AI for Medicine – Project Report

Prof. Stefano Diciotti – University of Bologna

Academic Year: 2024/2025

Student Name: Maede Shahbazi Zade

Degree Program: MSc in Electronics for intelligent systems, big-data and internet of things

Submission Date: 07/17/2025

Note: The report should be concise but complete. Typical length: 8–10 pages.

1. Project Title

Breast Cancer Prediction Using Machine Learning Algorithms

2. Problem Statement

Breast cancer is one of the most common cancers affecting women worldwide and remains a leading cause of cancer-related deaths. Early and accurate diagnosis is critical for improving patient outcomes, guiding treatment decisions, and reducing mortality rates. Traditional diagnostic methods often require invasive procedures or are time-consuming and expensive.
This project addresses the problem of breast cancer diagnosis by developing machine learning models that can classify tumors as benign or malignant based on diagnostic imaging features. Using the Breast Cancer Wisconsin (Diagnostic) dataset, the goal is to build a predictive system that can support clinicians in making faster and more accurate diagnoses, thereby contributing to better healthcare and early intervention.

3. Objective of the Study

The primary objective of this study is to develop and evaluate supervised machine learning models capable of predicting whether a breast tumor is malignant or benign based on various diagnostic features extracted from medical imaging. The goal is to build an accurate, reliable, and interpretable classification system that supports early detection of breast cancer, using techniques such as random forest within a well-structured ML pipeline.

4. Dataset Description

Breast Cancer Wisconsin (Diagnostic) Dataset (Kaggle).
Number of samples: 569 patient records
Type of data: Tabular data consisting of numerical features extracted from digitized images of breast mass fine needle aspirates (FNAs).
Features: 30 numeric features representing various cell characteristics (e.g., radius, texture, perimeter, area, smoothness).
Available labels:
1: Malignant (cancerous)
0: Benign (non-cancerous)
Issues such as class imbalance or noise:
Class Imbalance:
There is **mild class imbalance** in the dataset:

Malignant (1): 212 samples (≈37%)

Benign (0): 357 samples (≈63%)

Missing Values:
The dataset is mostly clean, but initial preprocessing ensured any missing data was removed.

5. Data Preprocessing

To prepare the dataset for machine learning analysis, several preprocessing steps were applied:
Data Cleaning:
Irrelevant or broken columns such as id and Unnamed: 32 were removed from the dataset to avoid noise and redundancy.
The target column diagnosis was encoded as a binary variable: *Malignant* (M) was mapped to 1 and *Benign* (B) to 0.
Missing Data Handling:
A check for missing values was performed across all columns.
Any rows containing missing values were removed to ensure a clean dataset for model training.
Feature and Target Separation:
The input features (X) were defined as all columns except diagnosis.
The target variable (y) was set as the diagnosis column.

Data Normalization:
A StandardScaler was used within a pipeline to standardize the feature values. This ensures that all features contribute equally to model training by centering them around zero and scaling to unit variance.
Class Balance Check:
A class distribution plot was generated to check for imbalances between benign and malignant cases. This is important to understand model bias potential.
Summary Statistics:
The dataset was summarized to confirm it contains 569 subjects and 30 numerical diagnostic features used for prediction.

6. Avoiding Data Leakage

To ensure a valid and unbiased evaluation of model performance, several strategies were used to prevent data leakage:
Pipeline Integration:
All preprocessing steps such as feature scaling (StandardScaler) and imputation (SimpleImputer, when used with Random Forest) were encapsulated within a Pipeline. This ensures that transformations are fit **only on the training folds** and then applied to validation data — avoiding any information leakage from the test set into the training process.

Nested Cross-Validation:
A nested cross-validation setup was employed, where:
The **inner loop** (GridSearchCV) performs hyperparameter tuning **only using the training portion** of each outer fold.
The **outer loop** evaluates model performance on truly unseen data.
This two-level CV prevents any leakage of test fold information during model selection or tuning.
No Global Preprocessing:
Feature selection, scaling, and hyperparameter tuning were **not applied to the entire dataset beforehand**. Instead, they were handled **within each fold** using the pipeline and GridSearchCV.
Proper Target Separation:
The target column (diagnosis) was explicitly separated before any transformation. All transformations were applied strictly to the feature matrix X.

7. Machine Learning Pipeline

Models Used:
Two supervised machine learning models were implemented and compared:

- Logistic Regression: A linear model used as a baseline. It was integrated into a pipeline with feature scaling (StandardScaler) and evaluated using nested cross-validation.
- Random Forest Classifier: An ensemble method that builds multiple decision trees and combines their predictions to improve accuracy and control overfitting. This was the final selected model due to its superior performance.

Validation Techniques:
A nested cross-validation strategy was applied:

- The inner loop (3-fold) used GridSearchCV to tune hyperparameters on the training folds.
- The outer loop (3-fold) estimated generalization performance on unseen test folds.

This two-level cross-validation framework ensured unbiased performance estimates and prevented data leakage during model selection.
Evaluation Metrics:
The models were evaluated using several classification metrics:

- Accuracy: Overall correctness of predictions
- Precision: Proportion of true positives among all predicted positives (focus on reducing false positives)
- Recall: Proportion of true positives among all actual positives (focus on reducing false negatives)
- F1 Score: Harmonic mean of precision and recall
- ROC Curve & AUC: Visual and quantitative measure of classification performance across thresholds; AUC indicates the model's overall ability to distinguish between classes

8. Results

8.1. Model Performance Comparison

Metric	Random Forest	Logistic Regression
Nested CV Accuracy	**0.965**	**0.956**
Train Accuracy (avg)	**1.000**	**-**
Test Accuracy (avg)	**0.956**	**-**
Mean Squared Error (MSE)	**0.044**	**-**
Precision (avg)	**0.948**	**-**
Recall (avg)	**0.934**	**-**
F1 Score (avg)	**0.941**	**-**

Interpretation:

- Logistic Regression achieved slightly higher overall accuracy.
- Random Forest performed better on class-specific metrics (Precision, Recall, F1), especially for malignant cases.
- Random Forest showed perfect training accuracy, indicating possible overfitting, but still generalized well.

8.2. ROC Curves – Random Forest (Outer Folds)

The ROC curves for each of the outer CV folds (Random Forest) indicate strong performance with **AUC values ranging from ~0.98 to 0.99**, demonstrating excellent class separability.

8.3. Confusion Matrix – Best Random Forest Model

Actual/Predicted	Benign (0)	Malignant (1)
Benign (0)	122 (TN)	3 (FP)
Malignant (1)	4 (FN)	64(TP)

- **Test Accuracy (this fold):** 0.963
- **Precision (Malignant):** 0.955
- **Recall (Malignant):** 0.941
- **F1 Score (Malignant):** 0.948

Note: The confusion matrix below reflects predictions from one test fold (1/3 of the data) during nested cross-validation with 3 outer folds. The full dataset contains 569 samples.

8.4. Classification Report – Best Random Forest Model

Class	Precision	Recall	F1-Score	Support
Benign (0)	0.97	0.98	0.97	122
Malignant (1)	0.96	0.94	0.95	68
Accuracy	-	-	**0.96**	**190**
Macro Avg	0.96	0.96	0.96	190
Weighted Avg	0.96	0.96	0.96	190

8.6. Best Hyperparameters – Random Forest

Hyperparameter	Selected Value
clf__n_estimators	50
clf__max_depth	None
clf__min_samples_split	2

8.7. Summary

Both Logistic Regression and Random Forest achieved strong results, but:

- **Logistic Regression** slightly outperformed in raw accuracy.
- **Random Forest** provided better performance in detecting the malignant class, which is critical in medical diagnostics.

Random Forest was therefore chosen as the final model due to its balanced and interpretable performance across key clinical metrics.

9. Discussion

The Random Forest model demonstrated high classification performance across all folds of the dataset using nested cross-validation. The combination of high AUC scores, balanced precision and recall, and strong performance on the test fold suggests the model is robust and generalizes well.

Strengths:

- Excellent discriminative power (AUC ≈ 0.99)
- Balanced sensitivity and specificity
- Interpretability through feature importance analysis

Limitations:

- Slight overfitting observed (training accuracy = 1.0)
- The dataset size is relatively small (569 samples)

- Only one dataset was used; external validation on a different cohort would strengthen generalizability
- No deep learning or image-based features were considered — only tabular data was used10. Conclusions and Future Work
- This project successfully developed a machine learning pipeline using Random Forest to predict breast cancer malignancy based on diagnostic imaging features.

Conclusions:

- The model achieved high performance with ~96% accuracy and excellent recall.
- It is capable of supporting early detection of breast cancer using a non-invasive and automated approach.

Future Work:

- Explore more advanced models (e.g., XGBoost, LightGBM, or neural networks)
- Validate on external datasets or real clinical data
- Incorporate image-based features or raw histopathological images

11. Ethics and Data Privacy

The dataset used in this project (Breast Cancer Wisconsin Diagnostic Dataset) is publicly available from the UCI Machine Learning Repository and Kaggle for academic and research purposes.

- No personally identifiable information (PII) is present in the data.
- All data preprocessing and model development were done using ethical machine learning practices.
- The code and analysis are intended for educational use and should not be deployed in real clinical settings without regulatory approval.

12. Code and Reproducibility

- **Notebook Title:** Breast_Cancer_Prediction.ipynb
- Developed in Google Colab
- Python Version: 3.x

Key Libraries Used:

- pandas, numpy, matplotlib, seaborn
- scikit-learn (sklearn)

How to Run:

- Upload the notebook to **Google Colab**
- Ensure internet connection to access the dataset from the GitHub raw link
- Run all cells sequentially
- No external dependencies beyond standard Python ML libraries were used.

13. References

Dua, D., & Graff, C. (2017). *UCI Machine Learning Repository: Breast Cancer Wisconsin (Diagnostic) Dataset*. University of California, Irvine. Retrieved from https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

Scikit-learn developers. (n.d.). *Plot ROC curve*. Scikit-learn. Retrieved July 17, 2025, from https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html