import pandas as pd

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

from sklearn.preprocessing import StandardScaler

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import confusion\_matrix, classification\_report

import matplotlib.pyplot as plt

import seaborn as sns

# Load dataset

dataset = pd.read\_csv('../Data/Social\_Network\_Ads.csv')

print("First 5 rows of the dataset:\n", dataset.head())

# Select features and target

X = dataset[['Age', 'EstimatedSalary']].values

y = dataset['Purchased'].values

# Split data into train and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

X, y, test\_size=0.25, random\_state=0

)

# Standardize features

sc = StandardScaler()

X\_train = sc.fit\_transform(X\_train)

X\_test = sc.transform(X\_test)

# Train Logistic Regression model

classifier = LogisticRegression(random\_state=0)

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

# Predict on test data

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

# Evaluate performance

cm = confusion\_matrix(y\_test, y\_pred)

print("Confusion Matrix:\n", cm)

cl\_report = classification\_report(y\_test, y\_pred)

print("Classification Report:\n", cl\_report)

# Optional: Visualize the confusion matrix

plt.figure(figsize=(6, 4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')

plt.title('Confusion Matrix')

plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.tight\_layout()

plt.show()