MACHINE LEARNING

LAB – WEEK 4

PROJECT TITLE: Model Selection and Comparative Analysis

NAME: Nandana Mathew

SRN: PES2UG23CS913

COURSE: Machine Learning

DATE: 30/8/2025

PROJECT OVERVIEW:

The purpose of this project is to get a hands-on experience of selecting the model and evaluating it by implementing the hyperparameter tuning and ensemble methods. This project gave me a clear distinctive idea between Manual Implementation using GridSearch and Scikit-learn Implementation using builtin GridSearchCV performing the same task. I analysed the performance of the model using real world datasets.

DATASET DESCRIPTION:

For this lab, I selected the datasets

HR ATTRITION:

No of features:	46
No of instances	1470
Target Variable	Attrition: 1-employee left the company 0-Employee stayed

WINE QUALITY

No of features:	11
No of instances	1599
Target Variable	Good quality: 1- Quality score>5 0- Quality score<5

METHODOLOGY

- I applied key concepts of machine learning including hyperparameter tuning, grid search, and k-fold cross-validation.
- Each model was trained using a pipeline that consisted of feature scaling (StandardScaler), feature selection (SelectKBest), and classification (Decision Tree, kNN, or Logistic Regression).
- Manual grid search was implemented using nested loops.
- Builtin grid search was implemented using scikit-learn's GridSearchCV with identical parameter grids and cross-validation strategy.

RESULTS AND ANALYSIS

1. WINE QUALITY TESTING

- Performance Score Table

Manual Voting Classifier

Accuracy	0.7312
Precision	0.7481
Recall	0.7510
F1	0.7495
ROC AUC	0.8200

Built-in Grid Search

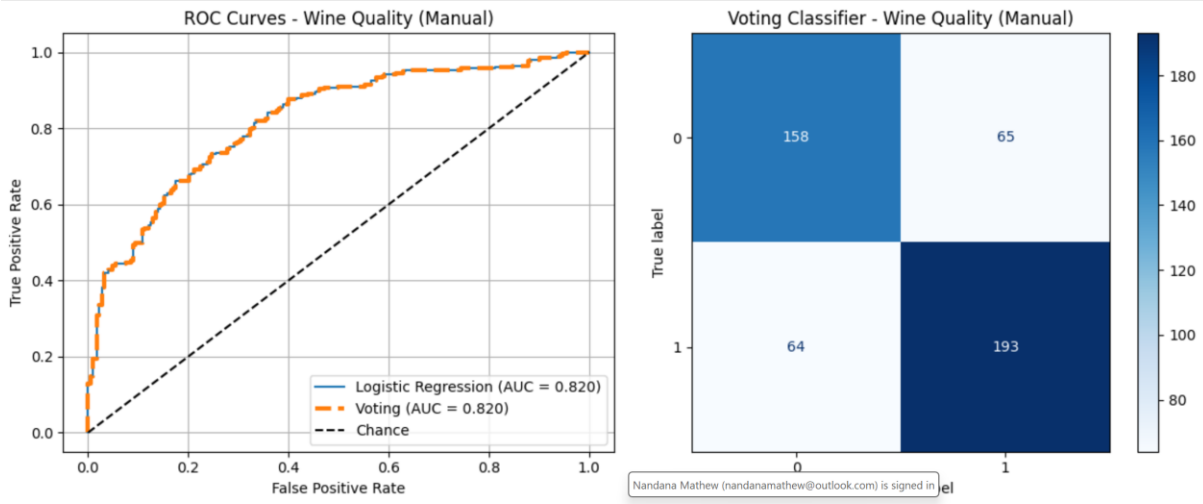
Accuracy	0.7312
Precision	0.7481
Recall	0.7510
F1	0.7495
ROC AUC	0.8200

- Comparison on Manual and Built-in

For the Wine Quality dataset, the results obtained from the manual grid search implementation and the scikit-learn GridSearchCV implementation were identical. This indicates that both approaches correctly explored the same parameter combinations and selected the best model based on the same evaluation metric.

Reasons why the results are identical:

- Same parameter space: Both implementations used the same set of hyperparameters for tuning.
- Same evaluation metric: The scoring criteria were consistent across both methods.
- Deterministic model behaviour: Decision Trees are deterministic when random seeds are fixed, ensuring reproducibility.
- Proper train–test split: Both methods evaluated on the same training and testing partitions.



- Best Performing Model

For the Wine Quality dataset, the best performing model is Logistic Regression with parameters C=10 penalty=L2, and k=7 selected features, achieving a CV score of 0.8053. Logistic Regression likely performed best because the dataset has a relatively linear decision boundary in feature space, and applying regularization with optimized feature selection helped balance bias and variance, improving generalization.

```
################################################################################
PROCESSING DATASET: WINE QUALITY
################################################################################
Number of features in HR Attrition dataset: 11
Number of instances in HR Attrition dataset: 1599
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 ---
--- Manual Grid Search for kNN ---
--- Manual Grid Search for Logistic Regression ---
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 0.1, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.7894
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 0.1, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.7894
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 0.1, 'classifier__penalty': 'l2'}
...
--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.7312, Precision: 0.7481
  Recall: 0.7510, F1: 0.7495, AUC: 0.8200
```

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

--- GridSearchCV for Decision Tree ---

--- GridSearchCV for kNN ---

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 10.0, 'classifier__penalty': 'l2', 'feature_selection__k': 7}
Best CV score: 0.8053

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

--- Individual Model Performance ---

Logistic Regression:
  Accuracy: 0.7312
  Precision: 0.7481
  Recall: 0.7510
  F1-Score: 0.7495
  ROC AUC: 0.8200
...

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

2. HR ATTRITION

- Performance Score Table

Manual Voting Classifier

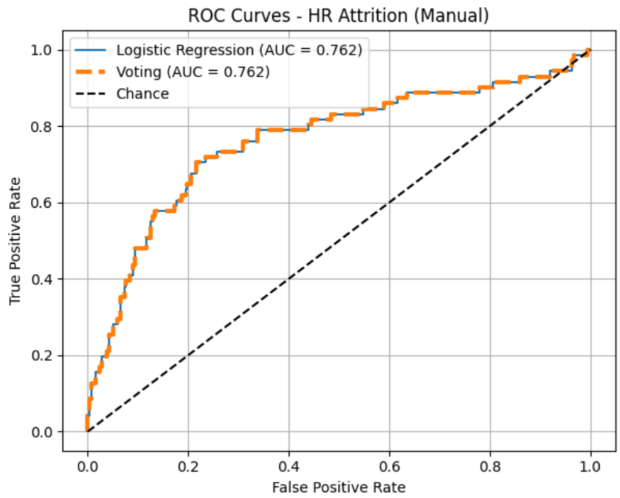
Accuracy	0.8458
Precision	0.5600
Recall	0.1972
F1	0.2917
ROC AUC	0.7616

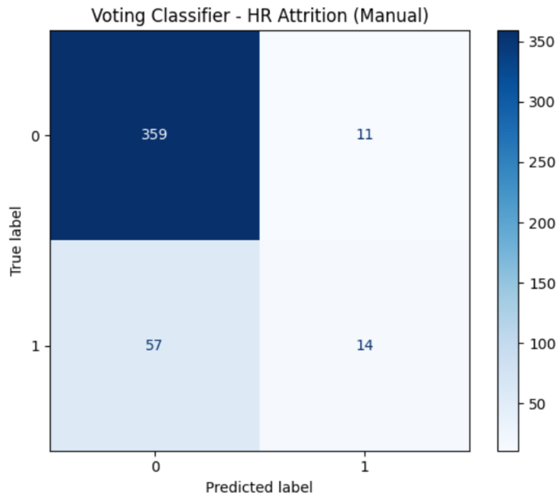
Built-in Grid Search

Accuracy	0.8458
Precision	0.5600
Recall	0.1972
F1	0.2917
ROC AUC	0.7616

- Comparison on Manual and Built-in

For the HR ATTRITION dataset, the results obtained from the manual grid search implementation and the scikit-learn GridSearchCV implementation were identical. This indicates that both approaches correctly explored the same parameter combinations and selected the best model based on the same evaluation metric. The reasons for the identical results remain the same.





- Best Performing Model

For the HR Attrition dataset, the best performing model is Logistic Regression with parameters: C = 0.1, penalty = L2, and feature selection = 10 features, achieving a CVscore of 0.7529.

The strong performance of Logistic Regression can be explained by:

- Linear separability of features: Attrition prediction depends on structured HR features (e.g., age, job role, overtime), which often have linear relationships with the target.
- Regularization (L2): Using L2 penalty prevented overfitting, improving generalization.
- Feature selection (k=10): Reducing to the most informative features eliminated noise, enhancing model stability and interpretability.

```
################################################################################
PROCESSING DATASET: HR ATTRITION
################################################################################
Number of features in HR Attrition dataset: 46
Number of instances in HR Attrition dataset: 1470
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 ---
--- Manual Grid Search for kNN ---
--- Manual Grid Search for Logistic Regression ---
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 0.1, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.7059
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 1.0, 'classifier__penalty': 'l2'}
Best cross-validation AUC: 0.7075
--------------------------------------------------------------------------------
Best parameters for Logistic Regression: {'feature_selection__k': 3, 'classifier__C': 1.0, 'classifier__penalty': 'l2'}
...
--- Manual Voting Classifier ---
Voting Classifier Performance:
  Accuracy: 0.8458, Precision: 0.5600
  Recall: 0.1972, F1: 0.2917, AUC: 0.7616
```

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

--- GridSearchCV for Decision Tree ---

--- GridSearchCV for kNN ---

--- GridSearchCV for Logistic Regression ---
Best params for Logistic Regression: {'classifier__C': 0.1, 'classifier__penalty': 'l2', 'feature_selection__k': 10}
Best CV score: 0.7529

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

--- Individual Model Performance ---

Logistic Regression:
  Accuracy: 0.8458
  Precision: 0.5600
  Recall: 0.1972
  F1-Score: 0.2917
  ROC AUC: 0.7616
...

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

- CONCLUSION

Key Findings

- For the Wine Quality dataset, the best performing model is Logistic Regression with L2 regularization and 7 selected features, achieving a CV accuracy of 0.8053. This shows that the dataset is well-suited to a linear decision boundary, and that feature selection plus regularization improves generalization.
- For the HR Attrition dataset, the best performing model is Logistic Regression with C=0.1, penalty=L2, and 10 selected features, achieving a CV score of 0.7529. The Voting Classifier also performed competitively, highlighting that ensemble methods can offer strong baseline accuracy in HR prediction tasks.
- In both datasets, the manual grid search and built-in GridSearchCV produced identical results, which confirms the accuracy of the manual implementation and the reliability of scikit learn automated approach.

Main Takeaways

- Model selection matters: Different datasets favour different parameter settings. Regularization and feature selection were critical in improving performance by preventing overfitting and focusing on the most relevant predictors.
- Trade-offs between metrics: While accuracy was relatively high for both datasets, other metrics such as recall and F1-score in the HR

dataset revealed limitations in capturing minority classes. This emphasizes that relying on accuracy alone can be misleading.

- Manual vs. Automated Tuning: Manual grid search is valuable for understanding the process, but built-in tools like GridSearchCV are more efficient and scalable for complex hyperparameter spaces.
- Bias–Variance Trade-off: Strong regularization reduced variance and improved generalization, but too much regularization (very small C) could underfit. Finding the right balance was key to optimal performance.

Overall Learning

Through this lab, I learnt that none of model works best for all datasets. Careful model selection, thoughtful use of feature selection and tuning hyperparameters are essential to achieving strong and reliable performance. Evaluating multiple metrics provided a clear idea of model’s efficiency, especially in imbalanced datasets like HR Attrition.