ASSIGNMENT - 1

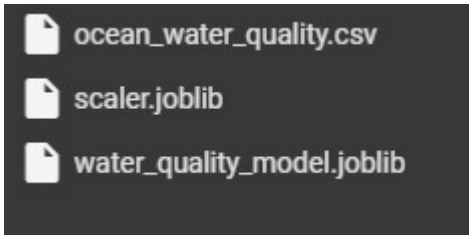
United Nations has defined 17 Sustainable Development Goals (UN-SDGs) for a collective bright future

TOPIC - SDG 14 Life Below Water

predicting ocean water quality based on environmental and chemical indicators. This can help identify areas at risk of pollution, thereby aiding in the conservation and sustainable use of marine resources.

dataset, ocean_water_quality.csv, with columns such as temperature, ph, salinity, nitrate, phosphate, dissolved_oxygen, and a target variable water_quality indicating whether the water quality is Good (1) or Poor (0).

CSV DATA -




```
temperature,ph,nitrate,dissolved_oxygen,salinity,water_quality
20.5,7.2,0.3,8.0,34.5,1
21.3,7.4,0.25,7.8,35.1,1
19.8,6.8,0.5,7.2,33.9,0
22.0,7.5,0.2,8.1,35.3,1
18.7,6.9,0.55,7.0,33.6,0
21.5,7.3,0.28,8.0,34.8,1
20.1,6.7,0.6,7.3,33.2,0
22.3,7.6,0.15,8.2,35.5,1
19.3,6.6,0.65,7.1,32.9,0
21.8,7.4,0.3,7.9,34.9,1
```

Define Methodology and Objectives

- **Objective**: Build a classification model to predict water quality based on chemical and environmental indicators.
- **Methodology**: This is a binary classification problem, so we'll use models like Logistic Regression and Random Forest.

Data Preprocessing

[3]
```python
# Import necessary libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix
from imblearn.over_sampling import RandomOverSampler
import joblib

# Step 1: Load Dataset
data = pd.read_csv('/content/ocean_water_quality.csv')
print("First 5 rows of the dataset:")
print(data.head())

# Step 2: Preprocess the Data
# Separate features and target variable
X = data.drop(columns=['water_quality'])  # Assuming 'water_quality' is the target column
y = data['water_quality']

# Handle class imbalance using RandomOverSampler
ros = RandomOverSampler(random_state=42)
X_res, y_res = ros.fit_resample(X, y)

# Split data into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X_res, y_res, test_size=0.2, random_state=42)

# Scale the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Save the scaler for future use in the Flask API
joblib.dump(scaler, 'scaler.joblib')
```

```
First 5 rows of the dataset:
   temperature   ph  nitrate  dissolved_oxygen  salinity  water_quality
0         20.5  7.2     0.30               8.0      34.5              1
1         21.3  7.4     0.25               7.8      35.1              1
2         19.8  6.8     0.50               7.2      33.9              0
3         22.0  7.5     0.20               8.1      35.3              1
4         18.7  6.9     0.55               7.0      33.6              0
['scaler.joblib']
```

[4]
```python
# Step 3: Train the Model
rf_model = RandomForestClassifier(random_state=42)
rf_model.fit(X_train, y_train)

# Cross-validation to check model reliability
cv_scores = cross_val_score(rf_model, X_train, y_train, cv=5, scoring='accuracy
print(f"Cross-Validation Accuracy Scores: {cv_scores}")
print(f"Mean Cross-Validation Accuracy: {cv_scores.mean()}")

# Save the trained model for Flask deployment
joblib.dump(rf_model, 'water_quality_model.joblib')
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:776:
  warnings.warn(
Cross-Validation Accuracy Scores: [1. 1. 1. 1. 1.]
Mean Cross-Validation Accuracy: 1.0
['water_quality_model.joblib']
```

[5]
```python
# Step 4: Model Evaluation
# Make predictions on the test set
predictions = rf_model.predict(X_test)

# Calculate evaluation metrics
accuracy = accuracy_score(y_test, predictions)
precision = precision_score(y_test, predictions)
recall = recall_score(y_test, predictions)
f1 = f1_score(y_test, predictions)
conf_matrix = confusion_matrix(y_test, predictions)

print("Model Evaluation Metrics:")
print(f"Accuracy: {accuracy}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1 Score: {f1}")
print(f"Confusion Matrix:\n{conf_matrix}")
```

```
First 5 rows of the dataset:
   temperature   ph  nitrate  dissolved_oxygen  salinity  water_quality
0         20.5  7.2     0.30               8.0      34.5              1
1         21.3  7.4     0.25               7.8      35.1              1
2         19.8  6.8     0.50               7.2      33.9              0
3         22.0  7.5     0.20               8.1      35.3              1
4         18.7  6.9     0.55               7.0      33.6              0
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:776:
  warnings.warn(
Cross-Validation Accuracy Scores: [1. 1. 1. 1. 1.]
Mean Cross-Validation Accuracy: 1.0
Model Evaluation Metrics:
Accuracy: 1.0
Precision: 1.0
Recall: 1.0
F1 Score: 1.0
Confusion Matrix:
[[1 0]
 [0 2]]
```

NAME – CHAITANYA GAUR
BATCH – 11
ROL NO – R2142230331
SAP ID – 500122644