI also contributes significantly to diabetes prediction.
* DiabetesPedigreeFunction (0.161): Genetic predisposition to diabetes.
* Age (0.112): Older age increases the likelihood of diabetes.

### Permutation Feature Importance

* **Glucose**: Consistently the most influential feature.
* **Diabetes Pedigree Function, BMI, Pregnancies**: Significant but less than glucose.
* **Blood Pressure, Skin Thickness, Insulin**: Least influential features.

# Assoctiation Rules

Association Rule Mining is a method to discover interesting relationships (associations) among a large set of data items. It is commonly used in market basket analysis to find product purchase patterns.

Support: The proportion of transactions in the dataset that contain a specific itemset. It indicates how frequently the itemset appears in the dataset.

Confidence: The likelihood that the consequent (Y) is found in transactions under the condition that the antecedent (X) is present. It is a measure of the rule's reliability.

Lift: Measures how much more likely the consequent (Y) is to be found in transactions that contain the antecedent (X) compared to a random transaction. A lift value greater than 1 indicates a positive correlation between X and Y.

## The Apriori Algorithm

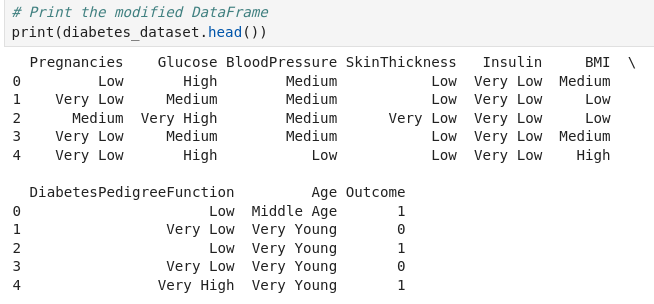
* used to identify frequent itemsets and derive association rules. It uses a breadth-first search strategy to count the support of itemsets and uses a candidate generation function to extend them.

### Data preprocessing

Continuous features are binned into categorical values to apply association rule mining. Binning converts continuous data into intervals, making it suitable for finding patterns in categorical data. Continuous features like Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, and Age were binned into categories (e.g., Very Low, Low, Medium, High, Very High) to convert them into categorical data.

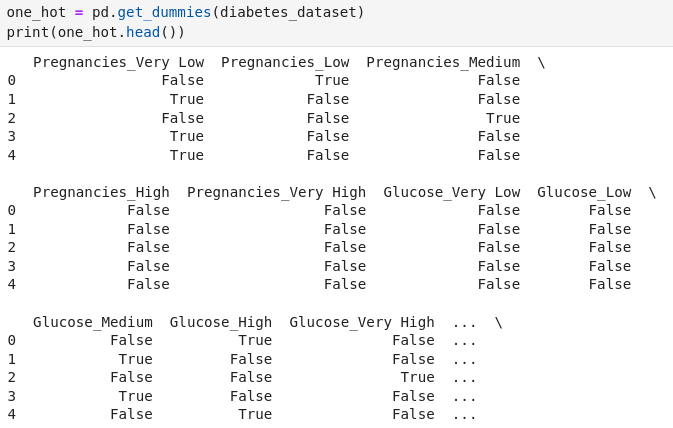
#### Binning Continuous Features

* Purpose: Convert continuous data into categorical intervals to apply association rule mining.
* Reason: Association rule mining algorithms like Apriori are designed for categorical data. Binning helps in transforming continuous features into a format suitable for these algorithms.



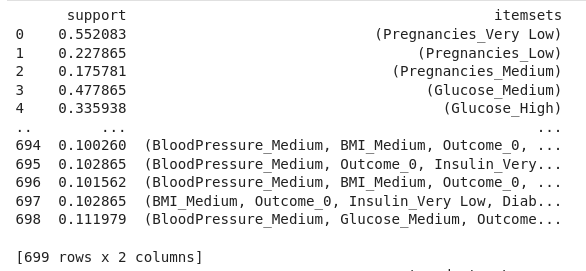
One-hot encoding: Converts categorical data into a format suitable for the Apriori algorithm by creating binary columns for each category.

* Reason: The Apriori algorithm works with binary (presence/absence) data, so one-hot encoding is essential for converting categorical data into this format.



### Applying the Apriori Algorithm

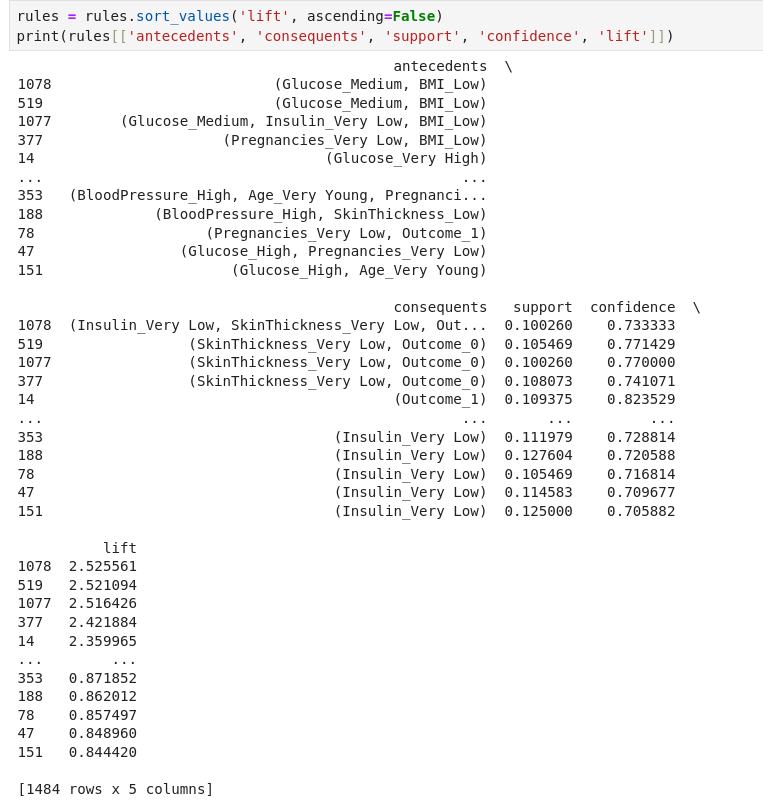
Frequent Itemsets: Itemsets that appear frequently in the dataset above a minimum support threshold.



Association Rules: Rules derived from frequent itemsets with confidence and lift measures to identify interesting relationships.

#### Results

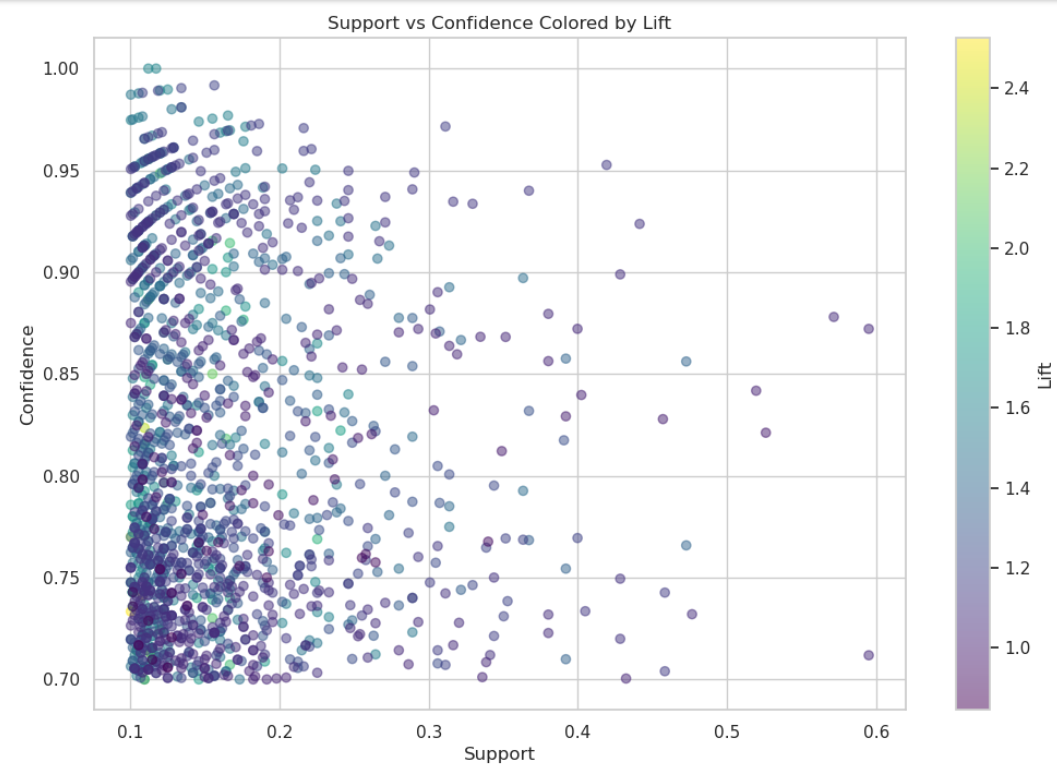
* Top Rules: Sorted by lift, confidence, and support to identify the most significant patterns.



## Interpretation of Results

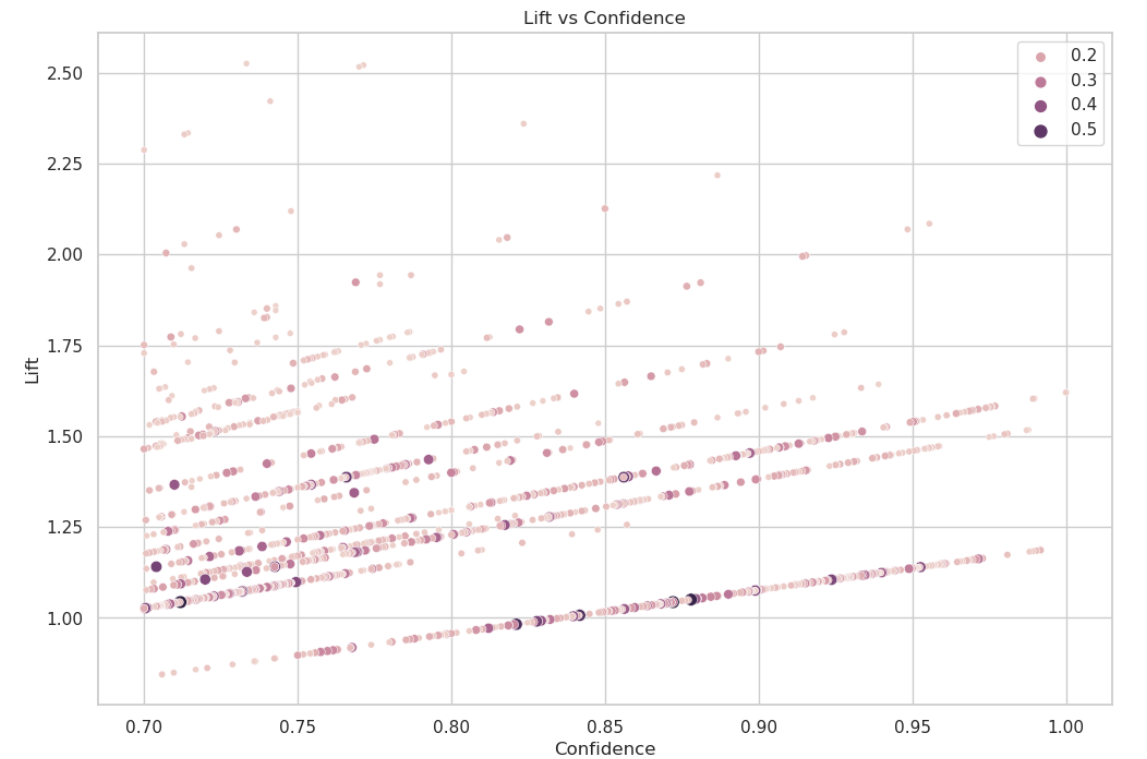
## **Support vs. Confidence Colored by Lift**:

* Shows the relationship between support and confidence for each rule.
* Higher lift values indicate stronger associations.
* Example: A rule with high support, confidence, and lift indicates a strong and frequent relationship in the data.



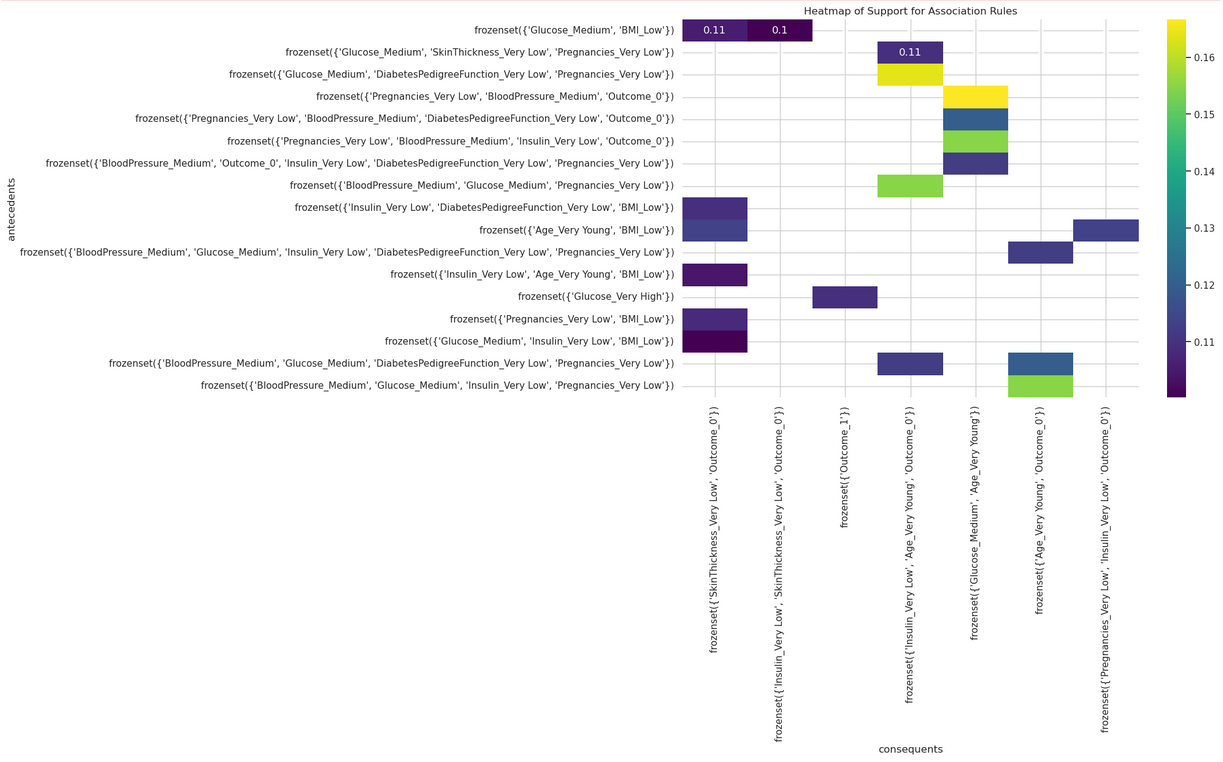
Lift vs. Confidence:

* Shows how lift varies with confidence.
* Helps identify rules that are not only reliable (high confidence) but also significant (high lift).



#### Heatmap

* Heatmap of Support for Top 20 Association Rules:
  + Visual representation of support for the most significant rules based on lift.
  + Allows quick identification of the strongest associations.



## Data Interpretation

For instance, a rule like {Glucose=Very High} -> {Outcome=1} with high support, confidence, and lift means:

* Support: A significant portion of the dataset has very high glucose levels and diabetes.
* Confidence: Patients with very high glucose levels are highly likely to have diabetes.
* Lift: This relationship is much stronger than what would be expected by random chance.

# Clustering

**StandardScaler**:

Scaling was necessary to ensure that features contribute equally to the distance calculations used in clustering algorithms.

## Partition Clustering

### **K-Means Clustering**:

This algorithm partitions data into ***k*** clusters. Each data point belongs to the cluster with the nearest mean, which serves as the cluster center.

**Initialization**: Define the number of clusters kk.

**Assignment Step**: Assign each data point to the nearest cluster center.

**Update Step**: Calculate the new mean of each cluster.

**Convergence**: Repeat the assignment and update steps until the cluster centers do not change significantly.

***Silhouette Score***: 0.1795.

Measures how similar a data point is to its own cluster compared to other clusters. A higher score indicates better-defined clusters.

* This indicates that the clusters are not well-separated; points in the same cluster are not significantly more similar to each other than to points in other clusters.

***ARI Score***: 0.1219.

Measures the similarity between the true labels and the clustering results. A higher score indicates better clustering performance.

* This indicates a low similarity between the true labels and the clustering results, suggesting that K-means is not capturing the natural structure of the data effectively.

### K-Medoids Clustering

Similar to K-means but uses medoids (actual data points) instead of means to define clusters.

***Silhouette Score***: 0.1212

* This is lower than the K-means score, suggesting that the clusters are even less well-defined.

***ARI Score***: 0.0804

* This indicates an even lower agreement between the true labels and the clustering results compared to K-means.

### Hierarchical Clustering

**Agglomerative Hierarchical Clustering**: Builds a hierarchy of clusters by iteratively merging the closest pairs of clusters.

**Linkage Methods**: Determines how the distance between clusters is calculated (e.g., single, complete, average).

**Dendrogram**: A tree-like diagram showing the arrangement of clusters produced by hierarchical clustering.

**Evaluation Metrics**: Silhouette Score for different numbers of clusters.

Silhouette Scores for Different Numbers of Clusters:

* 2 Clusters: 0.1568
* 3 Clusters: 0.1706
* 4 Clusters: 0.1695
* 5 Clusters: 0.1778
* 6 Clusters: 0.1345
* 7 Clusters: 0.1481

These scores suggest that 3 or 5 clusters might provide the best separation of data points, although the scores are generally low.

**Density-Based Clustering**

DBSCAN (Density-Based Spatial Clustering of Applications with Noise): Identifies clusters based on the density of data points.

Parameters:

* ***eps***: The maximum distance between two samples for them to be considered as in the same neighborhood.
* ***min\_samples***: The number of samples in a neighborhood for a point to be considered a core point.

Evaluation Metrics: Number of clusters, Silhouette Score, and ARI.

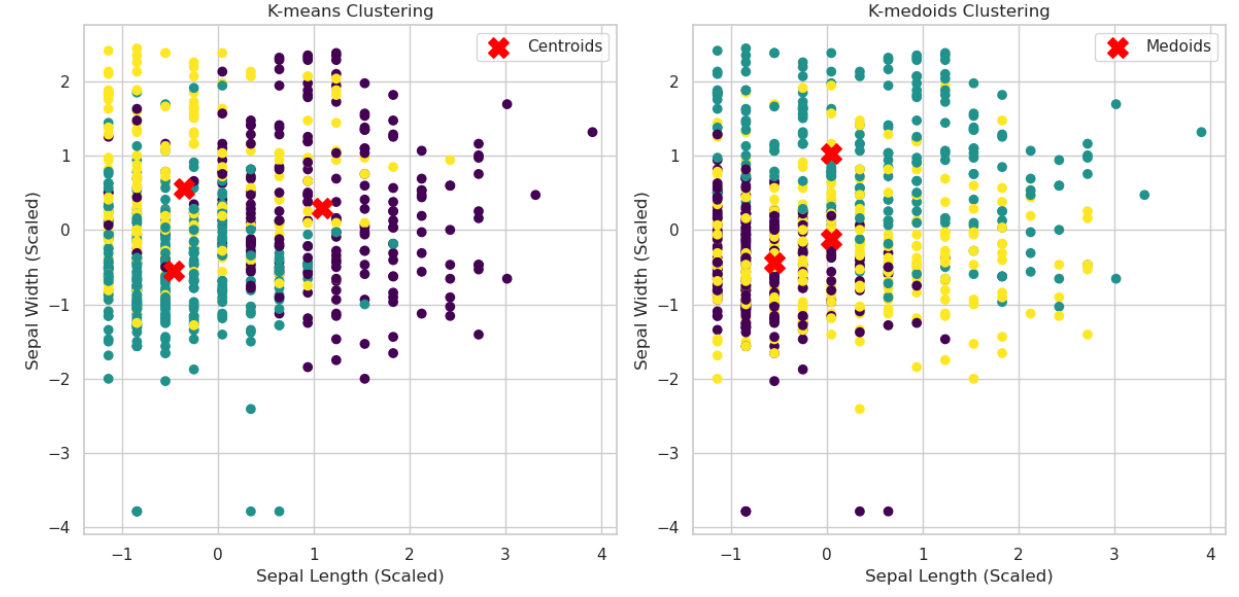
Different Combinations of eps and min\_samples:

* Most combinations resulted in 0 clusters, which is indicative of DBSCAN's sensitivity to parameter selection.
* A few combinations, such as (eps=0.5, min\_samples=2), identified more clusters, but the Silhouette Scores were negative or close to 0, indicating poor cluster quality.

## Interpretation of Results

### K-Means and K-Medoids

* The low Silhouette and ARI scores indicate that these partitioning methods are not well-suited for this dataset.
* Clusters identified do not correspond well with the natural groupings in the data, making it hard to derive meaningful insights.



### Hierarchical Clustering

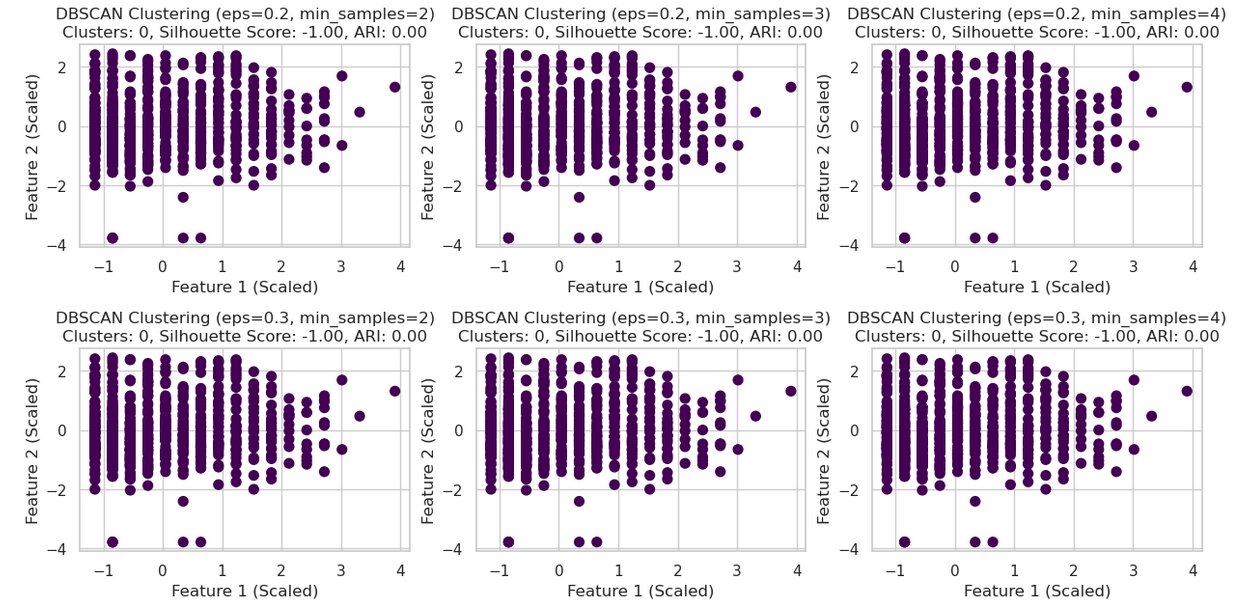
Hierarchical clustering with 3 or 5 clusters provided slightly better silhouette scores, suggesting a somewhat better-defined structure.

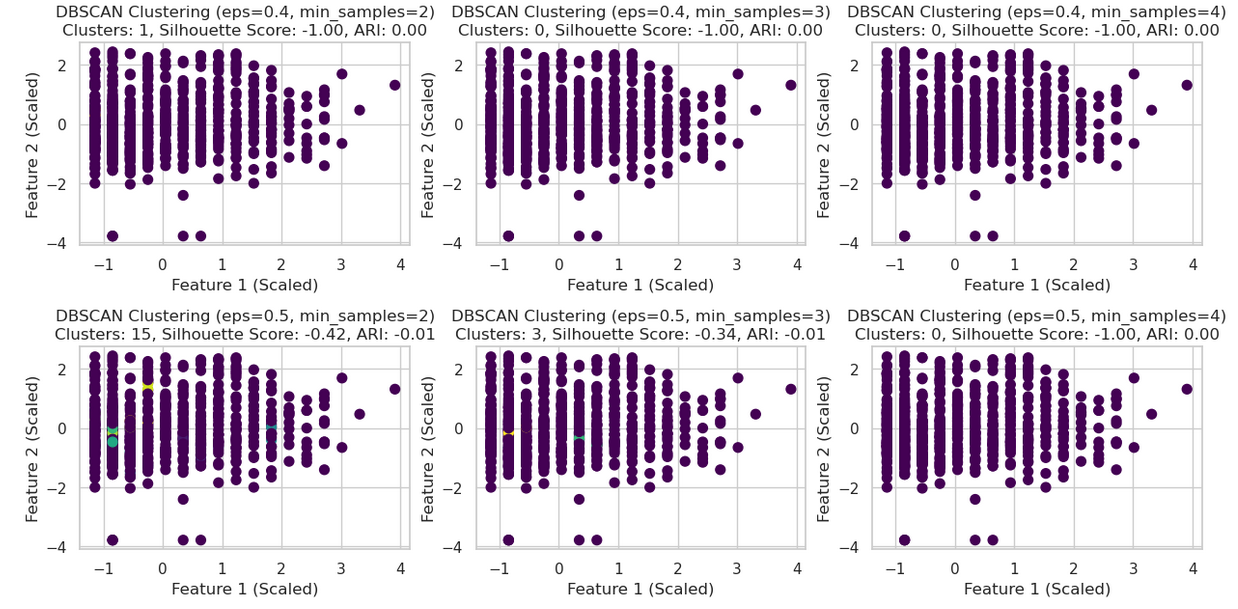
The dendrograms can provide insights into the hierarchical relationships between data points, potentially useful for understanding subgroups within the data.

### DBSCAN Clustering

DBSCAN's performance indicates that the data does not have well-defined density-based clusters.

The choice of eps and min\_samples heavily influences the outcome, and in most cases, DBSCAN failed to identify meaningful clusters.





## Results interpretation:

The clustering results suggest that the data does not naturally group into well-defined clusters using these methods.

For meaningful insights, more sophisticated techniques or domain-specific feature engineering might be necessary.

# Bottom Line Results

## Feature Analysis and Influence on Outcome

Pregnancies

* Mean: 3.85 pregnancies per individual.
* Impact: Higher number of pregnancies can affect insulin sensitivity and glucose metabolism, increasing diabetes risk.

Glucose

* Mean: 120.89 mg/dL.
* Impact: High glucose levels are a primary indicator of diabetes, indicating insulin resistance or insufficient insulin production.

Blood Pressure

* Mean: 69.11 mm Hg.
* Impact: High blood pressure is a common comorbidity of diabetes, contributing to cardiovascular and kidney disease risks.

Skin Thickness

* Mean: 20.54 mm.
* Impact: May be related to obesity and insulin resistance, though not directly influencing diabetes outcome.

Insulin

* Mean: 79.80 μU/ml.
* Impact: Directly related to diabetes, with high levels indicating insulin resistance, a precursor to type 2 diabetes.

BMI

* Mean: 31.99 kg/m^2.
* Impact: High BMI is strongly associated with type 2 diabetes due to its role in insulin resistance and metabolic dysfunction.

Diabetes Pedigree Function

* Mean: 0.47.
* Impact: Higher values indicate a stronger genetic predisposition to diabetes, increasing risk.

Age

* Mean: 33.24 years.
* Impact: Older age increases diabetes likelihood due to reduced insulin sensitivity and lifestyle changes.

## Implemented Models and Algorithms

### Logistic Regression

* Purpose: Binary classification to predict diabetes presence.
* Accuracy: Training (77.65%), Test (78.79%).
* Coefficients: Indicate feature importance.
* Performance Metrics:
  + Precision: 0.78
  + Recall: 0.63
  + Confusion Matrix: Shows true positives, false negatives, etc.

Strengths:

* Interpretability.
* Good performance for linear relationships.

Weaknesses:

* May not capture complex patterns.

### Decision Trees

* Training Accuracy: 100%
* Test Accuracy: 70.56% (overfitting).
* Confusion Matrix:
  + True Positives: 192
  + False Negatives: 0
  + False Positives: 0
  + True Negatives: 345
* Feature Importance: Highlights key predictors like glucose, BMI, and age.

Strengths:

* Interpretability, non-linear relationships.

Weaknesses:

* Overfitting without regularization.

### Association Rules

* Support: Frequency of itemsets in the dataset.
* Confidence: Reliability of the rule.
* Lift: Indicates strength of the association.
* Top Rules:
  + Example: {Glucose=Very High} -> {Outcome=1} with high support, confidence, and lift.

Strengths:

* Discovering patterns and associations

Weaknesses:

* Suitable only for categorical data.

### Clustering

1. K-Means Clustering
   * Silhouette Score: 0.1795
   * ARI Score: 0.1219
   * Interpretation: Poor separation, clusters not well-defined.
2. K-Medoids Clustering
   * Silhouette Score: 0.1212
   * ARI Score: 0.0804
   * Interpretation: Poor agreement with true labels.
3. Hierarchical Clustering
   * Silhouette Scores: Best with 3 or 5 clusters (0.1706 and 0.1778, respectively).
   * Interpretation: Slightly better structure, but generally low scores.
4. DBSCAN Clustering
   * Performance: Sensitive to parameters, often failing to identify meaningful clusters.
   * Interpretation: Data does not naturally group into well-defined clusters using DBSCAN.

## Summary of Findings

Best Model:

* Logistic Regression provided balanced accuracy, interpretability, and performance.

Worst Models:

* Clustering Algorithms (K-Means, K-Medoids, DBSCAN): Failed to identify meaningful clusters, indicating poor natural grouping in the dataset.

Key Insights:

* Glucose and BMI are critical predictors of diabetes.
* High Diabetes Pedigree Function scores indicate genetic predisposition.
* Age and number of pregnancies also contribute to diabetes risk.

### Conclusion

* Identifying Questions: Predictive models, feature influence, and pattern discovery.
* Data Processing: Cleaned, normalized, and binned for categorical analysis.
* Techniques Used: Logistic regression, decision trees, association rule mining, and clustering.
* Best Approach: Logistic regression due to its interpretability and balanced performance.

# Resources:

https://medium.com/analytics-vidhya/statistics-mean-median-mode-variance-standard-deviation-47fab926465a

https://machinelearningmastery.com/how-to-use-correlation-to-understand-the-relationship-between-variables/

<https://www.geeksforgeeks.org/standard-deviation-plot/>

https://www.youtube.com/watch?v=MXvd-bNU93E

https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/

https://www.datacamp.com/tutorial/understanding-logistic-regression-python

<https://books.google.ro/books/about/Data_Mining_with_Python.html?id=LHCd0AEACAAJ&redir_esc=y>

https://medium.com/@rithpansanga/logistic-regression-and-regularization-avoiding-overfitting-and-improving-generalization-e9afdcddd09d

<https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm>

<https://machinelearningmastery.com/calculate-feature-importance-with-python/>

https://books.google.ro/books/about/Data\_Mining\_with\_Python.html?id=LHCd0AEACAAJ&redir\_esc=y

https://www.datacamp.com/tutorial/association-rule-mining-python

<https://www.kaggle.com/code/mervetorkan/association-rules-with-python>