Water Quality Prediction Results

import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

from sklearn.impute import SimpleImputer

from sklearn.preprocessing import StandardScaler

from sklearn.linear\_model import LogisticRegression

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import classification\_report, accuracy\_score

# Load dataset

df = pd.read\_csv("water\_potability.csv")

# Handle missing values

imputer = SimpleImputer(strategy="mean")

df\_imputed = pd.DataFrame(imputer.fit\_transform(df), columns=df.columns)

# Features and target

X = df\_imputed.drop("Potability", axis=1)

y = df\_imputed["Potability"]

# Train-test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

X, y, test\_size=0.2, random\_state=42, stratify=y

)

# Feature scaling

scaler = StandardScaler()

X\_train\_scaled = scaler.fit\_transform(X\_train)

X\_test\_scaled = scaler.transform(X\_test)

# Logistic Regression

log\_reg = LogisticRegression(max\_iter=1000)

log\_reg.fit(X\_train\_scaled, y\_train)

y\_pred\_log = log\_reg.predict(X\_test\_scaled)

# Random Forest

rf = RandomForestClassifier(n\_estimators=100, random\_state=42)

rf.fit(X\_train, y\_train)

y\_pred\_rf = rf.predict(X\_test)

# Results

print("Logistic Regression Accuracy:", accuracy\_score(y\_test, y\_pred\_log))

print("Random Forest Accuracy:", accuracy\_score(y\_test, y\_pred\_rf))

print("\n--- Logistic Regression Report ---")

print(classification\_report(y\_test, y\_pred\_log))

print("\n--- Random Forest Report ---")

print(classification\_report(y\_test, y\_pred\_rf))

Logistic Regression Accuracy: 0.61

--- Logistic Regression Report --- precision recall f1-score support

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0.0 | 0.61 | 1.00 | 0.76 | 400 |
| 1.0 | 0.00 | 0.00 | 0.00 | 256 |
| Accuracy |  |  | 0.61 | 656 |
| Random Forest Accuracy: 0.66 | | |  |  |
| --- Random Forest Report --- precision recall f1-score | | | support |  |
| 0.0 0.67 0.88 | | | 0.76 | 400 |
| 1.0 0.62 0.31 | | | 0.41 | 256 |
| Accuracy | | | 0.66 | 656 |