Project Title: Classification of Students Based on Study Methods

Name: Anish Chaudhary

Roll No.: 20240110030046

Course: B.Tech in Computer Science and Engineering (AI)

Section: A

Date: April 22, 2025

Problem Statement:The aim of this project is to classify students into different learning styles—visual, auditory, and kinesthetic—based on their responses to a study method questionnaire. Using machine learning techniques, this classification will aid in understanding student behavior and tailoring educational strategies accordingly

Introduction

The objective of this project is to classify students based on their preferred study methods: visual, auditory, or kinesthetic. We use data collected through a questionnaire which measures students' preferences on these three learning modalities. Classification helps in personalizing the learning experience, improving academic performance, and understanding learning behaviors.

Methodology

1. Data Collection: The dataset consists of scores for visual, auditory, and kinesthetic learningpreferences, along with labeled learning styles.

2. Preprocessing:
    - Label encoding was used to convert categorical target variables into numeric form.
    - The dataset was split into training and testing subsets using an 80-20 split.

3. Model Selection: A Random Forest Classifier was chosen for its ability to handle multi-classclassification and avoid overfitting.

4. Evaluation: The model was evaluated using Accuracy, Precision, Recall, F1-Score, and aconfusion matrix.

Code

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix, accuracy_score, precision_score,
recall_score, f1_score

df = pd.read_csv("student_methods.csv") X = df[['visual_score',
'auditory_score', 'kinesthetic_score']] y =
df['learning_style']

le = LabelEncoder()
y_encoded = le.fit_transform(y)

X_train, X_test, y_train, y_test = train_test_split(X, y_encoded, test_size=0.2,
random_state=42)
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train) y_pred =
model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted') f1 =
f1_score(y_test, y_pred, average='weighted')

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='coolwarm',
xticklabels=le.classes_, yticklabels=le.classes_)
plt.xlabel("Predicted") plt.ylabel("Actual")
plt.title("Confusion Matrix Heatmap")
plt.savefig("confusion_matrix_heatmap.png")
plt.close()
```

Output/Result

- Accuracy: 0.50

- Precision: 0.46

- Recall: 0.50

- F1 Score: 0.47

References/Credits

- Dataset: student_methods.csv provided by course instructor

- Libraries used: Pandas, Scikit-learn, Seaborn, Matplotlib

- Image/Graph: Generated using Python's Seaborn library