import pandas as pd

import numpy as np

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

from sklearn.tree import DecisionTreeClassifier, plot\_tree

from sklearn.metrics import accuracy\_score

import matplotlib.pyplot as plt

# ---- Step 1: Load dataset ----

# Make sure your Social\_Network\_Ads.csv is in the same folder as this script

data = pd.read\_csv("/content/sample\_data/Social\_Networks\_Ads.csv")

# Display first few rows

print("Sample Data:")

print(data.head())

# ---- Step 2: Preprocess Data ----

# Convert categorical 'Gender' column into numeric (Male=1, Female=0)

data['Gender'] = data['Gender'].map({'Male': 1, 'Female': 0})

# Select features and target

X = data[['Age', 'EstimatedSalary', 'Gender']]

y = data['Purchased']

# ---- Step 3: Split into training and testing sets ----

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.25, random\_state=42)

# ---- Step 4: Model 1 - Using default (Gini) ----

clf\_gini = DecisionTreeClassifier(random\_state=42)

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

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

print("\n--- Gini Criterion ---")

print("Accuracy on Train Data:", accuracy\_score(y\_train, clf\_gini.predict(X\_train)))

print("Accuracy on Test Data :", accuracy\_score(y\_test, y\_pred\_gini))

# ---- Step 5: Model 2 - Using Entropy ----

clf\_entropy = DecisionTreeClassifier(criterion='entropy', random\_state=42)

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

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

print("\n--- Entropy Criterion ---")

print("Accuracy on Train Data:", accuracy\_score(y\_train, clf\_entropy.predict(X\_train)))

print("Accuracy on Test Data :", accuracy\_score(y\_test, y\_pred\_entropy))

# ---- Step 6: Model 3 - Entropy + min\_samples\_split ----

clf\_entropy\_min\_split = DecisionTreeClassifier(criterion='entropy', min\_samples\_split=20, random\_state=42)

clf\_entropy\_min\_split.fit(X\_train, y\_train)

y\_pred\_entropy\_min\_split = clf\_entropy\_min\_split.predict(X\_test)

print("\n--- Entropy + min\_samples\_split ---")

print("Accuracy on Train Data:", accuracy\_score(y\_train, clf\_entropy\_min\_split.predict(X\_train)))

print("Accuracy on Test Data :", accuracy\_score(y\_test, y\_pred\_entropy\_min\_split))

# ---- Step 7: Visualization ----

plt.figure(figsize=(14, 8))

plot\_tree(clf\_gini,

          feature\_names=X.columns,

          class\_names=['Not Purchased', 'Purchased'],

          filled=True,

          rounded=True)

plt.title("Decision Tree Visualization (Gini Criterion)")

plt.show()