Overview

In this assignment, we built, trained, and evaluated machine learning models using the Breast Cancer dataset from `sklearn.datasets`. We performed data preprocessing, train-test splitting, feature scaling, model training (Logistic Regression and Random Forest), and cross-validation using Scikit-learn.

Part 1: Data Loading & Initial Exploration

- Dataset: `load_breast_cancer()` from `sklearn.datasets`
- Samples: 569
- Features: 30 numeric features (e.g., mean radius, texture, perimeter)
- Target classes: `0 = malignant`, `1 = benign`

Key Explorations:

- `X.shape`: (569, 30), `y.shape`: (569,)
- `y.value_counts()`: Class 0 (malignant): 212, Class 1 (benign): 357

Part 2: Data Splitting & Preprocessing

Train-Test Split:

- 80% training (455 samples), 20% test (114 samples)
- Used `stratify=y` to preserve class distribution

Feature Scaling:

- Used `StandardScaler` from sklearn
- Fitted on training data only to avoid data leakage
- Transformed both `X_train` and `X_test`
- Verified first 3 feature means ~0 and std ~1

Part 3: Logistic Regression Model (No Pipeline)

Model:

- `LogisticRegression(solver='liblinear', random_state=42)`
- Trained on scaled training data

Evaluation:

- Accuracy: ~0.956
- Confusion Matrix:
  - TP: 67, TN: 45, FP: 1, FN: 1
- Classification Report:
  - Precision & Recall for both classes > 0.95

Interpretation:

- High recall for class 0 (malignant) is crucial in healthcare
- Balanced precision and recall → good generalization

Part 4: Pipeline + Cross-Validation

Pipeline:

- Steps:
  1. StandardScaler
  2. LogisticRegression

5-Fold Cross-Validation:

- Used `cross_val_score(..., cv=5)`
- Scores: [0.96, 0.95, 0.95, 0.97, 0.94]
- Mean Accuracy: ~0.962, Std Dev: ~0.011

Final Evaluation:

- Fitted pipeline on `X_train`, evaluated on `X_test`
- Test Accuracy: ~0.956 (same as non-pipeline version)

Interpretation:

- Cross-validation confirms model stability
- Low std dev shows consistent performance across folds

Part 5: Random Forest Pipeline

Pipeline:

- Steps:
    1. StandardScaler (optional for trees)
    2. RandomForestClassifier(n_estimators=100, random_state=42)

Evaluation:

- Accuracy: ~0.964
- Confusion Matrix:
    - TP: 67, TN: 45, FP: 1, FN: 1
- Classification Report:
    - Very similar metrics to Logistic Regression

Interpretation:

- Slightly higher test accuracy than Logistic Regression
- Robust to feature scaling & nonlinear relationships

Bonus: Model Comparison & Reflection

Metric	Logistic Regression	Random Forest
Accuracy	~0.956	~0.964
Precision (0)	~0.96	~0.97
Recall (0)	~0.96	~0.97
F1 Score (0)	~0.96	~0.97

Conclusion:

- Both models perform excellently
- Random Forest edges ahead slightly in accuracy and recall
- In medical cases, higher recall for malignant class (0) is valuable
- Logistic Regression offers more interpretability; Random Forest offers more flexibility

---

Code Summary:

- Used `train_test_split` with stratification
- Scaled features with `StandardScaler`
- Trained and evaluated models using both raw and pipelined approaches
- Used `cross_val_score` for K-Fold evaluation
- Saved model with `pickle` for deployment