EXPERIMENT 1

Name:-Ujwal Sahu

Roll no-34

Title:- Demonstrate handling of duplicate data using python in machine learning.

Tool:- Jupyter Notebook .

Theory:- Handling duplicate data is an important step in the data preprocessing phase of machine learning. Duplicates can skew the results of your model, leading to overfitting or biased predictions. Below, I'll demonstrate how to handle duplicate data in a machine learning context using Python and the **pandas** library, along with a simple machine learning model using `scikit-learn.`

Code:-

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score


# Sample data with duplicates

data = {

   'Feature1': [1, 2, 2, 3, 4, 4, 5, 6, 6, 7],

   'Feature2': [10, 20, 20, 30, 40, 40, 50, 60, 60, 70],

   'Target': [0, 1, 1, 0, 1, 1, 0, 1, 1, 0]

}


df = pd.DataFrame(data)

print("Original DataFrame:")

print(df)


# Identify duplicate rows
```

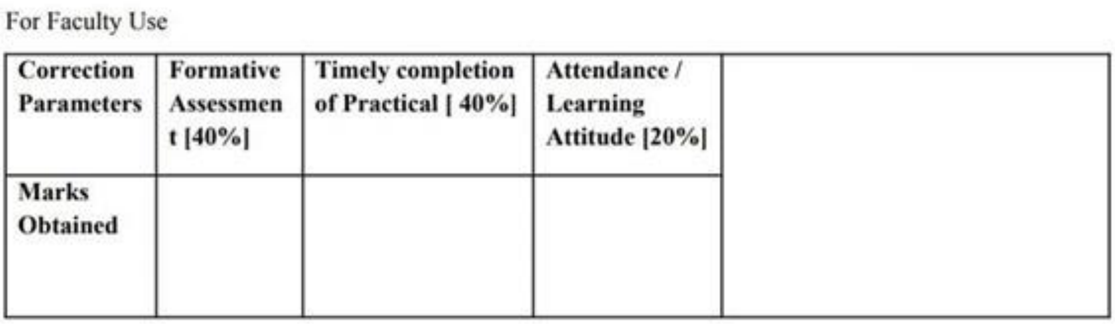
```
duplicates = df.duplicated()
print("\nDuplicate Rows:")
print(df[duplicates])

# Remove duplicate rows
df_no_duplicates = df.drop_duplicates()
print("\nDataFrame after removing duplicates:")
print(df_no_duplicates)

# Split the data into features and target
X = df_no_duplicates[['Feature1', 'Feature2']]
y = df_no_duplicates['Target']

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("\nTraining Features:")
print(X_train)
print("\nTesting Features:")
print(X_test)

# Initialize the model
model = RandomForestClassifier(random_state=42)

# Train the model
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
```

```
accuracy = accuracy_score(y_test, y_pred)

print("\nModel Accuracy:", accuracy)
```

Output:-

```
Original DataFrame:
   Feature1  Feature2  Target
0         1        10       0
1         2        20       1
2         2        20       1
3         3        30       0
4         4        40       1
5         4        40       1
6         5        50       0
7         6        60       1
8         6        60       1
9         7        70       0

Duplicate Rows:
   Feature1  Feature2  Target
2         2        20       1
5         4        40       1
8         6        60       1

DataFrame after removing duplicates:
   Feature1  Feature2  Target
0         1        10       0
1         2        20       1
3         3        30       0
4         4        40       1
6         5        50       0
7         6        60       1
9         7        70       0

Training Features:
   Feature1  Feature2
7         6        60
3         3        30
6         5        50
4         4        40
9         7        70

Testing Features:
   Feature1  Feature2
0         1        10
1         2        20
```

Conclusion:- In this demonstration, we highlighted the importance of handling duplicate data in machine learning. By using the **pandas** library, we identified and removed duplicates from a sample dataset, ensuring data quality before model training. We then trained a Random Forest Classifier and evaluated its accuracy, emphasizing that clean data is crucial for reliable model performance. Proper data preprocessing, including duplicate handling, significantly enhances the effectiveness of machine learning models.



For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				