Assignment 3 – Cancer Classification with Logistic Regression, Naive Bayes, SVM, and PCA

Name: Raj Patel
Student ID: 801334777
Assignment: Assignment 3

GitHub Repository:
https://github.com/RajP-17/Intro-To-ML-Assignments

Use of Gen AI:
The author acknowledges the use of ChatGPT and Gemini in the preparation or completion of this assignment. These tools were used for: debugging support, clarification of scikit-learn functions, performance comparison phrasing, plotting enhancements, and narrative structure in this report.

---

1. Introduction

In this assignment, we investigated the classification of breast cancer (malignant vs. benign) using multiple machine learning techniques on the **Breast Cancer Wisconsin dataset** from `sklearn.datasets`. The models implemented include:

- Logistic Regression (with and without regularization)
- Naive Bayes
- Support Vector Machine (SVM)
- Logistic Regression with PCA

Each model was evaluated using standard classification metrics—**accuracy, precision, recall, and F1-score**—along with visual tools such as **confusion matrices** and **metric plots**. Dimensionality reduction was explored via Principal Component Analysis (PCA) to optimize model performance.

---

2. Problem 1: Logistic Regression

2.1 Model without Regularization

- **Features Used:** All 30 input features.
- **Preprocessing:** Standardization using `StandardScaler`.
- **Data Split:** 80% training / 20% testing.
- **Classifier:** `LogisticRegression` from scikit-learn.

Results:

- Accuracy: **96.49%**
- Precision: **96.55%**
- Recall: **98.21%**
- Confusion Matrix: High true positive and true negative counts with minimal misclassifications.

2.2 Model with L2 Regularization

- **Penalty Used:** `'l2'` with default `C=1.0`.
- **Observations:** Regularization did not significantly change performance.

Metrics remained identical:

- Accuracy: **96.49%**
- Precision: **96.55%**
- Recall: **98.21%**

---

3. Problem 2: Naive Bayes Classifier

3.1 Model Setup

- **Classifier Used:** `GaussianNB` from scikit-learn.
- **Data Split:** 80% training / 20% testing.

3.2 Results

- Accuracy: **92.98%**
- Precision: **94.44%**
- Recall: **94.64%**
- F1 Score: **94.54%**

3.3 Observations

- Naive Bayes performed slightly worse than logistic regression.
- Its assumption of feature independence may not hold true for the cancer dataset.
- Still provided solid baseline performance.

---

4. Problem 3: Support Vector Machine (SVM)

4.1 Model Configuration

- **Classifier Used:** `SVC(kernel='linear')`.
- **Data Split:** 80/20 with scaling.

4.2 Results

- Accuracy: **97.37%**
- Precision: **98.15%**
- Recall: **98.21%**
- F1 Score: **98.18%**

4.3 Insights

- SVM outperformed both logistic regression and Naive Bayes.
- Best performance across all metrics.
- Robust to feature correlations and scales well with standardized data.

---

5. Problem 4: Logistic Regression with PCA

5.1 PCA-Based Dimensionality Reduction

- PCA was applied to reduce the dataset to **K principal components**, ranging from 1 to 30.
- Logistic regression was trained for each value of K.
- Performance metrics were recorded for each.

5.2 Optimal Component Selection

- **Optimal K:** 15 components yielded the best performance.

At K = 15:

- Accuracy: **96.49%**
- Precision: **97.14%**

- Recall: **97.30%**
- F1 Score: **97.22%**

5.3 Observations

- Performance with PCA was comparable to full-feature logistic regression.
- PCA offers dimensionality reduction with minimal loss in performance.
- A good strategy for reducing overfitting and computational complexity.

---

6. Conclusions

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	96.49%	96.55%	98.21%	N/A
Naive Bayes	92.98%	94.44%	94.64%	94.54%
SVM	97.37%	98.15%	98.21%	98.18%
Logistic + PCA (K=15)	96.49%	97.14%	97.30%	97.22%

- **SVM** emerged as the best classifier in terms of overall performance.
- **Logistic Regression with PCA** offers a strong alternative with fewer features.
- **Naive Bayes** provides simplicity but at a minor cost to accuracy.

---

7. Assumptions and Notes

- Dataset was loaded via `sklearn.datasets.load_breast_cancer`.

- All experiments used a consistent 80/20 train-test split.
- StandardScaler was applied where required.
- Default hyperparameters were used unless otherwise noted.