Assignment 1: Part 2

Name: Bismay Parija

Roll Number: 20CS30067

In [1]:
```
# import all the necessary libraries here
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
```

In [2]:
```
# Load the dataset
data = pd.read_csv('../../dataset/decision-tree.csv')
print(data.shape)
feature_names = list(data.keys())[:-1]
print(feature_names)
data.head()
```

```
(768, 9)
['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'Diab
etesPedigreeFunction', 'Age']
```

Out[2]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunc
0	6	148	72	35	0	33.6	0
1	1	85	66	29	0	26.6	0
2	8	183	64	0	0	23.3	0
3	1	89	66	23	94	28.1	0
4	0	137	40	35	168	43.1	2

In [3]:
```
# Separate features and target
X = data.iloc[:, :-1].values
# Reshape to (num_samples, 1)
Y = data.iloc[:, -1].values.reshape(-1,1)

# Split into train, test and validation sets
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=.2, random_stat
#X_train, X_val, Y_train, Y_val = train_test_split(X_train, Y_train, test_size=.125
```

In [4]:
```
class Node():
    def __init__(self, feature_id = None, threshold = None, left = None, right = No
        # For interior nodes
        self.feature_id = feature_id
        self.threshold = threshold
        self.left = left
        self.right = right
```

```
        self.info_gain = info_gain

        # For leaf nodes
        self.label = label
```

In [5]:

```
class ID3():
    def __init__(self, min_samples_leaf = 10):
        self.root = None
        self.min_samples_leaf = min_samples_leaf

    def build_tree(self, data, curr_depth = 0):
        X, Y = data[:, :-1], data[:, -1]
        num_samples, num_features = np.shape(X)

        # Split nodes recursively
        if num_samples >= self.min_samples_leaf:
            # Find the best split
            best_split = self.find_best_split(data, num_samples, num_features)
            # Split the node if information gain is positive
            if best_split["info_gain"] > 0:
                # Generate left and right subtrees
                left = self.build_tree(best_split["left_subset"], curr_depth + 1)
                right = self.build_tree(best_split["right_subset"], curr_depth + 1)
                # Return interior node
                return Node(best_split["feature_id"], best_split["threshold"], left

        # Return leaf node
        leaf_label = self.mode_class(Y)
        return Node(label = leaf_label)

    def find_best_split(self, data, num_samples, num_features):
        # Initialise dictionary to store information about the best split
        best_split = {}
        max_info_gain = -float("inf")

        # Iterate over the features
        for feature_id in range(num_features):
            # Create a list of the labels for the given feature
            labels = data[:, feature_id]
            # Create a list of the unique labels for the given feature
            unique_labels = np.unique(labels)
            # Iterate over the labels of the given feature
            for threshold in unique_labels:
                # Split the node at the current threshold
                left_subset, right_subset = self.split(data, feature_id, threshold)
                # Check if children are non-empty
                if len(left_subset) > 0 and len(right_subset) > 0:
                    # Compute information gain
                    y, y_left, y_right = data[:, -1], left_subset[:, -1], right_sub
                    curr_info_gain = self.information_gain(y, y_left, y_right)
                    # Update the best split
                    if curr_info_gain > max_info_gain:
                        best_split["feature_id"] = feature_id
                        best_split["threshold"] = threshold
                        best_split["left_subset"] = left_subset
                        best_split["right_subset"] = right_subset
```

```
                    best_split["info_gain"] = curr_info_gain
                    max_info_gain = curr_info_gain

        # Return the most informative split
        return best_split

    def split(self, data, feature_id, threshold):
        # Split the dataset into left and right subsets
        left_subset = np.array([row for row in data if row[feature_id] <= threshold
        right_subset = np.array([row for row in data if row[feature_id] > threshold
        return left_subset, right_subset

    def information_gain(self, parent, left, right):
        left_weight = len(left) / len(parent)
        right_weight = len(right) / len(parent)
        gain = self.entropy(parent) - (left_weight * self.entropy(left) + right_wei
        return gain

    def entropy(self, y):
        labels = np.unique(y)
        entropy = 0
        for label in labels:
            p_class = len(y[y == label]) / len(y)
            entropy += p_class * np.log2(p_class)
        return -entropy

    def mode_class(self, Y):
        Y = list(Y)
        return max(Y, key = Y.count)

    def print_tree(self, tree=None, depth = 0):
        if tree is None:
            tree = self.root

        # Print information about leaf node
        if tree.label is not None:
            print("  " * depth + f"Leaf Node: Class: {tree.label}")

        # Print information about interior node
        else:
            print("  " * depth + f"Interior Node:")
            print("  " * (depth + 1) + f"Feature: {feature_names[tree.feature_id]}

            # Recursively print left and right subtrees
            print("  " * (depth + 1) + f"Left Node:")
            self.print_tree(tree.left, depth + 1)
            print("  " * (depth + 1) + f"Right Node:")
            self.print_tree(tree.right, depth + 1)

    def fit(self, X, Y):
        data = np.concatenate((X, Y), axis=1)
        self.root = self.build_tree(data)

    def predict(self, X):
        predictions = [self.make_prediction(x, self.root) for x in X]
        return predictions
```

```
    def make_prediction(self, x, tree):
        if tree.label != None:
            return tree.label
        feature_val = x[tree.feature_id]
        if feature_val <= tree.threshold:
            return self.make_prediction(x, tree.left)
        else:
            return self.make_prediction(x, tree.right)
```

In [6]:

```
# Initialize and train your ID3 model
model = ID3(min_samples_leaf=10)
model.fit(X_train, Y_train)
model.print_tree()

Y_pred = model.predict(X_test)
print(f"Accuracy score without pruning {accuracy_score(Y_test, Y_pred)}")
```

```
Interior Node:
  Feature: Glucose Threshold: 127.0 Information Gain: 0.1277
  Left Node:
  Interior Node:
    Feature: Age Threshold: 28.0 Information Gain: 0.0854
    Left Node:
    Interior Node:
      Feature: BMI Threshold: 30.9 Information Gain: 0.0697
      Left Node:
      Interior Node:
        Feature: DiabetesPedigreeFunction Threshold: 0.666 Information Gain: 0.0241
        Left Node:
        Leaf Node: Class: 0.0
        Right Node:
        Interior Node:
          Feature: DiabetesPedigreeFunction Threshold: 0.678 Information Gain: 0.337
3
          Left Node:
          Leaf Node: Class: 1.0
          Right Node:
          Leaf Node: Class: 0.0
      Right Node:
      Interior Node:
        Feature: BloodPressure Threshold: 30.0 Information Gain: 0.0552
        Left Node:
        Leaf Node: Class: 1.0
        Right Node:
        Interior Node:
          Feature: DiabetesPedigreeFunction Threshold: 0.498 Information Gain: 0.042
1
          Left Node:
          Interior Node:
            Feature: SkinThickness Threshold: 0.0 Information Gain: 0.0937
            Left Node:
            Leaf Node: Class: 0.0
            Right Node:
            Interior Node:
              Feature: BMI Threshold: 31.6 Information Gain: 0.1545
              Left Node:
              Leaf Node: Class: 1.0
              Right Node:
              Leaf Node: Class: 0.0
          Right Node:
          Interior Node:
            Feature: DiabetesPedigreeFunction Threshold: 0.543 Information Gain: 0.1
142
            Left Node:
            Leaf Node: Class: 1.0
            Right Node:
            Interior Node:
              Feature: BloodPressure Threshold: 66.0 Information Gain: 0.1312
              Left Node:
              Interior Node:
                Feature: Glucose Threshold: 85.0 Information Gain: 0.4200
                Left Node:
                Leaf Node: Class: 0.0
```

```
                Right Node:
                Leaf Node: Class: 1.0
              Right Node:
              Interior Node:
                Feature: Age Threshold: 21.0 Information Gain: 0.1565
                Left Node:
                Leaf Node: Class: 0.0
                Right Node:
                Leaf Node: Class: 0.0
    Right Node:
    Interior Node:
      Feature: BMI Threshold: 26.2 Information Gain: 0.1074
      Left Node:
      Interior Node:
        Feature: BMI Threshold: 0.0 Information Gain: 0.1914
        Left Node:
        Leaf Node: Class: 1.0
        Right Node:
        Leaf Node: Class: 0.0
      Right Node:
      Interior Node:
        Feature: Glucose Threshold: 94.0 Information Gain: 0.0887
        Left Node:
        Interior Node:
          Feature: Glucose Threshold: 0.0 Information Gain: 0.2161
          Left Node:
          Leaf Node: Class: 1.0
          Right Node:
          Interior Node:
            Feature: Pregnancies Threshold: 9.0 Information Gain: 0.1537
            Left Node:
            Leaf Node: Class: 0.0
            Right Node:
            Leaf Node: Class: 0.0
        Right Node:
        Interior Node:
          Feature: DiabetesPedigreeFunction Threshold: 0.512 Information Gain: 0.063
0
          Left Node:
          Interior Node:
            Feature: SkinThickness Threshold: 27.0 Information Gain: 0.0851
            Left Node:
            Interior Node:
              Feature: BloodPressure Threshold: 82.0 Information Gain: 0.2233
              Left Node:
              Interior Node:
                Feature: DiabetesPedigreeFunction Threshold: 0.38 Information Gain:
0.0946
                Left Node:
                Interior Node:
                  Feature: Pregnancies Threshold: 7.0 Information Gain: 0.2075
                  Left Node:
                  Interior Node:
                    Feature: BloodPressure Threshold: 60.0 Information Gain: 0.2577
                    Left Node:
                    Leaf Node: Class: 0.0
```

```
                  Right Node:
                  Leaf Node: Class: 1.0
                Right Node:
                Leaf Node: Class: 0.0
              Right Node:
              Leaf Node: Class: 0.0
            Right Node:
            Interior Node:
              Feature: SkinThickness Threshold: 12.0 Information Gain: 0.4395
              Left Node:
              Leaf Node: Class: 0.0
              Right Node:
              Leaf Node: Class: 1.0
          Right Node:
          Interior Node:
            Feature: BloodPressure Threshold: 88.0 Information Gain: 0.1745
            Left Node:
            Interior Node:
              Feature: BloodPressure Threshold: 66.0 Information Gain: 0.1683
              Left Node:
              Leaf Node: Class: 0.0
              Right Node:
              Interior Node:
                Feature: Insulin Threshold: 135.0 Information Gain: 0.1864
                Left Node:
                Leaf Node: Class: 0.0
                Right Node:
                Leaf Node: Class: 0.0
            Right Node:
            Leaf Node: Class: 1.0
        Right Node:
        Interior Node:
          Feature: Pregnancies Threshold: 6.0 Information Gain: 0.2220
          Left Node:
          Interior Node:
            Feature: BloodPressure Threshold: 86.0 Information Gain: 0.1439
            Left Node:
            Interior Node:
              Feature: BloodPressure Threshold: 68.0 Information Gain: 0.1699
              Left Node:
              Leaf Node: Class: 0.0
              Right Node:
              Interior Node:
                Feature: Glucose Threshold: 119.0 Information Gain: 0.3204
                Left Node:
                Leaf Node: Class: 1.0
                Right Node:
                Leaf Node: Class: 0.0
            Right Node:
            Leaf Node: Class: 0.0
          Right Node:
          Leaf Node: Class: 1.0
Right Node:
Interior Node:
  Feature: BMI Threshold: 29.9 Information Gain: 0.1015
  Left Node:
```

```
Interior Node:
  Feature: Glucose Threshold: 145.0 Information Gain: 0.1493
  Left Node:
  Interior Node:
    Feature: Pregnancies Threshold: 2.0 Information Gain: 0.0953
    Left Node:
    Leaf Node: Class: 0.0
    Right Node:
    Interior Node:
      Feature: Pregnancies Threshold: 4.0 Information Gain: 0.2365
      Left Node:
      Interior Node:
        Feature: Glucose Threshold: 139.0 Information Gain: 0.1710
        Left Node:
        Leaf Node: Class: 0.0
        Right Node:
        Leaf Node: Class: 0.0
      Right Node:
      Leaf Node: Class: 0.0
  Right Node:
  Interior Node:
    Feature: Age Threshold: 59.0 Information Gain: 0.1294
    Left Node:
    Interior Node:
      Feature: Age Threshold: 40.0 Information Gain: 0.2008
      Left Node:
      Interior Node:
        Feature: Glucose Threshold: 159.0 Information Gain: 0.2490
        Left Node:
        Interior Node:
          Feature: Pregnancies Threshold: 2.0 Information Gain: 0.3219
          Left Node:
          Leaf Node: Class: 0.0
          Right Node:
          Leaf Node: Class: 0.0
        Right Node:
        Leaf Node: Class: 1.0
      Right Node:
      Interior Node:
        Feature: Pregnancies Threshold: 9.0 Information Gain: 0.1935
        Left Node:
        Leaf Node: Class: 1.0
        Right Node:
        Leaf Node: Class: 1.0
    Right Node:
    Leaf Node: Class: 0.0
Right Node:
Interior Node:
  Feature: Glucose Threshold: 165.0 Information Gain: 0.0717
  Left Node:
  Interior Node:
    Feature: BloodPressure Threshold: 90.0 Information Gain: 0.0588
    Left Node:
    Interior Node:
      Feature: BloodPressure Threshold: 60.0 Information Gain: 0.0586
      Left Node:
```

```
Interior Node:
  Feature: Age Threshold: 39.0 Information Gain: 0.2374
  Left Node:
  Leaf Node: Class: 1.0
  Right Node:
  Leaf Node: Class: 1.0
Right Node:
Interior Node:
  Feature: Age Threshold: 30.0 Information Gain: 0.0982
  Left Node:
  Interior Node:
    Feature: Insulin Threshold: 250.0 Information Gain: 0.2170
    Left Node:
    Interior Node:
      Feature: Glucose Threshold: 154.0 Information Gain: 0.1280
      Left Node:
      Interior Node:
        Feature: BloodPressure Threshold: 84.0 Information Gain: 0.1130
        Left Node:
        Interior Node:
          Feature: BloodPressure Threshold: 72.0 Information Gain: 0.1514
          Left Node:
          Interior Node:
            Feature: Insulin Threshold: 166.0 Information Gain: 0.2488
            Left Node:
            Interior Node:
              Feature: Glucose Threshold: 128.0 Information Gain: 0.1710
              Left Node:
              Leaf Node: Class: 0.0
              Right Node:
              Leaf Node: Class: 0.0
            Right Node:
            Leaf Node: Class: 1.0
          Right Node:
          Leaf Node: Class: 0.0
        Right Node:
        Leaf Node: Class: 1.0
      Right Node:
      Leaf Node: Class: 1.0
    Right Node:
    Leaf Node: Class: 0.0
  Right Node:
  Interior Node:
    Feature: Age Threshold: 33.0 Information Gain: 0.0553
    Left Node:
    Leaf Node: Class: 1.0
    Right Node:
    Interior Node:
      Feature: SkinThickness Threshold: 44.0 Information Gain: 0.0561
      Left Node:
      Interior Node:
        Feature: BloodPressure Threshold: 74.0 Information Gain: 0.0557
        Left Node:
        Interior Node:
          Feature: Pregnancies Threshold: 6.0 Information Gain: 0.3167
          Left Node:
```

```
                    Leaf Node: Class: 1.0
                    Right Node:
                    Leaf Node: Class: 1.0
                  Right Node:
                  Interior Node:
                    Feature: Pregnancies Threshold: 9.0 Information Gain: 0.1427
                    Left Node:
                    Interior Node:
                      Feature: Glucose Threshold: 155.0 Information Gain: 0.1271
                      Left Node:
                      Interior Node:
                        Feature: Glucose Threshold: 151.0 Information Gain: 0.1251
                        Left Node:
                        Interior Node:
                          Feature: SkinThickness Threshold: 15.0 Information Gain:
0.2533
                          Left Node:
                          Leaf Node: Class: 0.0
                          Right Node:
                          Leaf Node: Class: 1.0
                        Right Node:
                        Leaf Node: Class: 0.0
                      Right Node:
                      Leaf Node: Class: 1.0
                    Right Node:
                    Leaf Node: Class: 0.0
                Right Node:
                Leaf Node: Class: 1.0
        Right Node:
        Leaf Node: Class: 1.0
      Right Node:
      Interior Node:
        Feature: Pregnancies Threshold: 2.0 Information Gain: 0.0551
        Left Node:
        Leaf Node: Class: 1.0
        Right Node:
        Interior Node:
          Feature: Glucose Threshold: 178.0 Information Gain: 0.1116
          Left Node:
          Leaf Node: Class: 1.0
          Right Node:
          Interior Node:
            Feature: DiabetesPedigreeFunction Threshold: 1.213 Information Gain: 0.2
188
            Left Node:
            Interior Node:
              Feature: DiabetesPedigreeFunction Threshold: 0.299 Information Gain:
0.2650
              Left Node:
              Leaf Node: Class: 1.0
              Right Node:
              Leaf Node: Class: 1.0
            Right Node:
            Leaf Node: Class: 0.0
Accuracy score without pruning 0.7402597402597403
```