Decision Tree Implementation

Author: Srikar Kalle

Student ID: C00313529

Change Log

SL No.	Change Category	Description	Duration (mins)	Difficulty (1-10)
1	Dataset Update	Replaced synthetic dataset (`make_blobs`) with Penguins dataset for real-world classification.	15	4
2	Feature Engineering	Encoded categorical variables (`species` , `island` , `sex`) using `LabelEncoder()` .	10	5
3	Preprocessing	Dropped `species` from the features to use as target variable.	5	3
4	Train-Test Split	Implemented `train_test_split` (80/20 split) with `random_state=42` for reproducibility.	10	4
5	Model Training	Trained `DecisionTreeClassifier` on Penguins dataset.	20	6
6	Model Evaluation	Added `accuracy_score` and `classification_report` for evaluation.	15	6
7	Visualization Update	Used `plot_tree()` with feature names for improved readability.	15	5

In [1]:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import accuracy_score, classification_report
```

In [2]:

```python
# Load the Penguins dataset
penguins = sns.load_dataset("penguins")

# Drop rows with missing values
penguins.dropna(inplace=True)
```

In [3]:

```python
# Encode categorical variables
encoder = LabelEncoder()
```

```
penguins['species'] = encoder.fit_transform(penguins['species'])
penguins['island'] = encoder.fit_transform(penguins['island'])
penguins['sex'] = encoder.fit_transform(penguins['sex'])
```

In [4]:

```
# Define features and target
X = penguins.drop(columns=['species'])  # Features
y = penguins['species']  # Target
```

In [5]:

```
# 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_
```

In [6]:

```
# Initialize and train Decision Tree Classifier
clf = DecisionTreeClassifier(random_state=42)
clf.fit(X_train, y_train)
```

Out[6]:

```
▾     DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
```

In [7]:

```
# Make predictions
y_pred = clf.predict(X_test)
```

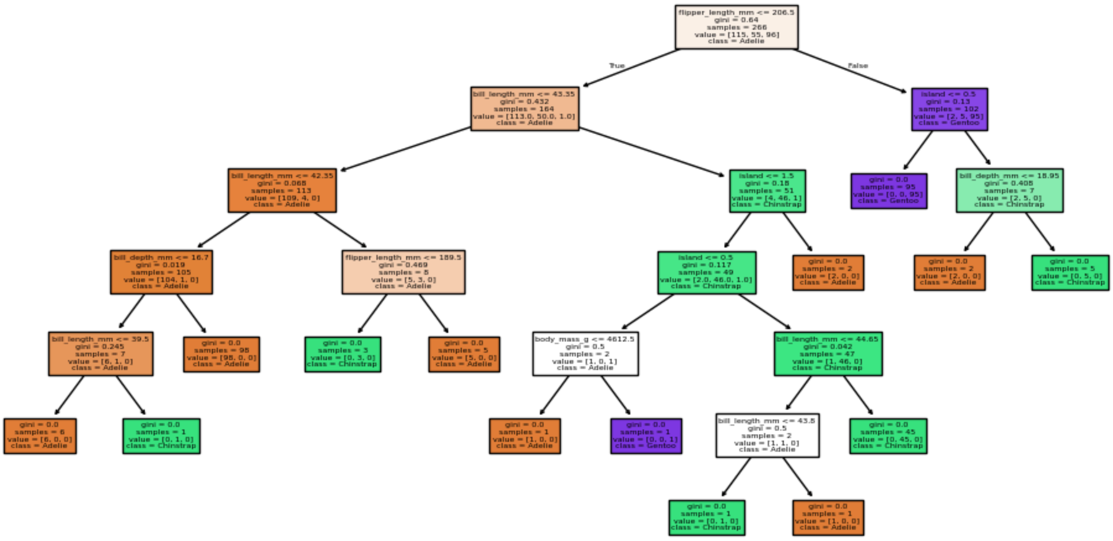
In [8]:

```
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred, target_names=['Adelie', 'Chinstra

# Print evaluation results
print(f"Model Accuracy: {accuracy:.2f}")
print("\nClassification Report:\n", report)
```

```
Model Accuracy: 1.00

Classification Report:
               precision    recall  f1-score   support

      Adelie       1.00      1.00      1.00        31
   Chinstrap       1.00      1.00      1.00        13
      Gentoo       1.00      1.00      1.00        23

    accuracy                           1.00        67
   macro avg       1.00      1.00      1.00        67
weighted avg       1.00      1.00      1.00        67
```

In [9]:

```
# Plot the Decision Tree
plt.figure(figsize=(12, 6))
plot_tree(clf, feature_names=X.columns, class_names=['Adelie', 'Chinstrap', 'Gen
plt.show()
```


flipper_length_mm <= 206.5
gini = 0.64
samples = 266
value = [115, 55, 96]
class = Adelie
True
False
bill_length_mm <= 43.35
gini = 0.432
samples = 164
value = [113.0, 50.0, 1.0]
class = Adelie
island <= 0.5
gini = 0.13
samples = 102
value = [2, 5, 95]
class = Gentoo
bill_length_mm <= 42.35
gini = 0.068
samples = 113
value = [109, 4, 0]
class = Adelie
island <= 1.5
gini = 0.18
samples = 51
value = [4, 46, 1]
class = Chinstrap
gini = 0.0
samples = 95
value = [0, 0, 95]
class = Gentoo
bill_depth_mm <= 18.95
gini = 0.408
samples = 7
value = [2, 5, 0]
class = Chinstrap
bill_depth_mm <= 16.7
gini = 0.019
samples = 105
value = [104, 1, 0]
class = Adelie
flipper_length_mm <= 189.5
gini = 0.469
samples = 8
value = [5, 3, 0]
class = Adelie
island <= 0.5
gini = 0.117
samples = 49
value = [2.0, 46.0, 1.0]
class = Chinstrap
gini = 0.0
samples = 2
value = [2, 0, 0]
class = Adelie
gini = 0.0
samples = 2
value = [2, 0, 0]
class = Adelie
gini = 0.0
samples = 5
value = [0, 5, 0]
class = Chinstrap
bill_length_mm <= 39.5
gini = 0.245
samples = 7
value = [6, 1, 0]
class = Adelie
gini = 0.0
samples = 98
value = [98, 0, 0]
class = Adelie
gini = 0.0
samples = 3
value = [0, 3, 0]
class = Chinstrap
gini = 0.0
samples = 5
value = [5, 0, 0]
class = Adelie
body_mass_g <= 4612.5
gini = 0.5
samples = 2
value = [1, 0, 1]
class = Adelie
bill_length_mm <= 44.65
gini = 0.042
samples = 47
value = [1, 46, 0]
class = Chinstrap
gini = 0.0
samples = 6
value = [6, 0, 0]
class = Adelie
gini = 0.0
samples = 1
value = [0, 1, 0]
class = Chinstrap
gini = 0.0
samples = 1
value = [1, 0, 0]
class = Adelie
gini = 0.0
samples = 1
value = [0, 0, 1]
class = Gentoo
bill_length_mm <= 43.8
gini = 0.5
samples = 2
value = [1, 1, 0]
class = Adelie
gini = 0.0
samples = 45
value = [0, 45, 0]
class = Chinstrap
gini = 0.0
samples = 1
value = [0, 1, 0]
class = Chinstrap
gini = 0.0
samples = 1
value = [1, 0, 0]
class = Adelie