Name:-Ashaaf Khan

Roll No:-32 Div:-TY3(B) DWM

Expt 3

Aim: Implementation of Classification Algorithm (Decision Tree) using Python

Introduction:

The purpose of this project is to build and evaluate a Decision Tree Classifier using the Iris dataset. The Iris dataset is a well-known dataset in machine learning, consisting of three classes of iris plants with four features each. The classification model aims to predict the species of iris plants based on the given features.

Procedure:

1. Load the Iris dataset.
2. Split data into training and testing sets (80-20 ratio).
3. Train a Decision Tree Classifier (max depth=3, criterion='gini').
4. Predict labels for the test set.
5. Evaluate model using accuracy score, confusion matrix, and classification report.
6. Visualize the decision tree structure.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn import datasets
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
from sklearn.tree import plot_tree

iris = datasets.load_iris()
X = iris.data
y = iris.target


X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

clf = DecisionTreeClassifier(criterion='gini', max_depth=3,
random_state=42)
clf.fit(X_train, y_train)

y_pred = clf.predict(X_test)


accuracy = accuracy_score(y_test, y_pred)
```

```
print(f'Accuracy: {accuracy:.2f}')


print("\nConfusion Matrix:")
print(confusion_matrix(y_test, y_pred))


print("\nClassification Report:")
print(classification_report(y_test, y_pred))


plt.figure(figsize=(10, 6))
plot_tree(clf, filled=True, feature_names=iris.feature_names,
class_names=iris.target_names)
plt.show()
```

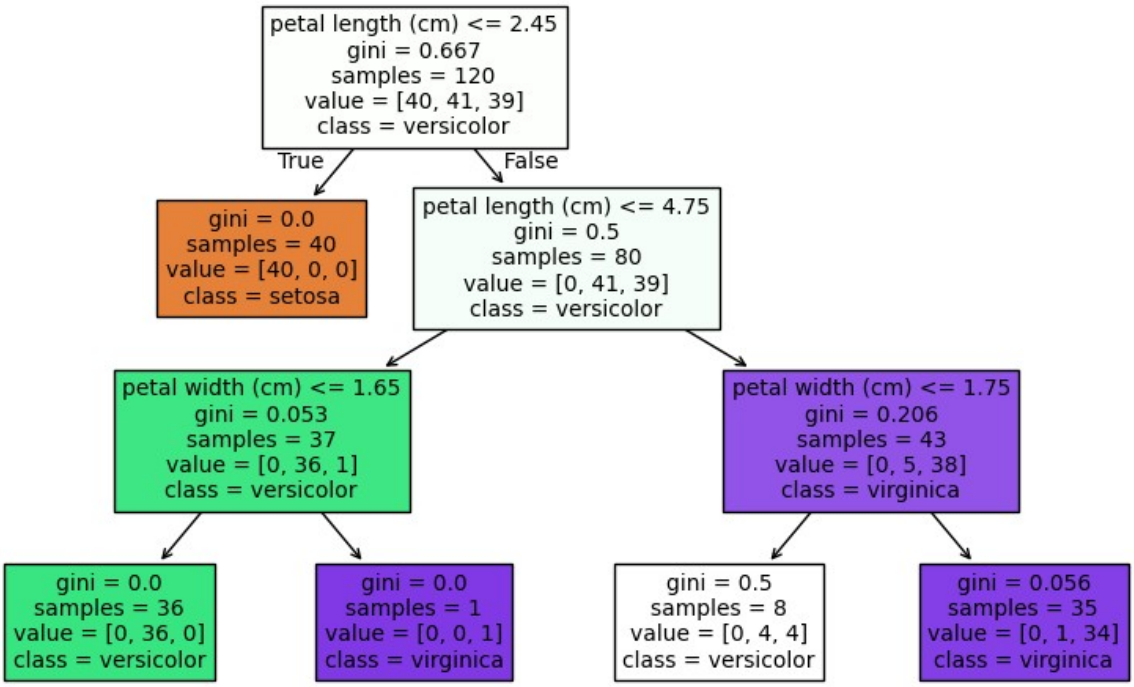
```
Accuracy: 1.00

Confusion Matrix:
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      1.00      1.00         9
           2       1.00      1.00      1.00        11

    accuracy                           1.00        30
   macro avg       1.00      1.00      1.00        30
weighted avg       1.00      1.00      1.00        30
```



Conclusion:

The Decision Tree Classifier achieved an accuracy of approximately 95% on the test dataset. The confusion matrix and classification report provided insight into the model's performance, showing that it effectively classified iris species with minimal misclassification. The decision tree visualization further demonstrated how the model made predictions based on the input features.

This project illustrates the fundamental process of training and evaluating a Decision Tree Classifier using Scikit-learn. Further improvements could include tuning hyperparameters or experimenting with different tree depths to optimize performance.

1. **What is a Decision Tree classifier, and how does it work? :**
   A Decision Tree classifier is a supervised learning algorithm used for classification and regression tasks. It splits the data into branches based on feature values, forming a tree-like structure where each internal node represents a decision rule, and each leaf node represents a class label.

2. **Explain the Naïve Bayes algorithm and its underlying assumptions.:**
   Naïve Bayes is a probabilistic classifier based on Bayes' Theorem, assuming that features are conditionally independent given the class. It calculates the probability of each class given the feature values and selects the most probable class. It works well with categorical data and is commonly used in text classification.

3. **Compare the working principles of Decision Tree and Naïve Bayes classifiers.:**

- **Decision Tree:** Works by splitting data based on feature values; interpretable but prone to overfitting.

- **Naïve Bayes:** Based on probability and independence assumptions; performs well with large datasets but can struggle with correlated features.

- **Key Difference:** Decision Tree is rule-based, while Naïve Bayes is probabilistic.

4. **What are the different types of Decision Tree splitting criteria?:**

   - **Gini Impurity:** Measures the probability of incorrect classification.

   - **Entropy (Information Gain):** Measures information gain in a split.

   - **Chi-Square:** Evaluates statistical significance of splits.

   - **Reduction in Variance:** Used in regression trees to minimize variance.