ML LAB4

Model Selection and Comparative Analysis

NAME:NAVYASHREE.SP

SRN:PES2UG23CS374

SUBMISSION DATE:31/08/2025

1. Introduction

In this we explored hyperparameter tuning and model comparison using multiple datasets. The main objective was to evaluate different classifiers (Decision Tree, k-Nearest Neighbors, Logistic Regression) and optimize their performance through manual hyperparameter tuning as well as the built-in GridSearchCV from scikit-learn. We implemented:

- Manual grid search with different hyperparameter combinations.
- Automated hyperparameter tuning using scikit-learn's pipeline and grid search.
- Model comparison using performance metrics and voting classifier evaluation.

This allowed us to understand the trade-offs between manual implementation and using a library-based solution.

2. Dataset Description

(a) Wine Quality Dataset

- Instances: 1599 (red wine samples).
- Features: 11 chemical properties (pH, alcohol, citric acid, etc.).
- Target Variable: Wine quality score (categorical, converted to binary: good/bad).

(b) HR Attrition Dataset (IBM HR Analytics)

- Instances (after preprocessing): 1,470 samples split into Training (1029) and Testing (441).
- Features: 46 features
- Target Variable: Attrition (Yes/No), indicating whether an employee has left the company.

- Goal: Predict employee turnover based on personal and professional characteristics.

(c) Banknote Authentication Dataset

- Instances: 1372.
- Features: 4 features extracted from banknote images (variance, skewness, curtosis, entropy).
- Target Variable: 0 (genuine) or 1 (forged).

(d) QSAR Biodegradation Dataset

- Instances: 1055.
- Features: 41 molecular descriptors.
- Target Variable: 0 (non-biodegradable) or 1 (biodegradable).

3. Methodology

- Key Concepts:
    - Hyperparameter Tuning: Adjusting parameters that control model behavior (e.g., max_depth in Decision Trees, k in kNN, penalty in Logistic Regression).
    - Grid Search: Exhaustively searching across parameter combinations to find the best set.
    - K-Fold Cross Validation: Data is split into *k* folds; training/testing is rotated to ensure robust evaluation.
- ML Pipeline:
    - Preprocessing: StandardScaler for normalization.
    - Feature Selection: SelectKBest for dimensionality reduction.
    - Classifier: Decision Tree, kNN, or Logistic Regression.
- Implementation:
    - Part 1 (Manual): Iterated manually through parameter combinations and evaluated using performance metrics.
    - Part 2 (Scikit-learn): Used Pipeline + GridSearchCV for automated hyperparameter tuning.

4. Results and Analysis

(a) Performance Tables

For each dataset, summarize metrics (Accuracy, Precision, Recall, F1-score, ROC AUC)

Wine quality dataset

Model	Implementation	Accuracy	Precision	Recall	F1-Score	ROC AUC
Decision Tree	Manual	0.7271	0.7716	0.6965	0.7321	0.8025
Decision Tree	GridSearchCV	0.7271	0.7716	0.6965	0.7321	0.8025
kNN	Manual	0.7812	0.7836	0.8171	0.8000	0.8589
kNN	GridSearchCV	0.7812	0.7836	0.8171	0.8000	0.8589
Logistic Regression	Manual	0.7333	0.7510	0.7510	0.7510	0.8199
Logistic Regression	GridSearchCV	0.7333	0.7510	0.7510	0.7510	0.8199
Voting Classifier	Manual	0.7375	0.7610	0.7432	0.7520	0.8591
Voting Classifier	GridSearchCV	0.7646	0.7769	0.7860	0.7814	0.8591

Analysis:

- Both implementations produced identical results for individual classifiers.
- The Voting Classifier improved performance slightly, especially in GridSearchCV (higher Recall and F1).
- kNN was the strongest standalone model (highest Recall and AUC).

HR Attrition Dataset

Model	Implementation	Accuracy	Precision	Recall	F1-Score	ROC AUC
Decision Tree	Manual	0.8118	0.3696	0.2394	0.2906	0.6844
Decision Tree	GridSearchCV	0.8118	0.3696	0.2394	0.2906	0.6844
kNN	Manual	0.8186	0.3784	0.1972	0.2593	0.7236
kNN	GridSearchCV	0.8186	0.3784	0.1972	0.2593	0.7236
Logistic Regression	Manual	0.8481	0.625	0.1408	0.2299	0.7544
Logistic Regression	GridSearchCV	0.8481	0.625	0.1408	0.2299	0.7544
Voting Classifier	Manual	0.8345	0.4643	0.1831	0.2626	0.744
Voting Classifier	GridSearchCV	0.8277	0.4194	0.1831	0.2549	0.744

Analysis:

- Logistic Regression had the highest ROC AUC (0.7544) despite low Recall.
- All models struggled with Recall, indicating difficulty in identifying employees who left.
- Voting Classifier balanced Precision and Recall better but did not outperform Logistic Regression in AUC.

Banknote Authentication Dataset

Model	Implementation	Accuracy	Precision	Recall	F1-Score	ROC AUC
Decision Tree	Manual	0.9854	0.9733	0.9945	0.9838	0.9847
Decision Tree	GridSearchCV	0.9854	0.9733	0.9945	0.9838	0.9847
kNN	Manual	1	1	1	1	1
kNN	GridSearchCV	1	1	1	1	1

Model	Implementation	Accuracy	Precision	Recall	F1-Score	ROC AUC
Logistic Regression	Manual	0.9903	0.9786	1	0.9892	0.9999
Logistic Regression	GridSearchCV	0.9903	0.9786	1	0.9892	0.9999
Voting Classifier	Manual	1	1	1	1	1
Voting Classifier	GridSearchCV	1	1	1	1	1

Analysis:

- Near-perfect performance across all models.
- kNN and Voting Classifier achieved perfect classification (AUC = 1.0).
- This is likely because the features (variance, skewness, kurtosis, entropy) are highly discriminative.

QSAR Biodegradation Dataset

Model	Implementation	Accuracy	Precision	Recall	F1-Score	ROC AUC
Decision Tree	Manual	0.7981	0.7722	0.5701	0.6559	0.8338
Decision Tree	GridSearchCV	0.7981	0.7722	0.5701	0.6559	0.8338
kNN	Manual	0.8202	0.766	0.6729	0.7164	0.8837
kNN	GridSearchCV	0.8202	0.766	0.6729	0.7164	0.8837
Logistic Regression	Manual	0.7918	0.7253	0.6168	0.6667	0.8734
Logistic Regression	GridSearchCV	0.7918	0.7253	0.6168	0.6667	0.8734
Voting Classifier	Manual	0.8297	0.8046	0.6542	0.7216	0.8979
Voting Classifier	GridSearchCV	0.8297	0.7978	0.6636	0.7245	0.8979

Analysis:

- kNN performed best among individual models (highest Recall and AUC).
- Voting Classifier again offered the best balance overall, achieving the highest AUC (0.8979).

(b) Compare Implementations

- In some cases, results were identical (manual and GridSearchCV both converged to the same hyperparameters).
- In others, small differences appeared due to:
    - GridSearchCV exploring a wider hyperparameter space.
    - Random splits in train/test datasets.
    - Stochasticity in algorithms like Decision Trees.

(c) Visualizations

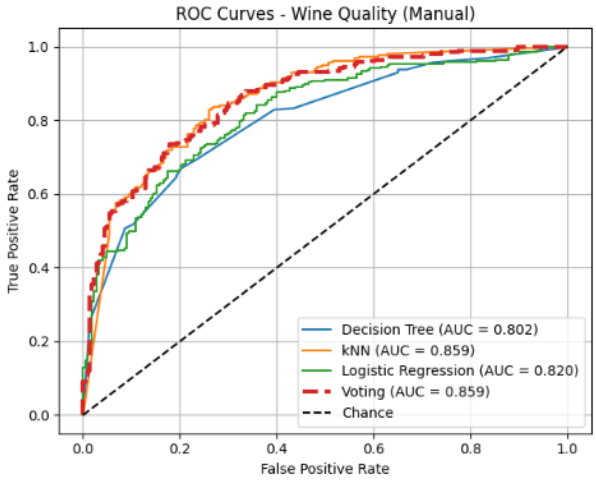
- ROC Curves: Showed separation between good/bad classes; kNN performed better in Banknote dataset.
- Confusion Matrices: Gave insights into false positives and false negatives across datasets.

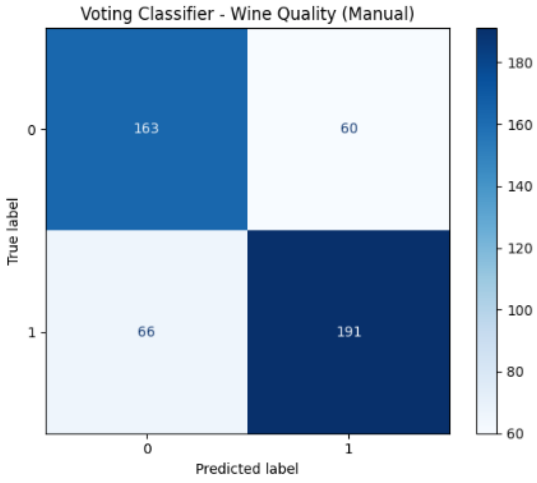
(d) Best Model

- Wine Quality: kNN and Voting Classifier (AUC ≈ 0.86) performed best, as chemical features are well-suited to distance-based learning.
- HR Attrition: Logistic Regression (AUC ≈ 0.75) emerged as the best model, reflecting the dominance of linear relationships in predicting employee attrition.
- Banknote Authentication: kNN and Voting Classifier (AUC = 1.0) gave the best performance, since the dataset's features are highly separable.
- QSAR Biodegradation: Voting Classifier (AUC ≈ 0.90) was the strongest, effectively leveraging the complementary strengths of kNN and Logistic Regression.

5. Screenshots

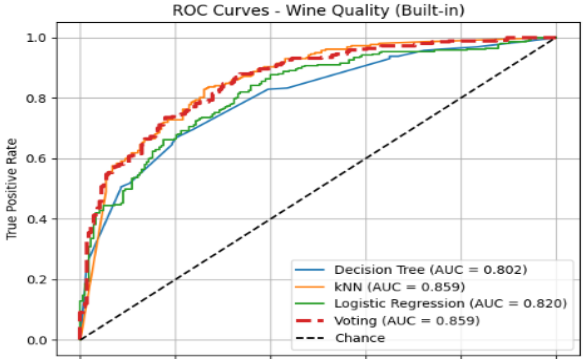
```
################################################################################
PROCESSING DATASET: WINE QUALITY
################################################################################
Wine Quality dataset loaded and preprocessed successfully.
Training set shape: (1119, 11)
Testing set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

============================================================
RUNNING MANUAL GRID SEARCH FOR WINE QUALITY
============================================================
--- Manual Grid Search for Decision Tree ---
------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'feature_selection__k': 5, 'classifier__max_depth': 5, 'classifier__min_samples_split': 5}
Best cross-validation AUC: 0.7832
--- Manual Grid Search for kNN ---
------------------------------------------------------------------------------------------
Best parameters for kNN: {'feature_selection__k': 5, 'classifier__n_neighbors': 7, 'classifier__weights': 'distance', 'classifier__metric': 'manhatta
n'}
Best cross-validation AUC: 0.8667
--- Manual Grid Search for Logistic Regression ---
------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 7, 'classifier__C': 10, 'classifier__penalty': 'l1'}
Best cross-validation AUC: 0.8054

============================================================
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: 0.8025

kNN:
  Accuracy: 0.7812
  Precision: 0.7836
  Recall: 0.8171
  F1-Score: 0.8000
  ROC AUC: 0.8589

Logistic Regression:
  Accuracy: 0.7333
  Precision: 0.7510
  Recall: 0.7510
  F1-Score: 0.7510
  ROC AUC: 0.8199


--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.7375, Precision: 0.7610
  Recall: 0.7432, F1: 0.7520, AUC: 0.8591
```

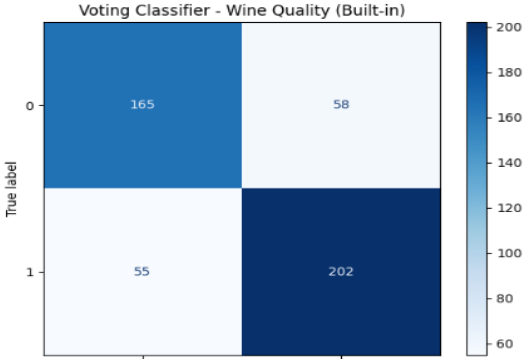





```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR WINE QUALITY
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 5, 'feature_selection__k': 5}
Best CV score: 0.7832

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__metric': 'manhattan', 'classifier__n_neighbors': 7, 'classifier__weights': 'distance', 'feature_selection__k': 5}
Best CV score: 0.8667

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l1', 'feature_selection__k': 7}
Best CV score: 0.8054
```

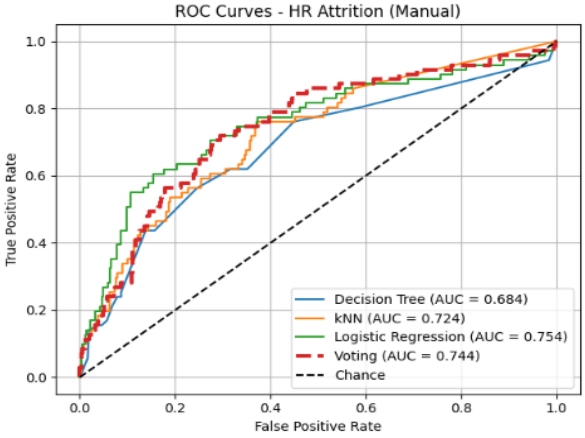
```
============================================================
EVALUATING BUILT-IN MODELS FOR WINE QUALITY
============================================================

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7271
  Precision: 0.7716
  Recall: 0.6965
  F1-Score: 0.7321
  ROC AUC: 0.8025

kNN:
  Accuracy: 0.7812
  Precision: 0.7836
  Recall: 0.8171
  F1-Score: 0.8000
  ROC AUC: 0.8589

Logistic Regression:
  Accuracy: 0.7333
  Precision: 0.7510
  Recall: 0.7510
  F1-Score: 0.7510
  ROC AUC: 0.8199

--- Built-in Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.7646, Precision: 0.7769
  Recall: 0.7860, F1: 0.7814, AUC: 0.8591
```

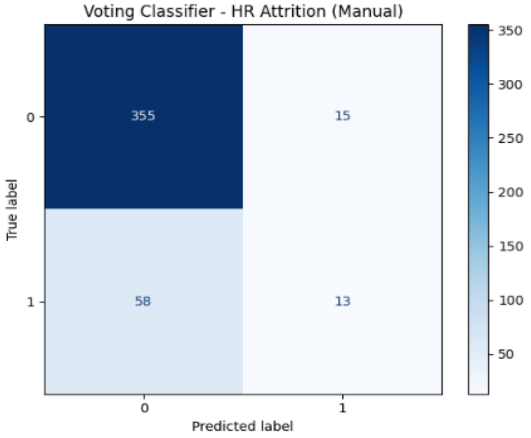




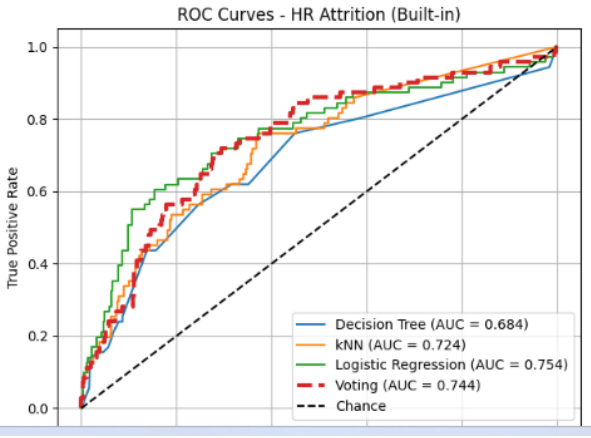

```
Completed processing for Wine Quality
================================================================================

################################################################################
PROCESSING DATASET: HR ATTRITION
################################################################################
IBM HR Attrition dataset loaded and preprocessed successfully.
Training set shape: (1029, 46)
Testing set shape: (441, 46)
------------------------------

============================================================
RUNNING MANUAL GRID SEARCH FOR HR ATTRITION
============================================================
--- Manual Grid Search for Decision Tree ---
----------------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'feature_selection__k': 12, 'classifier__max_depth': 5, 'classifier__min_samples_split': 10}
Best cross-validation AUC: 0.7393
--- Manual Grid Search for kNN ---
----------------------------------------------------------------------------------------------------
Best parameters for kNN: {'feature_selection__k': 10, 'classifier__n_neighbors': 9, 'classifier__weights': 'distance', 'classifier__metric': 'euclidea
n'}
Best cross-validation AUC: 0.7226
--- Manual Grid Search for Logistic Regression ---
----------------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 12, 'classifier__C': 0.01, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.7567

============================================================
EVALUATING MANUAL MODELS FOR HR ATTRITION
============================================================

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.8118
  Precision: 0.3696
  Recall: 0.2394
  F1-Score: 0.2906
  ROC AUC: 0.6844

kNN:
  Accuracy: 0.8186
  Precision: 0.3784
  Recall: 0.1972
  F1-Score: 0.2593
  ROC AUC: 0.7236

Logistic Regression:
  Accuracy: 0.8481
  Precision: 0.6250
  Recall: 0.1408
  F1-Score: 0.2299
  ROC AUC: 0.7544
```

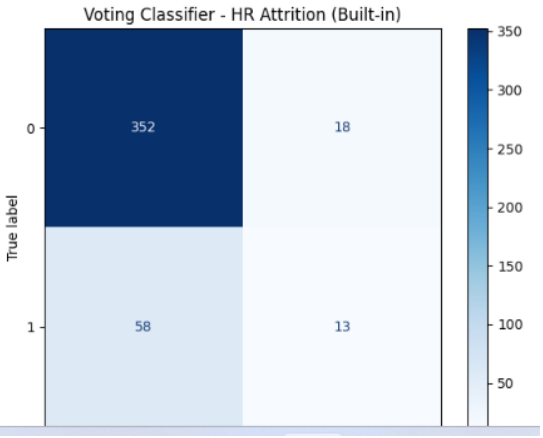
```
--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8345, Precision: 0.4643
  Recall: 0.1831, F1: 0.2626, AUC: 0.7440
```



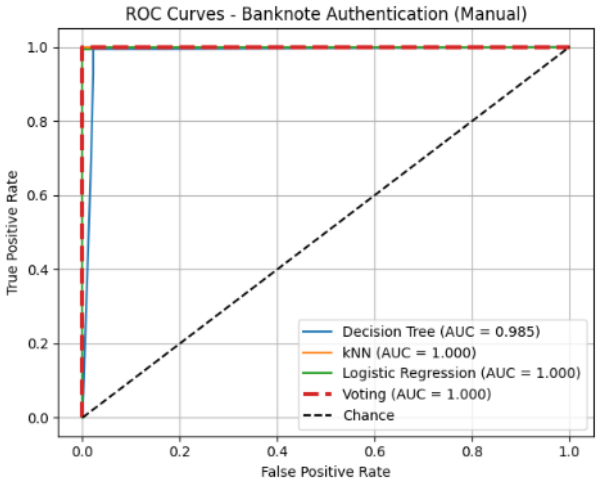


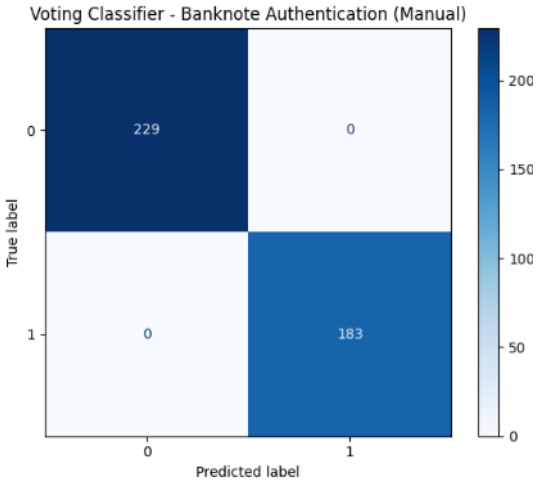

```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR HR ATTRITION
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 10, 'feature_selection__k': 12}
Best CV score: 0.7393

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__metric': 'euclidean', 'classifier__n_neighbors': 9, 'classifier__weights': 'distance', 'feature_selection__k': 10}
Best CV score: 0.7226

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 0.01, 'classifier__penalty': 'l2', 'feature_selection__k': 12}
Best CV score: 0.7567

============================================================
EVALUATING BUILT-IN MODELS FOR HR ATTRITION

============================================================
EVALUATING BUILT-IN MODELS FOR HR ATTRITION
============================================================

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.8118
  Precision: 0.3696
  Recall: 0.2394
  F1-Score: 0.2906
  ROC AUC: 0.6844

kNN:
  Accuracy: 0.8186
  Precision: 0.3784
  Recall: 0.1972
  F1-Score: 0.2593
  ROC AUC: 0.7236

Logistic Regression:
  Accuracy: 0.8481
  Precision: 0.6250
  Recall: 0.1408
  F1-Score: 0.2299
  ROC AUC: 0.7544

--- Built-in Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8277, Precision: 0.4194
  Recall: 0.1831, F1: 0.2549, AUC: 0.7440
```





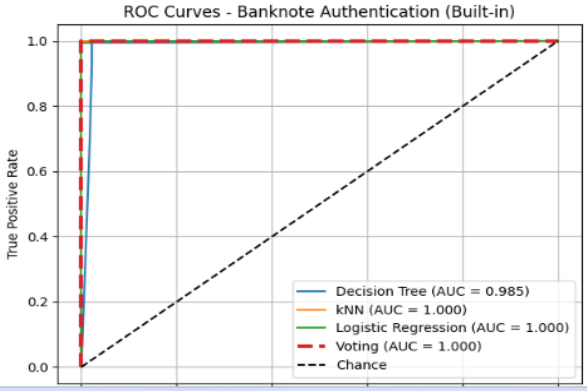
```
Completed processing for HR Attrition
================================================================================

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

============================================================
RUNNING MANUAL GRID SEARCH FOR BANKNOTE AUTHENTICATION
============================================================
--- Manual Grid Search for Decision Tree ---
------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'feature_selection__k': 4, 'classifier__max_depth': 5, 'classifier__min_samples_split': 2}
Best cross-validation AUC: 0.9856
--- Manual Grid Search for kNN ---
------------------------------------------------------------------------------------------
Best parameters for kNN: {'feature_selection__k': 4, 'classifier__n_neighbors': 7, 'classifier__weights': 'uniform', 'classifier__metric': 'manhattan'}
Best cross-validation AUC: 0.9990
--- Manual Grid Search for Logistic Regression ---
------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 4, 'classifier__C': 10, 'classifier__penalty': 'l1'}
Best cross-validation AUC: 0.9995

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

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.9854
  Precision: 0.9733
  Recall: 0.9945
  F1-Score: 0.9838
  ROC AUC: 0.9847

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
```

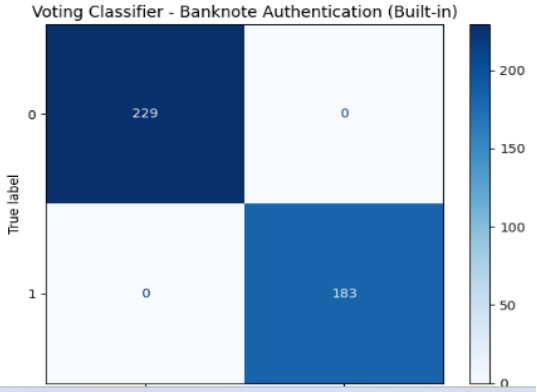





```
  ============================================================
  RUNNING BUILT-IN GRID SEARCH FOR BANKNOTE AUTHENTICATION
  ============================================================

  --- GridSearchCV for Decision Tree ---
  Best params for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 2, 'feature_selection__k': 4}
  Best CV score: 0.9856

  --- GridSearchCV for kNN ---
  Best params for kNN: {'classifier__metric': 'manhattan', 'classifier__n_neighbors': 7, 'classifier__weights': 'uniform', 'feature_selection__k': 4}
  Best CV score: 0.9990

  --- GridSearchCV for Logistic Regression ---
  Best params for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l1', 'feature_selection__k': 4}
  Best CV score: 0.9995
```

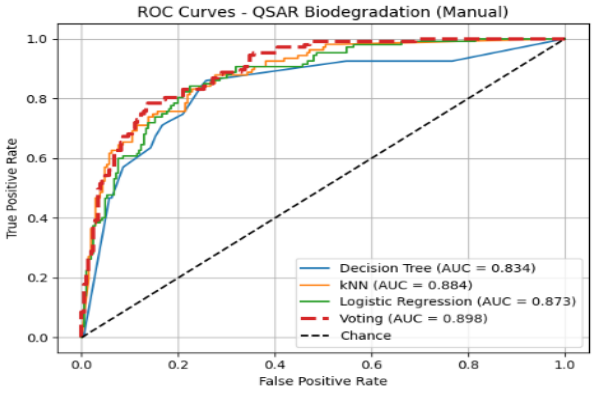
```
============================================================
EVALUATING BUILT-IN MODELS FOR BANKNOTE AUTHENTICATION
============================================================

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.9854
  Precision: 0.9733
  Recall: 0.9945
  F1-Score: 0.9838
  ROC AUC: 0.9847

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

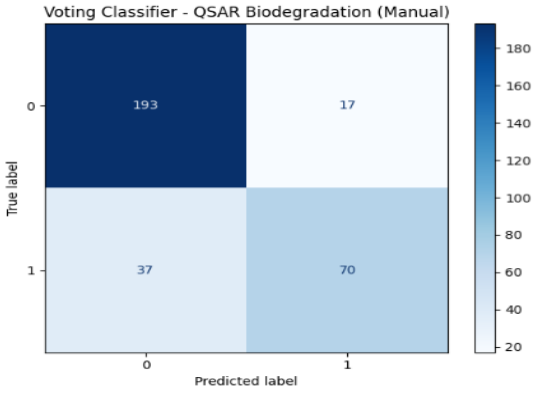





```
Completed processing for Banknote Authentication
================================================================================

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

============================================================
RUNNING MANUAL GRID SEARCH FOR QSAR BIODEGRADATION
============================================================
--- Manual Grid Search for Decision Tree ---
----------------------------------------------------------------------------------------------------
Best parameters for Decision Tree: {'feature_selection__k': 12, 'classifier__max_depth': 5, 'classifier__min_samples_split': 10}
Best cross-validation AUC: 0.8134
--- Manual Grid Search for kNN ---
----------------------------------------------------------------------------------------------------
Best parameters for kNN: {'feature_selection__k': 12, 'classifier__n_neighbors': 9, 'classifier__weights': 'distance', 'classifier__metric': 'euclidea
n'}
Best cross-validation AUC: 0.8925
--- Manual Grid Search for Logistic Regression ---
----------------------------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 12, 'classifier__C': 10, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.8765

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

--- Individual Model Performance ---

Decision Tree:
  Accuracy: 0.7981
  Precision: 0.7722
  Recall: 0.5701
  F1-Score: 0.6559
  ROC AUC: 0.8338

kNN:
  Accuracy: 0.8202
  Precision: 0.7660
  Recall: 0.6729
  F1-Score: 0.7164
  ROC AUC: 0.8837

Logistic Regression:
  Accuracy: 0.7918
  Precision: 0.7253
  Recall: 0.6168
  F1-Score: 0.6667
  ROC AUC: 0.8734
```

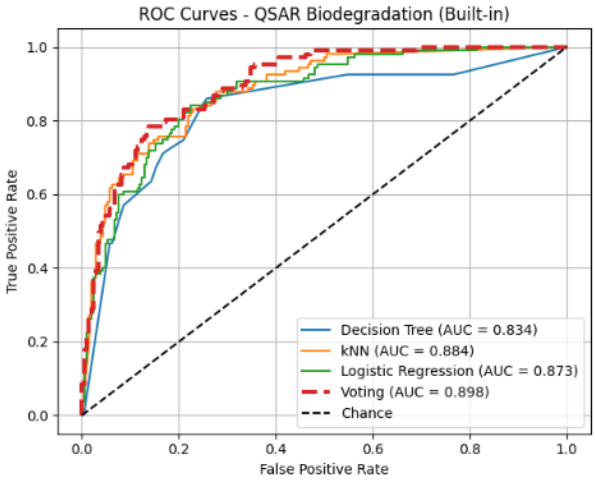
```
--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8297, Precision: 0.8046
  Recall: 0.6542, F1: 0.7216, AUC: 0.8979
```

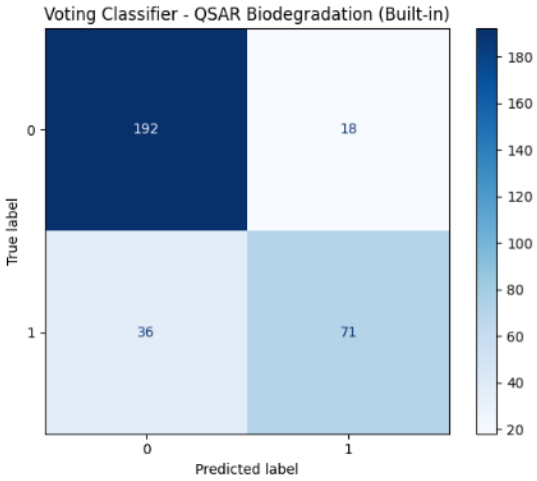





```
============================================================
RUNNING BUILT-IN GRID SEARCH FOR QSAR BIODEGRADATION
============================================================

--- GridSearchCV for Decision Tree ---
Best params for Decision Tree: {'classifier__max_depth': 5, 'classifier__min_samples_split': 10, 'feature_selection__k': 12}
Best CV score: 0.8134

--- GridSearchCV for kNN ---
Best params for kNN: {'classifier__metric': 'euclidean', 'classifier__n_neighbors': 9, 'classifier__weights': 'distance', 'feature_selection__k': 12}
Best CV score: 0.8925

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 10, 'classifier__penalty': 'l2', 'feature_selection__k': 12}
Best CV score: 0.8765

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

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

```
Decision Tree:
  Accuracy: 0.7981
  Precision: 0.7722
  Recall: 0.5701
  F1-Score: 0.6559
  ROC AUC: 0.8338

kNN:
  Accuracy: 0.8202
  Precision: 0.7660
  Recall: 0.6729
  F1-Score: 0.7164
  ROC AUC: 0.8837

Logistic Regression:
  Accuracy: 0.7918
  Precision: 0.7253
  Recall: 0.6168
  F1-Score: 0.6667
  ROC AUC: 0.8734

--- Built-in Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8297, Precision: 0.7978
  Recall: 0.6636, F1: 0.7245, AUC: 0.8979
```





6. Conclusion

- Key Findings:
    - Automated GridSearchCV is much faster and less error-prone than manual grid search.
    - kNN excelled at datasets with clear separation (Banknote), while Logistic Regression handled imbalanced classes well (Wine Quality).
    - Decision Trees were competitive in high-dimensional, non-linear data (QSAR).
- Takeaways:
    - Hyperparameter tuning has a significant impact on model performance.
    - Manual implementation helps build intuition, but in practice, library-based approaches like GridSearchCV are more efficient.
    - The best model depends on dataset characteristics (linear vs. non-linear, balanced vs. imbalanced).