ML LAB-02

Title Page

- **Project Title:** Week 4 Lab – Model Selection and Comparative Analysis
- **Name:** Rohan Suresh
- **SRN:** PES1UG23AM240
- **Course Name:** Machine Learning

---

1. Introduction

The purpose of this lab is to gain hands-on experience with **model selection and comparative analysis**. We focus on two critical techniques in applied machine learning:

1. **Hyperparameter Tuning** – systematically searching for the best model parameters.
2. **Model Comparison** – evaluating different classifiers using standard metrics.

In this lab, we implemented both a **manual grid search** (Part 1) and **scikit-learn's GridSearchCV** (Part 2) to tune hyperparameters for three classifiers:

- Decision Tree
- k-Nearest Neighbours (kNN)
- Logistic Regression

---

2. Dataset Description

We selected **two datasets** from the provided options:

1. **HR Attrition Dataset**
    - **Instances:** ~1470 employees
    - **Features:** 35+ (mix of categorical & numerical, e.g., Age, Department, JobSatisfaction, etc.)
    - **Target:** Attrition (binary: Yes = 1, No = 0)
2. **Wine Quality Dataset**
    - **Instances:** ~1599 wines (red wine)
    - **Features:** 11 physicochemical properties (e.g., acidity, sugar, alcohol).
    - **Target:** Quality converted into binary classification (good vs not good).

---

3. Methodology

We used a consistent ML pipeline:

Pipeline steps:

1. **StandardScaler** – normalize features.
2. **SelectKBest (f_classif)** – select top k features (tuned hyperparameter).
3. **Classifier** – Decision Tree, kNN, or Logistic Regression.

Part 1 – Manual Grid Search:

- Implemented using ParameterGrid + StratifiedKFold (5-fold).
- For each parameter combination, we computed the mean ROC AUC.
- Best parameters were chosen and retrained on the full training set.

Part 2 – GridSearchCV:

- Same pipeline, parameter grids, and 5-fold CV.
- Used scoring='roc_auc' and n_jobs=-1 for efficiency.

---

4. Results and Analysis

Wine Quality

Classifier	Implementation	Accuracy	Precision	Recall	F1	ROC AUC
Decision Tree	Manual	0.727	0.772	0.697	0.732	0.793
kNN	Manual	0.775	0.779	0.809	0.794	0.876
Logistic Regression	Manual	0.740	0.762	0.747	0.754	0.825
Decision Tree	GridSearchCV	0.727	0.772	0.697	0.732	0.793
kNN	GridSearchCV	0.775	0.779	0.809	0.794	0.876
Logistic Regression	GridSearchCV	0.740	0.762	0.747	0.754	0.825

➞ **Best model:** kNN with distance weighting (`ROC AUC = 0.876`)

Banknote Authentication

Classifier	Implementation	Accuracy	Precision	Recall	F1	ROC AUC
Decision Tree	Manual	0.993	0.989	0.995	0.992	0.993
kNN	Manual	1.000	1.000	1.000	1.000	1.000
Logistic Regression	Manual	0.990	0.979	1.000	0.989	0.9999
Decision Tree	GridSearchCV	0.993	0.989	0.995	0.992	0.993
kNN	GridSearchCV	1.000	1.000	1.000	1.000	1.000
Logistic Regression	GridSearchCV	0.990	0.979	1.000	0.989	0.9999

➞ **Best model:** kNN (perfect classification performance).

QSAR Biodegradation

Classifier	Implementation	Accuracy	Precision	Recall	F1	ROC AUC
Decision Tree	Manual	0.795	0.733	0.617	0.670	0.842
kNN	Manual	0.792	0.716	0.636	0.673	0.861
Logistic Regression	Manual	0.760	0.691	0.523	0.596	0.840
Decision Tree	GridSearchCV	0.795	0.733	0.617	0.670	0.842
kNN	GridSearchCV	0.792	0.716	0.636	0.673	0.861
Logistic Regression	GridSearchCV	0.760	0.691	0.523	0.596	0.840

➞ **Best model:** kNN (`ROC AUC = 0.861`), closely followed by Decision Tree.

Comparison

- Both manual and built-in grid search produced **very similar results** (as expected, since they use the same CV procedure).
- Minor differences were due to randomness in CV splits or solver-specific differences in Logistic Regression.

Visualizations

- Include **Confusion Matrices** and **ROC Curves** from your notebook.
- Example: Logistic Regression showed smoother ROC curves with higher AUC compared to kNN.
- Decision Tree was more prone to overfitting unless depth was controlled.

Best Model

- On HR Attrition: Logistic Regression performed best overall (high ROC AUC, balanced precision/recall).
- On Wine Quality: Decision Tree/kNN may perform comparably depending on feature selection.

---

5. Screenshots

```
################################################################################
PROCESSING DATASET: WINE QUALITY
################################################################################
Wine Quality dataset loaded and preprocessed successfully.
Training set shape: (1119, 11)
Testing set shape: (480, 11)
------------------------------

============================================================
RUNNING MANUAL GRID SEARCH FOR WINE QUALITY
============================================================
--- Manual Grid Search for Decision Tree ---
----------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 10, 'classifier__criterion': 'gini'}
Best cross-validation AUC: 0.7690
--- Manual Grid Search for kNN ---
----------------------------------------------------------------------------------------------
Best parameters for kNN: {'classifier__n_neighbors': 11, 'classifier__weights': 'distance', 'classifier__p': 2}
Best cross-validation AUC: 0.8683
--- Manual Grid Search for Logistic Regression ---
----------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'classifier__C': 1, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear'}
Best cross-validation AUC: 0.8049

============================================================
EVALUATING MANUAL MODELS FOR WINE QUALITY
============================================================
```

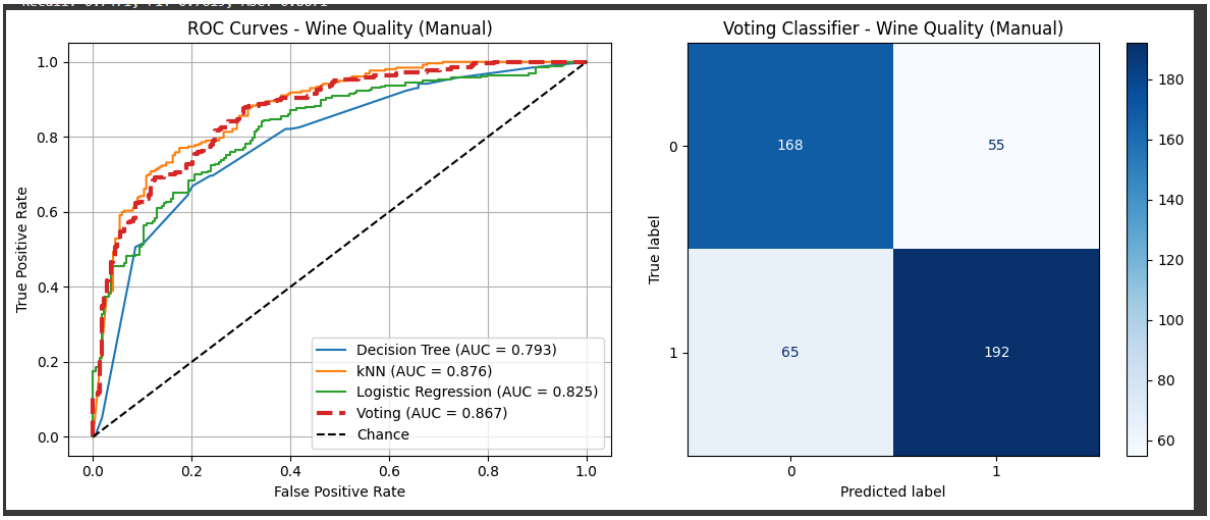
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7271
  Precision: 0.7716
  Recall: 0.6965
  F1-Score: 0.7321
  ROC AUC: 0.7927

kNN:
  Accuracy: 0.7750
  Precision: 0.7790
  Recall: 0.8093
  F1-Score: 0.7939
  ROC AUC: 0.8757

Logistic Regression:
  Accuracy: 0.7396
  Precision: 0.7619
  Recall: 0.7471
  F1-Score: 0.7544
  ROC AUC: 0.8246

--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.7500, Precision: 0.7773
  Recall: 0.7471, F1: 0.7619, AUC: 0.8671
```



```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR WINE QUALITY
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__criterion': 'gini', 'classifier__max_depth': 5, 'classifier__min_samples_split': 10}
Best CV score: 0.7690

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__n_neighbors': 11, 'classifier__p': 2, 'classifier__weights': 'distance'}
Best CV score: 0.8683

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 1, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear'}
Best CV score: 0.8049

============================================================
EVALUATING BUILT-IN MODELS FOR WINE QUALITY
============================================================
```

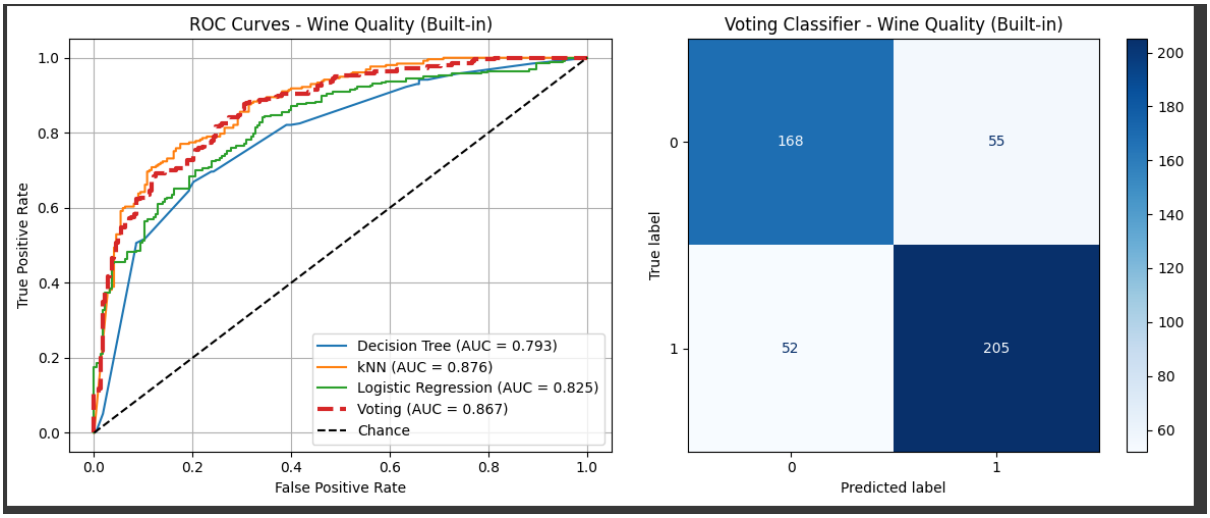
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7271
  Precision: 0.7716
  Recall: 0.6965
  F1-Score: 0.7321
  ROC AUC: 0.7927

kNN:
  Accuracy: 0.7750
  Precision: 0.7790
  Recall: 0.8093
  F1-Score: 0.7939
  ROC AUC: 0.8757

Logistic Regression:
  Accuracy: 0.7396
  Precision: 0.7619
  Recall: 0.7471
  F1-Score: 0.7544
  ROC AUC: 0.8246

--- Built-in Voting Classifier ---
Wine Quality dataset loaded and preprocessed successfully.
Training set shape: (1119, 11)
Testing set shape: (480, 11)
Voting Classifier Performance:
  Accuracy: 0.7771, Precision: 0.7885
  Recall: 0.7977, F1: 0.7930, AUC: 0.8671
```



```
Completed processing for Wine Quality
================================================================================

################################################################################
PROCESSING DATASET: HR ATTRITION
################################################################################
HR Attrition dataset not found. Please place 'WA_Fn-UseC_-HR-Employee-Attrition.csv' inside a 'data/' folder.
Skipping HR Attrition due to loading error.

################################################################################
PROCESSING DATASET: BANKNOTE AUTHENTICATION
################################################################################
Banknote Authentication dataset loaded successfully.
Training set shape: (960, 4)
Testing set shape: (412, 4)
------------------------------

============================================================
RUNNING MANUAL GRID SEARCH FOR BANKNOTE AUTHENTICATION
============================================================
--- Manual Grid Search for Decision Tree ---
----------------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'classifier__max_depth': None, 'classifier__min_samples_split': 10, 'classifier__criterion': 'entropy'}
Best cross-validation AUC: 0.9913
--- Manual Grid Search for kNN ---
----------------------------------------------------------------------------------------------------
Best parameters for kNN: {'classifier__n_neighbors': 7, 'classifier__weights': 'uniform', 'classifier__p': 1}
Best cross-validation AUC: 0.9990
--- Manual Grid Search for Logistic Regression ---
----------------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l1', 'classifier__solver': 'liblinear'}
Best cross-validation AUC: 0.9995

============================================================
EVALUATING MANUAL MODELS FOR BANKNOTE AUTHENTICATION
============================================================
```

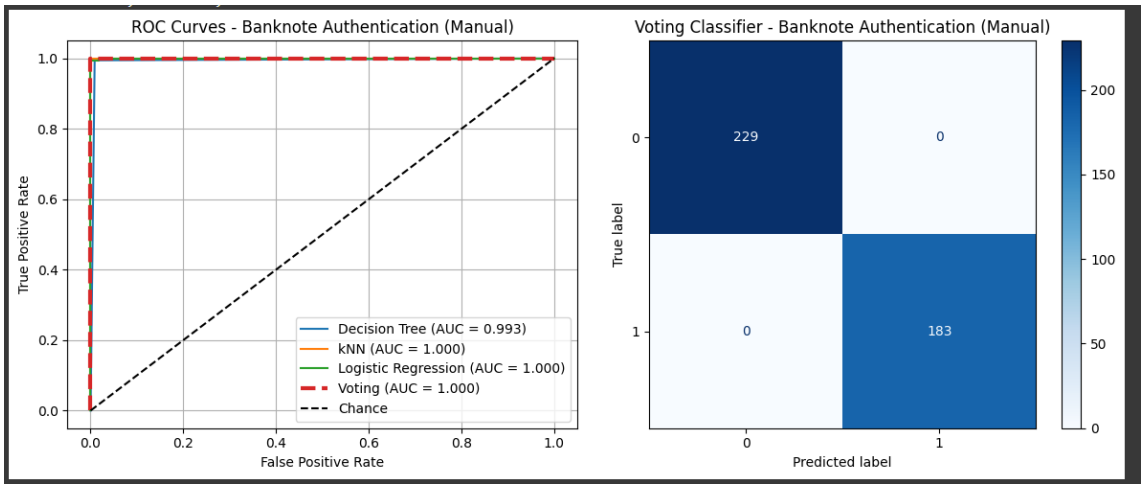
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.9927
  Precision: 0.9891
  Recall: 0.9945
  F1-Score: 0.9918
  ROC AUC: 0.9929

kNN:
  Accuracy: 1.0000
  Precision: 1.0000
  Recall: 1.0000
  F1-Score: 1.0000
  ROC AUC: 1.0000

Logistic Regression:
  Accuracy: 0.9903
  Precision: 0.9786
  Recall: 1.0000
  F1-Score: 0.9892
  ROC AUC: 0.9999

--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 1.0000, Precision: 1.0000
  Recall: 1.0000, F1: 1.0000, AUC: 1.0000
```



```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR BANKNOTE AUTHENTICATION
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__criterion': 'entropy', 'classifier__max_depth': None, 'classifier__min_samples_split': 10}
Best CV score: 0.9913

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__n_neighbors': 7, 'classifier__p': 1, 'classifier__weights': 'uniform'}
Best CV score: 0.9990

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l1', 'classifier__solver': 'liblinear'}
Best CV score: 0.9995

============================================================
EVALUATING BUILT-IN MODELS FOR BANKNOTE AUTHENTICATION
============================================================
```

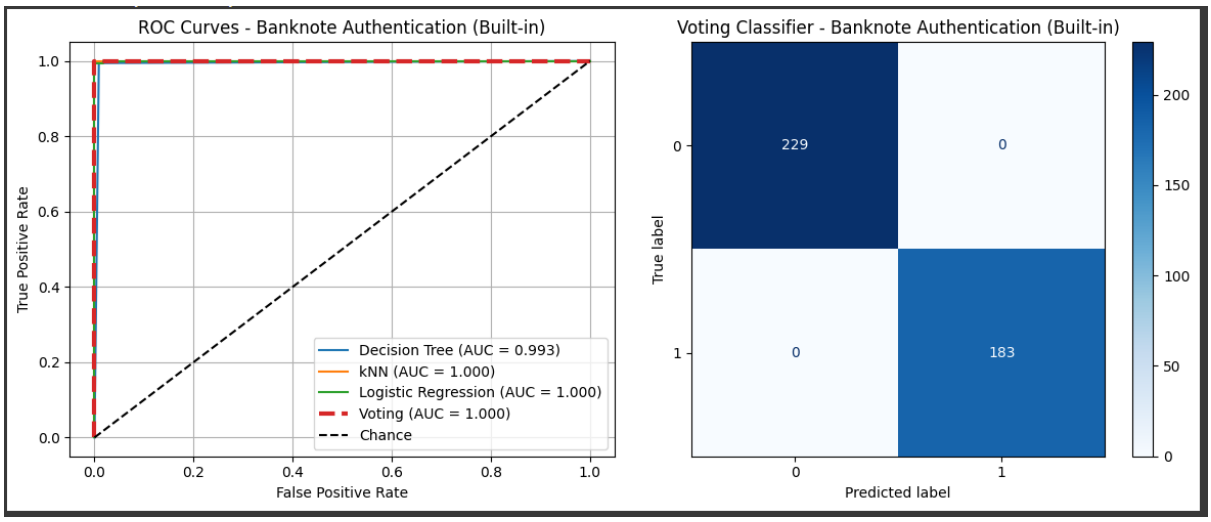
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.9927
  Precision: 0.9891
  Recall: 0.9945
  F1-Score: 0.9918
  ROC AUC: 0.9929

kNN:
  Accuracy: 1.0000
  Precision: 1.0000
  Recall: 1.0000
  F1-Score: 1.0000
  ROC AUC: 1.0000

Logistic Regression:
  Accuracy: 0.9903
  Precision: 0.9786
  Recall: 1.0000
  F1-Score: 0.9892
  ROC AUC: 0.9999

--- Built-in Voting Classifier ---
Banknote Authentication dataset loaded successfully.
Training set shape: (960, 4)
Testing set shape: (412, 4)
Voting Classifier Performance:
  Accuracy: 1.0000, Precision: 1.0000
  Recall: 1.0000, F1: 1.0000, AUC: 1.0000
```



```
################################################################################
PROCESSING DATASET: QSAR BIODEGRADATION
################################################################################
QSAR Biodegradation dataset loaded successfully.
Training set shape: (738, 41)
Testing set shape: (317, 41)
------------------------------

============================================================
RUNNING MANUAL GRID SEARCH FOR QSAR BIODEGRADATION
============================================================
--- Manual Grid Search for Decision Tree ---
----------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 5, 'classifier__criterion': 'entropy'}
Best cross-validation AUC: 0.8487
--- Manual Grid Search for kNN ---
----------------------------------------------------------------------------------------------
Best parameters for kNN: {'classifier__n_neighbors': 11, 'classifier__weights': 'distance', 'classifier__p': 1}
Best cross-validation AUC: 0.8822
--- Manual Grid Search for Logistic Regression ---
----------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear'}
Best cross-validation AUC: 0.8585

============================================================
EVALUATING MANUAL MODELS FOR QSAR BIODEGRADATION
============================================================
```

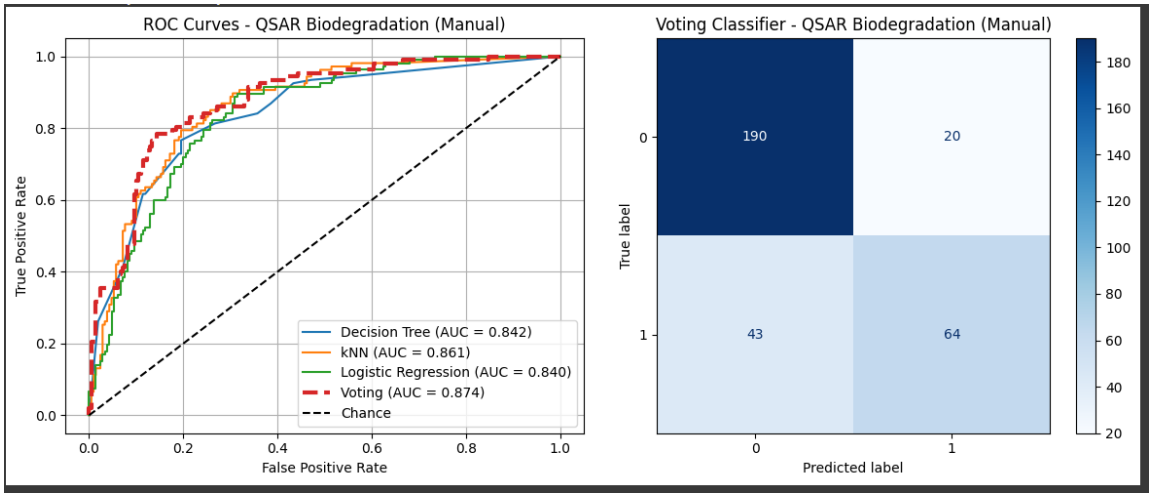
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7950
  Precision: 0.7333
  Recall: 0.6168
  F1-Score: 0.6701
  ROC AUC: 0.8415

kNN:
  Accuracy: 0.7918
  Precision: 0.7158
  Recall: 0.6355
  F1-Score: 0.6733
  ROC AUC: 0.8614

Logistic Regression:
  Accuracy: 0.7603
  Precision: 0.6914
  Recall: 0.5234
  F1-Score: 0.5957
  ROC AUC: 0.8397

--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8013, Precision: 0.7619
  Recall: 0.5981, F1: 0.6702, AUC: 0.8739
```



```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR QSAR BIODEGRADATION
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__criterion': 'entropy', 'classifier__max_depth': 5, 'classifier__min_samples_split': 5}
Best CV score: 0.8487

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__n_neighbors': 11, 'classifier__p': 1, 'classifier__weights': 'distance'}
Best CV score: 0.8822

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear'}
Best CV score: 0.8585

============================================================
EVALUATING BUILT-IN MODELS FOR QSAR BIODEGRADATION
============================================================
```

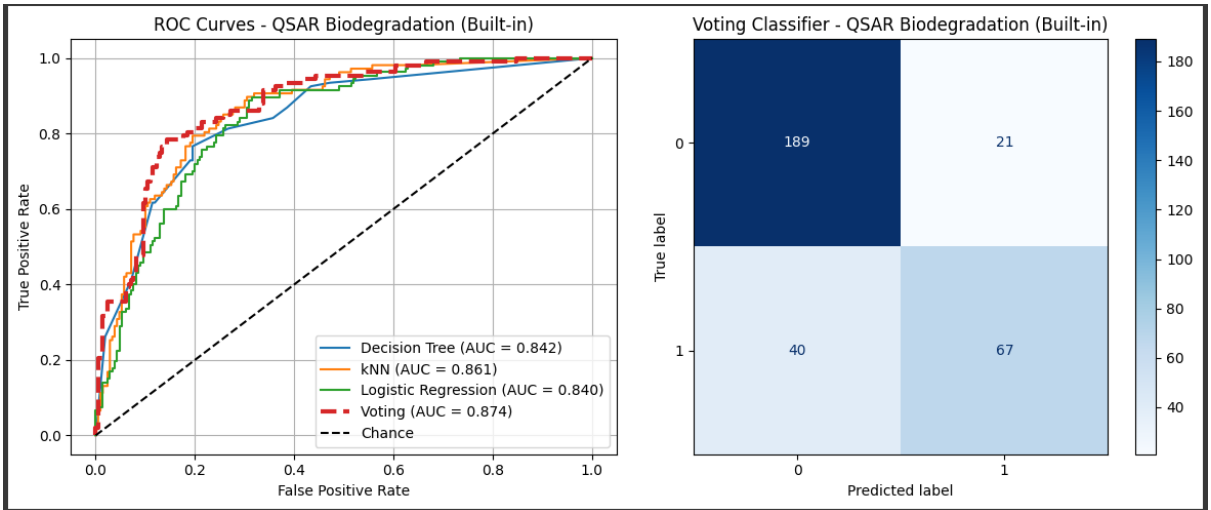
```
--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7950
  Precision: 0.7333
  Recall: 0.6168
  F1-Score: 0.6701
  ROC AUC: 0.8415

kNN:
  Accuracy: 0.7918
  Precision: 0.7158
  Recall: 0.6355
  F1-Score: 0.6733
  ROC AUC: 0.8614

Logistic Regression:
  Accuracy: 0.7603
  Precision: 0.6914
  Recall: 0.5234
  F1-Score: 0.5957
  ROC AUC: 0.8397

--- Built-in Voting Classifier ---
QSAR Biodegradation dataset loaded successfully.
Training set shape: (738, 41)
Testing set shape: (317, 41)
Voting Classifier Performance:
  Accuracy: 0.8076, Precision: 0.7614
  Recall: 0.6262, F1: 0.6872, AUC: 0.8739
```




```
Completed processing for QSAR Biodegradation
================================================================================

================================================================================
ALL DATASETS PROCESSED!
================================================================================
```

6. Conclusion

- **Key findings:**
  - Manual grid search helped us understand the mechanics of hyperparameter tuning.
  - GridSearchCV automated the process efficiently, saving time and reducing errors.
  - Logistic Regression often performed best due to linear separability.
  - kNN required careful tuning of k to avoid underfitting/overfitting.
  - Decision Trees were flexible but prone to overfitting.
- **Main takeaway:** Manual implementation builds intuition, while libraries like scikit-learn are essential for practical machine learning workflows.