Day 2 Problem Set: Machine Learning Classification Projects

Instructions:

Please choose one of the following real-world problems. Build and evaluate a classification model pipeline using at least three classifiers introduced in class (e.g., Logistic Regression, Decision Tree, Random Forest, Gradient Boost, SVM, KNN).

Your final notebook should include:

- Data exploration and preprocessing

- Model training and performance evaluation

- Metric interpretation (precision, recall, F1, ROC-AUC)

- An output for comparison of the models

- Hyperparameter tuning with GridSearchCV

1. Heart Disease Prediction

Objective: Predict whether a patient has heart disease based on clinical features like chest pain type, cholesterol, and resting ECG results.

Dataset: UCI Heart Disease Dataset

Challenge: Understand feature contributions in medical prediction.

2. Water Quality Classification

Objective: Classify water samples as safe or unsafe to drink using chemical attributes like pH, turbidity, and sulfate levels.

Dataset: Kaggle - Water Quality Dataset

Challenge: Class imbalance and feature normalization.

3. Adult Income Classification

Objective: Predict whether an individual earns more than $50K per year based on census attributes such as age, education, marital status, and occupation.

Dataset: UCI Adult Income Dataset

Challenge: Working with mixed data types (categorical + numeric) and class imbalance.

4. Plant Species Identification

Objective: Classify plant species based on morphological features of leaves such as shape, texture,

and length.

Dataset: Leaf Dataset (UCI or Kaggle)

Challenge: High-dimensional, continuous feature space.

5. Customer Churn Prediction

Objective: Predict whether a telecom customer is likely to cancel their service subscription using

usage patterns, contract types, and payment methods.

Dataset: Telco Customer Churn Dataset (Kaggle)

Challenge: Feature engineering and interpreting categorical features.

Evaluation Checklist

- Preprocessing: Clean, encode, and scale data

- Models: Apply at least 3 classifiers

- Evaluation: Use confusion matrix, precision, recall, F1-score, ROC-AUC

- Cross-Validation: Apply k-fold CV (e.g., k=5)

- Model Comparison: Provide a written summary (2-3 bullets or a table)

- Hyperparameter tuning with GridSearchCV