
Experiment No: 09

Code:

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.neighbors import KNeighborsClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder

from sklearn.metrics import confusion_matrix, classification_report, accuracy_score

import matplotlib

matplotlib.use('Agg') # Use non-interactive backend

import matplotlib.pyplot as plt

import seaborn as sns


# Load the data

data =
pd.read_csv(r'D:\Current_Learning\TY_NOTES\ML\Practical\Exp_No_09\Experiment9.csv'
)


# Print columns to verify

print("Columns in dataset:", data.columns)


# Encode categorical variables

label_encoder = LabelEncoder()

if 'Gender' in data.columns:

    data['Gender'] = label_encoder.fit_transform(data['Gender'])
```

```
# Select features and the target variable for KNN

X_knn = data[['Age', 'Gender']]

y_knn = data['Purchased']


# Split data into training and testing sets for KNN

X_train_knn, X_test_knn, y_train_knn, y_test_knn = train_test_split(X_knn, y_knn,
test_size=0.3, random_state=42)


# Initialize and train the KNN classifier

knn_classifier = KNeighborsClassifier(n_neighbors=5)

knn_classifier.fit(X_train_knn, y_train_knn)


# Make predictions on the test set

y_pred_knn = knn_classifier.predict(X_test_knn)


# Generate and display the confusion matrix

conf_matrix_knn = confusion_matrix(y_test_knn, y_pred_knn)

plt.figure(figsize=(8, 5))

sns.heatmap(conf_matrix_knn, annot=True, fmt="d", cmap="Blues", cbar=False,
            xticklabels=['Not Purchased', 'Purchased'], yticklabels=['Not Purchased', 'Purchased'])

plt.xlabel('Predicted Label')

plt.ylabel('True Label')

plt.title('KNN Classification Confusion Matrix')


# Save the figure instead of showing it

plt.savefig('knn_confusion_matrix.png')
```

```
plt.close()
```

```
# Print classification report for additional performance metrics
```

```
print("\nClassification Report for KNN:\n", classification_report(y_test_knn, y_pred_knn))
```

Output:

```
Microsoft Windows [Version 10.0.26100.2314]
(c) Microsoft Corporation. All rights reserved.

D:\Current_Learning\TY_NOTES\ML\Practical\Exp_No_09>python -u "d:\Current_Learning\TY_NOTES\ML\Practical\Exp_No_09\tempCodeRunnerFile.py"
Columns in dataset: Index(['User ID', 'Gender', 'Age', 'EstimatedSalary', 'Purchased'], dtype='object')

Classification Report for KNN:
              precision    recall  f1-score   support

     0       0.84         0.88         0.86         73
     1       0.80         0.74         0.77         47

 accuracy          0.82
 macro avg          0.82
weighted avg          0.82
```

