

7]

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
dataset = pd.read_csv('Social_Network_Ads.csv')
if 'Gender' in dataset.columns:
    from sklearn.preprocessing import LabelEncoder
    le = LabelEncoder()
    dataset['Gender'] = le.fit_transform(dataset['Gender'])
X = dataset.iloc[:, [1, 2, 3]].values
y = dataset.iloc[:, -1].values
print("First few rows of the dataset:")
print(dataset.head())
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42)
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
print("\nScaled Training Features:")
print(X_train)
from sklearn.neighbors import KNeighborsClassifier
classifier = KNeighborsClassifier(
    n_neighbors=5, metric='minkowski', p=2
)
classifier.fit(X_train, y_train)
custom_prediction = classifier.predict(sc.transform([[1, 46, 28000]]))
print("\nPrediction for [Gender=Male(1), Age=46, EstimatedSalary=28000]:")
print(custom_prediction[0])
y_pred = classifier.predict(X_test)
```

```
results = np.concatenate(  
    (y_pred.reshape(len(y_pred), 1), y_test.reshape(len(y_test), 1)), axis=1)  
  
print("\nPredicted vs Actual values:")  
  
print(results)  
  
from sklearn.metrics import confusion_matrix, accuracy_score  
  
cm = confusion_matrix(y_test, y_pred)  
  
accuracy = accuracy_score(y_test, y_pred)  
  
print("\nConfusion Matrix:")  
  
print(cm)  
  
print(f"\nAccuracy Score: {accuracy:.2f}")
```