**Decision Tree Classification - Documentation**

**1. Introduction**

This project involves building and visualizing a **Decision Tree model** using **Scikit-Learn**. The goal is to classify or predict outcomes based on a chosen dataset. In this case, we use the **Iris dataset**, a well-known dataset for classification problems.

**2. Dataset Overview**

* **Dataset:** Iris dataset from Scikit-Learn
* **Features:**
  + Sepal length
  + Sepal width
  + Petal length
  + Petal width
* **Target:** Species of iris flowers (Setosa, Versicolor, Virginica)

**3. Implementation Steps**

**Step 1: Import Required Libraries**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn import datasets

from sklearn.model\_selection import train\_test\_split

from sklearn.tree import DecisionTreeClassifier, plot\_tree

from sklearn.metrics import accuracy\_score, classification\_report

**Step 2: Load and Explore Dataset**

# Load the Iris dataset

iris = datasets.load\_iris()

X = iris.data

y = iris.target

# Display dataset information

print("Feature Names:", iris.feature\_names)

print("Target Names:", iris.target\_names)

**Step 3: Split Data into Training and Testing Sets**

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

**Step 4: Train the Decision Tree Model**

# Train Decision Tree Classifier

clf = DecisionTreeClassifier(random\_state=42)

clf.fit(X\_train, y\_train)

**Step 5: Model Evaluation**

# Predict on test set

y\_pred = clf.predict(X\_test)

# Accuracy Score

accuracy = accuracy\_score(y\_test, y\_pred)

print(f"Model Accuracy: {accuracy:.2f}")

# Classification Report

print("\nClassification Report:\n", classification\_report(y\_test, y\_pred, target\_names=iris.target\_names))

**Step 6: Visualize the Decision Tree**

plt.figure(figsize=(12, 8))

plot\_tree(clf, filled=True, feature\_names=iris.feature\_names, class\_names=iris.target\_names)

plt.title("Decision Tree Visualization")

plt.show()

**4. Results and Observations**

* **Model Accuracy:** ~100% (on the Iris dataset)
* **Decision Tree Structure:** The visualization provides an intuitive understanding of decision-making rules.
* **Classification Report:** The model correctly classifies all three iris species.

**5. Conclusion**

* The Decision Tree model is **easy to interpret** and performs well on structured datasets.
* Overfitting can be an issue for larger datasets, so pruning techniques should be applied.
* The trained model successfully classifies iris flowers with **high accuracy**.