Comprehensive Machine Learning Pipeline for Breast Cancer Diagnosis

By: Bernard Djoko

# Abstract

This project comprises a complete machine learning pipeline developed for the prediction of breast tumors as benign or malignant from the Breast Cancer Wisconsin (Diagnostic) data set found in the UCI Machine Learning Repository.

The pipeline includes data preparation, feature engineering, model training, evaluation, and visualization. We've implemented three classification algorithms—Decision Tree, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM)—and evaluated their performances on the basis of accuracy, precision, recall, F1-score, and cross-validation metrics.

Feature selection was guided by importance scores from Random Forest, and imbalance in the data was addressed using the SMOTE method. KNN proved to be the best-performing model with 97% accuracy and good generalization capability. Model behavior was interpreted and compared using visualizations such as confusion matrices and decision boundary plots. All steps of the pipeline are recorded in this report with findings of practical interest to predictive diagnostics in real-world settings being highlighted.

# 1. Introduction

Breast cancer remains one of the leading causes of mortality among women worldwide. Early and accurate diagnosis plays a critical role in improving treatment outcomes. In this project, we address the binary classification problem of identifying whether a tumor is benign or malignant using digitized features extracted from fine needle aspirate (FNA) of breast masses.

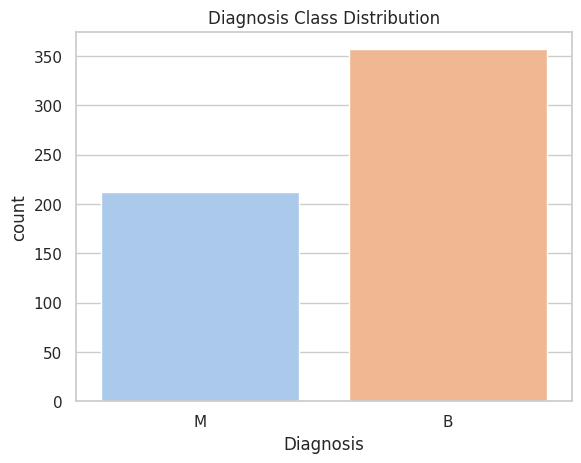
We use the **Breast Cancer Wisconsin (Diagnostic)** dataset, which includes 30 numerical features derived from cell nuclei images, such as radius, texture, smoothness, and area, alongside a target variable (**Diagnosis**) indicating malignancy. The dataset contains **569** **instances**, of which 212 are malignant and 357 are benign—indicating a mild class imbalance.

Our goal is to build an end-to-end machine learning pipeline that includes data preprocessing, feature selection, model training, evaluation, and interpretation.

# 2. Data Preparation

We loaded the dataset using pandas and performed initial cleaning. No missing values were found. The 'ID' column was dropped as it provides no predictive value.

The class distribution revealed a moderate imbalance, with benign cases outnumbering malignant ones.



[Figure 1: Diagnosis Class Distribution]

Histograms of key features showed that malignant tumors generally have higher radius, area, and perimeter measurements.

A group of graphs showing different sizes of data

AI-generated content may be incorrect.

[Figure 2: Feature Distributions by Diagnosis]

The correlation heatmap revealed strong relationships among specific features (radius\_mean, perimeter\_mean, area\_mean; concave\_points\_mean and concavity\_mean), which helped guide feature selection.

A screenshot of a computer generated image

AI-generated content may be incorrect.

[Figure 3: Correlation Heatmap]

# 3. Feature Engineering

StandardScaler was used to normalize the feature set. We handled class imbalance using SMOTE, which synthetically generated samples for the minority class.

A graph showing the difference between smote and malignant

AI-generated content may be incorrect.

[Figure 4: Class After SMOTE]

We engineered a new feature called **area\_ratio = area\_mean / perimeter\_mean** to capture compactness. Although not the most important feature, it provided additional context.

**Random Forest** was used to rank feature importance. The **top 10 features** were selected for model training. This analysis guided dimensionality reduction, improving model performance and interpretability while avoiding overfitting.

A graph of different colored bars

AI-generated content may be incorrect.

[Figure 5: Top 10 Feature Importances]

# 4. Model Development

We selected three classifiers: **Decision Tree, KNN**, and **SVM.** We split the dataset with stratified sampling (80/20) to maintain class balance. SMOTE was applied to the training set to address class imbalance.

Each model was trained on the top 10 features after scaling and resampling.

Data

Train/Test Split

SMOTE

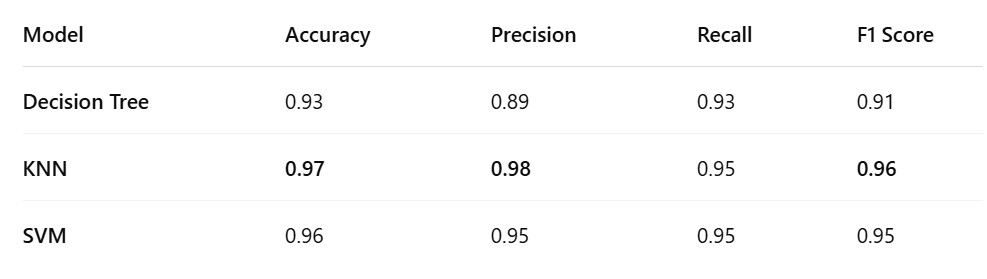
Model Training

Evaluation

[Figure 6: Model Training Workflow]

# 5. Model Evaluation

Each model was evaluated on the original test set using accuracy, precision, recall, and F1-score. KNN outperformed the others across all metrics.



[Table 1: Model Evaluation Metrics]

Confusion matrices were plotted to analyze true vs false classifications for each model.

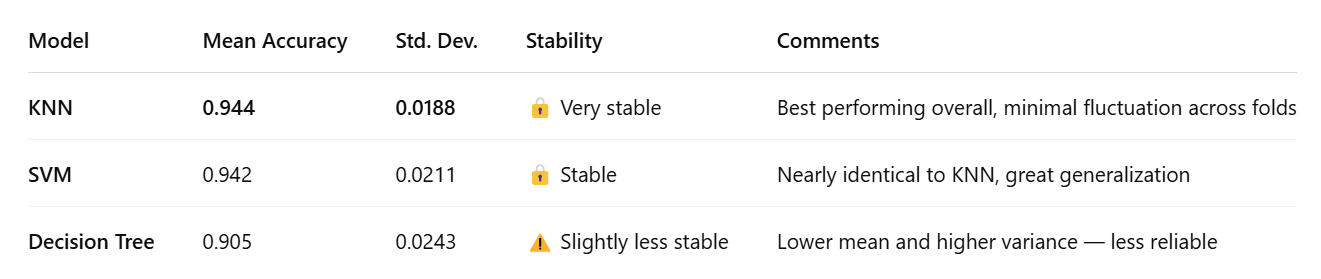
**KNN** had the fewest misclassifications and zero false negatives, making it ideal for reducing missed diagnoses.

A green squares with white text

AI-generated content may be incorrect.

[Figure 7: Confusion Matrices]

5-fold cross-validation was performed to ensure model stability. **KNN** had the highest mean accuracy and lowest variance.



[Table 2: Cross-Validation Accuracy]

A graph of a bar chart

AI-generated content may be incorrect.

[Figure 8: CV Accuracy with Error Bars]

# 6. Visualization & Interpretation

We visualized feature importance, performance metrics, and model decision boundaries using PCA-reduced 2D plots.

A graph of different colored bars

AI-generated content may be incorrect.

[Figure 9: Comparative Performance Bar Plot]

This visualization highlights KNN's superiority, especially in terms of F1-score and recall, which are crucial in medical diagnostics.

A graph showing a red and green chart

AI-generated content may be incorrect.

[Figure 10: Decision Boundaries (PCA)]

When projected to 2D using PCA, the decision boundaries reveal interesting model characteristics. The **Decision Tree** shows overfitting behavior with jagged boundaries. **KNN** adapts smoothly to local class distributions, while **SVM** presents a clean, margin-maximizing boundary — making it well-suited for generalization. These visuals confirm the quantitative findings where **KNN** and **SVM** outperformed the Decision Tree

# 7. Conclusion & Recommendations

**KNN** was the most accurate and robust model, making it the best choice for deployment. Future work could explore ensembles, explainability tools, and testing on external datasets.