FINAL ASSESSMENT DATA SCIENCE TRAINING

Instructions for the Test **90 marks**

- This test consists of three sections A, B and C
- Sec A 10 questions, each worth 6 marks, Sec B 10 questions 1 marks each and Sec C 2 questions 10 marks each.
- Answer each question to the best of your ability, showcasing your understanding of the concepts and practical applications in machine learning.
- Use the provided dataset and Python code examples where necessary to illustrate your solutions.

PART - A

Q 1.1 You are given a dataset with **credit card transactions for fraud detection**. Explain the steps you would take to preprocess the data, including handling missing values, feature scaling. Write Python code to demonstrate these steps.

Q 1.2 You need to **select a model for detecting credit card fraud**. Compare the performance of a Logistic Regression model and a Random Forest model, Decision Tree and SVM. Write Python code to implement this comparison.

Q 1.3 After training a model, you evaluate it using a test set. Write Python code to generate the **confusion matrix and calculate precision, recall, and F1-score**.

Q 1.4 Describe the process of deploying a trained credit card fraud detection model and making real-time predictions on new transaction data. Write Python code to simulate this process with a sample transaction.

Q 1.5 Credit card fraud patterns can evolve over time. Discuss how you would ensure that your machine learning model remains effective over time. Include both technical and operational strategies.

Q 2.1 You have a dataset **laptops.csv** with features mentioned in dataset. Describe and implement the steps you would take to preprocess this data, including handling missing values, encoding categorical variables, and scaling numerical features.

Q 2.2 Explain how you would select the most relevant features for predicting laptop prices. Which methods would you use and why?

Q 2.3 Create and compare at least three different regression models (**e.g., Linear Regression, Decision Tree Regressor, Random Forest Regressor**) and determine which model performs best. Show your code and explain your choice of the best model.

Q 2.4 After selecting and training the best model, evaluate its performance on a test set. Show your code and explain the evaluation metrics you used.

Q 2.5 Write a Python function that takes in features such as **'Brand', 'Processor', 'RAM', 'Storage', and 'GPU'**, and predicts the laptop price using the trained model. Assume the model and required libraries are already imported.

PART – B

Q1. Explain the concept of a JOIN in SQL. What are the different types of JOINs and when would you use each?

Q2. Define a constraint in SQL. Provide examples of commonly used constraints.

Q3. What are the different types of machine learning? Provide brief descriptions and examples of each type.

Q4. How would you handle missing or corrupted data in a dataset before applying a machine learning algorithm? Discuss potential techniques and their implications.

Q5. Define precision and recall. How are these metrics used to evaluate the performance of a binary classification model?

Q6. Describe the difference between a primary key and a foreign key in SQL. When would you use one over the other?

Q7. Explain the concept of indexing in SQL databases. What are its benefits and when would you consider creating an index?

Q8. Name two parameters commonly found in a confusion matrix. Explain their significance in evaluating classification models.

Q9. What does 'naive' refer to in the Naive Bayes classifier? How does this assumption simplify the model's calculations?

Q10. Describe the difference between a primary key and a foreign key in SQL. When would you use one over the other?

PART – C

Q1. Error Finding and Correction

```python
# Error Correction:- AM

import pandas as pd
from sklearn.model_selection import test_train_split
from sklearn.scaler import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.matrix import accuracy_score_classification


data = pd.Read_Csv('data.csv')


X = data.drop('target', axis=[0])
y = data[['target']]


X_train, y_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)


scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)


model = RandomForestClassifier(n_estimators=100, random_state=42, max_depth=5, min_samples_split=2)
model.fit(X_train_scaled, y_train)


y_pred = model.predict(X_test_scaled)

accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy of the model: {accuracy:.2f}')
```

Q2. Error Finding and Correction

```sql
-- Incorrect SQL Query
SELECT
    product_category,
    SUM(sales_amount) AS total_sales
FROM
    sales_data
GROUP BY
    category
ORDER BY
    total_sales DESC
```

```python
from sklearn import svm
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score


X, y = make_classification(n_samples=100, n_features=20, random_state=42)


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


clf = svm.SVC(C=1.0, kernel='linear', random_state=42, gamma='scale')


clf.fit(X_train, y_train)


y_pred = clf.predict(X_test)


accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy of the model: {accuracy:.2f}')
```