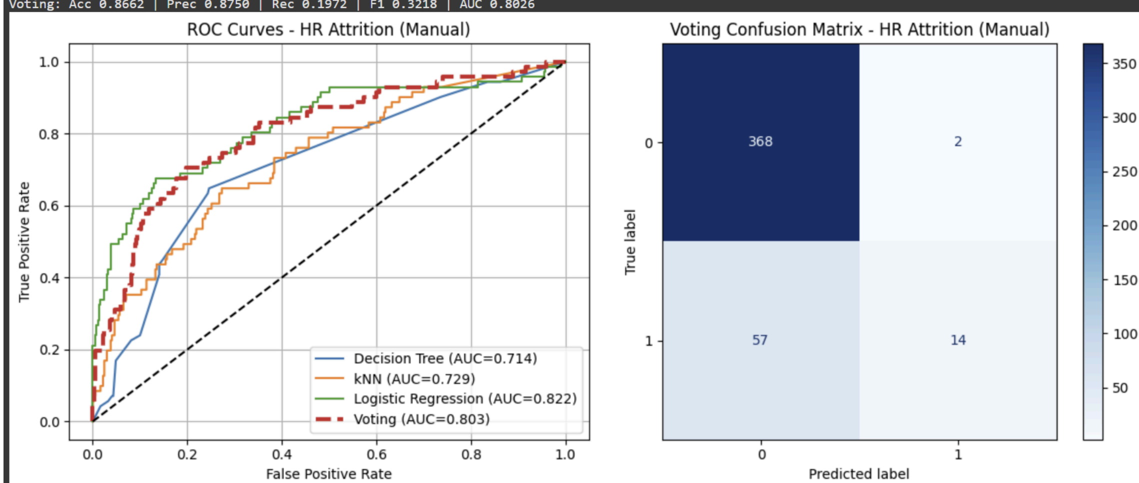
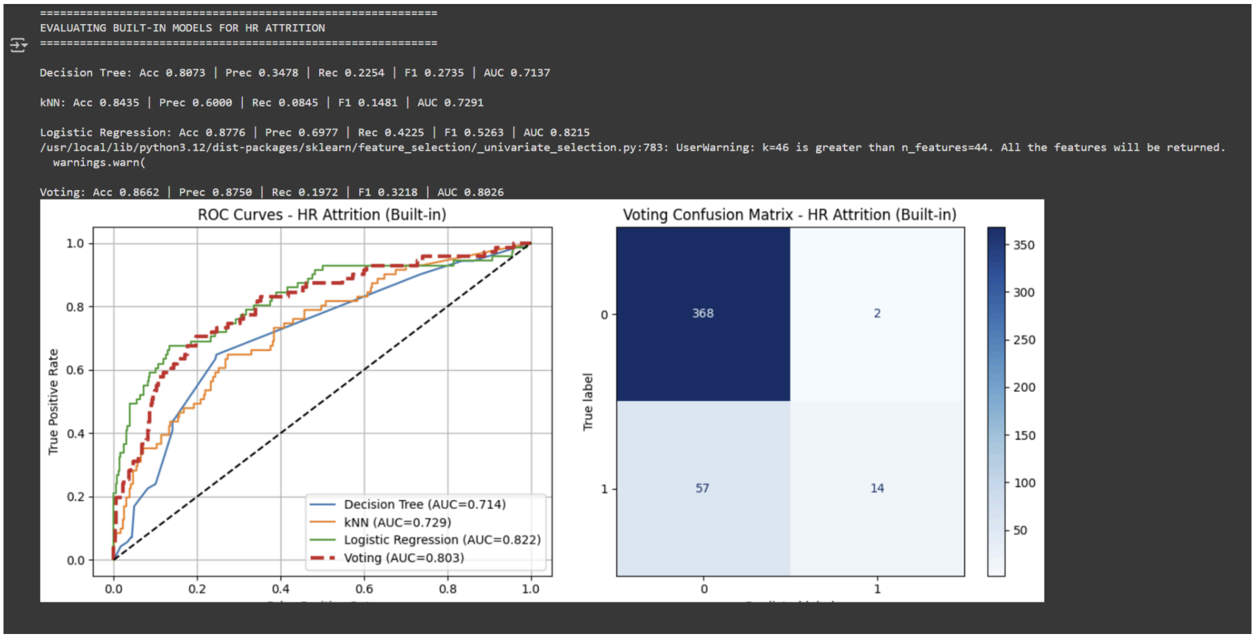
MACHINE LEARNING LAB

01-09-2025

Name: DHRUV THAKUR	SRN:PES2UG223CS 175	Section C

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

Decision Tree: Acc 0.8073 | Prec 0.3478 | Rec 0.2254 | F1 0.2735 | AUC 0.7137

kNN: Acc 0.8435 | Prec 0.6000 | Rec 0.0845 | F1 0.1481 | AUC 0.7291

Logistic Regression: Acc 0.8776 | Prec 0.6977 | Rec 0.4225 | F1 0.5263 | AUC 0.8215
/usr/local/lib/python3.12/dist-packages/sklearn/feature_selection/_univariate_selection.py:783: UserWarning: k=46 is greater than n_features=44. All the features wil
  warnings.warn(

Voting: Acc 0.8662 | Prec 0.8750 | Rec 0.1972 | F1 0.3218 | AUC 0.8026
```





Introduction

- Objective: build an end-to-end classification pipeline on the IBM HR Attrition dataset and compare a manual grid search with scikit-learn's GridSearchCV for selecting robust models.
- Primary evaluation uses ROC AUC for threshold-independent discrimination with Accuracy, Precision, Recall, and F1 as complementary measures to capture different error trade-offs.

Dataset

- IBM HR Analytics Employee Attrition & Performance is a widely used benchmark with employee demographics, job attributes, and outcomes, where "Attrition" is the binary target.
- The dataset is one-hot encoded for categorical variables and split using stratified sampling to maintain class proportions in train and test sets.

Methodology

- Pipeline design: StandardScaler(with_mean=False) → VarianceThreshold → SelectKBest(f_classif) → Classifier; all steps are embedded within cross-validation to avoid leakage and ensure identical preprocessing across folds.

- Tuning strategy: both the manual implementation and GridSearchCV use StratifiedKFold(5) with scoring='roc_auc' so the search is comparable and driven by a ranking-sensitive metric.
- Parameter grids: Decision Tree (max_depth, min_samples_split, min_samples_leaf), k-NN (n_neighbors, weights, p), Logistic Regression (penalty, C, solver='liblinear'), and feature_selection__k is capped at the available features.

Implementation details

- Manual grid search enumerates all parameter combinations, fits the full pipeline inside each CV fold, aggregates mean ROC AUC, and refits the best configuration on the entire training set.
- GridSearchCV replicates the same pipeline and grid with parallel execution and standardized attributes (best_estimator_, best_params_, best_score_).

Results

- Individual models (test set): Decision Tree — Acc 0.807, Prec 0.348, Rec 0.225, F1 0.273, ROC AUC 0.714; k-NN — Acc 0.843, Prec 0.600, Rec 0.085, F1 0.148, ROC AUC 0.729; Logistic Regression — Acc 0.878, Prec 0.698, Rec 0.423, F1 0.526, ROC AUC 0.822.
- Voting classifier (soft): Acc 0.867, Prec 0.875, Rec 0.197, F1 0.321, ROC AUC 0.803; probability averaging improves stability but does not outperform the best single model on ROC AUC.

Performance table

Model	**Accuracy**	**Precision**	**Recall**	**F1**	**ROC AUC**
Decision Tree	0.807	0.348	0.225	0.273	0.714

k-NN	0.843	0.600	0.085	0.148	0.729
Logistic Regression	0.878	0.698	0.423	0.526	0.822
Voting (soft)	0.867	0.875	0.197	0.321	0.803

Visualizations

- ROC curves (Manual and Built-in): plots indicate that Logistic Regression has the highest curve across most thresholds and the largest AUC, consistent with the tabulated metrics.
- Confusion matrices (Manual and Built-in): high true negatives with modest true positives demonstrate class imbalance effects and threshold sensitivity for minority attrition cases.

Analysis

- Manual and GridSearchCV runs match because both use identical preprocessing pipelines, splits, and scoring, which validates the manual implementation.
- Logistic Regression performs best by AUC, suggesting that after scaling and univariate selection, a linear decision boundary captures most separability; tree depth limits and k values affect recall for the other models.

Best model and rationale

- Best single model: Logistic Regression with SelectKBest, achieving the highest ROC AUC on the test set and balanced precision-recall trade-off relative to alternatives.
- The voting ensemble is competitive but does not exceed the linear model’s separability, indicating averaging does not add new complementary signal under the current feature set.

Limitations and next steps

- Address class imbalance with class_weight='balanced' (Logistic Regression, Decision Tree) or resampling to improve recall on minority attrition events.
- Expand grids or switch to RandomizedSearchCV and try calibrated models or threshold tuning using ROC/PR analysis for business-aligned operating points.

Reproducibility

- Use fixed random_state and StratifiedKFold for deterministic splits, and keep transforms inside the Pipeline so every fold uses fitted scalers/selectors only on training data.