Name: Khushboo Suthar	Roll No.: 22EPCCA027

ASSIGNMENT

POORNIMA COLLEGE OF ENGINEERING, JAIPUR
III B.TECH. (VI Sem.) SEC- D
Code: 6CAI6-02
Subject Name–Machine Learning
(BRANCH: ADVANCE COMPUTING (AI))

Max. Time: 2 hrs. **Max. Marks: 20 Marks**

INSTRUCTIONS: UPLOAD THE SOLUTION ON YOUR GITHUB REPOSITORY and MENTIONED THE URL OF THE REPOSITORY ON TCSION

ASSIGNMENT QUESTION 1: CO3

Fault Prediction Using Supervised Machine Learning

Problem Context

You are an engineer working for a power distribution company responsible for maintaining and ensuring the reliability of the electrical grid. Your task is to develop a system for detecting and classifying electrical faults in the grid. Electrical faults can lead to disruptions, damage equipment, and pose safety hazards. The company is interested in a predictive maintenance system that can identify and classify different types of electrical faults to facilitate timely intervention.

Fault Prediction Dataset: https://www.kaggle.com/code/pythonafroz/fault-prediction-usingdecision-tree-algorithm

S/r No.	Question	Marks
Q1.	Name any 4 libraries required for the implementation of the problem statement using python	2
Ans1.	pandas – for data manipulation and analysis numpy – for numerical computations sklearn – for implementing machine learning algorithms matplotlib – for data visualization	

Q2.	Go through the above Kaggle link and answer the following: About this dataset file: https://www.kaggle.com/code/pythonafroz/fault-prediction-using-decisiontree-algorithm a. Total no. of columns in the dataset: 9 b. Write and count input columns: 8 c. Write and count the output column: 1 (Fault_Severity)	3
Q3.	What is the purpose this library used for the given problem statement: `from sklearn.preprocessing import LabelEncoder`	1
Ans3.	Label Encoder is used to convert categorical variables into numeric format. In the given problem, it helps encode non-numeric labels (like fault type) into numeric values for machine learning algorithms.	
Q4.	What is the purpose this library used for the given problem statement: `from sklearn.model_selection import train_test_split`	1
Ans4.	train_test_split is used to split the dataset into training and testing sets. This allows model training on one subset and testing on another to evaluate performance.	
Q4.	List all the algorithms through which you can able to find Electrical Faults Detection and Classification	3
Ans4.	☐ Decision Tree ☐ Random Forest ☐ Support Vector Machine (SVM) ☐ K-Nearest Neighbors (KNN) ☐ Logistic Regression ☐ Naive Bayes ☐ XGBoost	
Q5.	How to read the Classification_Report generated by several mod els in the given problem statement: https://www.kaggle.com/code/pythonafroz/fault-prediction-using-decisiontree-algorithm	5
Ans5.	The classification_report includes: • **Precision**: True positives / (True positives + False positives) • **Recall**: True positives / (True positives + False negatives) • **F1-Score**: Harmonic mean of precision and recall • **Support**: Number of occurrences of each class in the dataset These metrics help compare the performance of different models.	

Q6.	From the mentioned link: https://www.kaggle.com/code/pythonafroz/faultprediction-using-decision-tree-algorithm Do one sight analysis and figure out which algorithms work well on the given dataset. And on what basis are Model comparisons done over there?	5
Ans6.	From the notebook in the link: • Models compared: Decision Tree, Random Forest, SVM, etc. • Basis of comparison: Accuracy, precision, recall, F1-score • **Best Performing Model**: Random Forest gave higher accuracy and better performance metrics compared to others.	

Name:	Roll No.:

ASSIGNMENT QUESTION 2: CO4

Customer Segmentation using Unsupervised Problem

Context:

You are a data scientist working for a retail company that wants to improve its marketing strategies by better understanding customer behaviour. One approach is to segment customers into distinct groups based on their purchasing habits. This will allow the company to tailor marketing campaigns to specific groups, ultimately increasing sales and customer satisfaction.

Task:

Design and explain the customer segmentation model using any one of the unsupervised algorithms.

1. Data Collection: Obtain a dataset containing customer purchase history, including details such as purchase frequency, amount spent, types of products purchased, etc. You may use publicly available datasets or simulate data for this assignment Include the first 10 rows of the dataset that you are going to consider. 5 marks

2. Data Preprocessing: Clean the dataset and perform necessary preprocessing steps such as normalization, handling missing values, and feature engineering. 5 marks

3. Unsupervised Learning (Clustering): Apply an unsupervised learning algorithm (e.g., Kmeans clustering, hierarchical clustering) to segment customers into distinct groups based on their purchasing behaviour. 5 marks

4. Evaluate the clustering results using appropriate evaluation metrics. 5 marks

Step 1: Dataset (Simulated Sample)

Customer_ID	Purchase_Frequency	Amount_Spent	Product_Types
1	10	5000	3
2	5	3000	2
3	12	8000	4
4	2	1000	1
5	8	4500	3
6	15	9000	5
7	3	2000	1
8	11	7000	4
9	4	2500	2
10	9	6000	3

Step 2: Data Preprocessing

- **Missing Values**: Handle with .fillna() or .dropna()
- **Normalization**: Use MinMaxScaler or StandardScaler
- **Feature Engineering**: Encode Product_Types if necessary

Step 3: Apply Clustering Algorithm (K-Means)

python
Copy code

```
from sklearn.cluster import KMeans
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Simulated Data
data = pd.DataFrame({
   'Purchase_Frequency': [10, 5, 12, 2, 8, 15, 3, 11, 4, 9],
   'Amount_Spent': [5000, 3000, 8000, 1000, 4500, 9000, 2000, 7000, 2500, 6000],
   'Product_Types': [3, 2, 4, 1, 3, 5, 1, 4, 2, 3]
})

# Preprocessing
scaler = StandardScaler()
scaled_data = scaler.fit_transform(data)

# Clustering
kmeans = KMeans(n_clusters=3, random_state=0)
clusters = kmeans.fit_predict(scaled_data)
data['Cluster'] = clusters
```

Step 4: Evaluation

- **Silhouette Score** (for evaluating clustering quality):

python

Copy code

```
from sklearn.metrics import silhouette_score
score = silhouette_score(scaled_data, clusters)
print("Silhouette Score:", score)
```

- **Inertia**: Lower means better cluster fit