## **Assignment 2**

**Student Name**: Izabella Gogaeva  
**Student ID**: 7115152300001  
**Date**:9/05/2025  
**GitHub Repository**: https://github.com/capybbarra/Assignment-2-MLinCB

### **Abstract**

Accurate classification of breast cancer subtypes using machine learning is critical for early diagnosis and personalized treatment. However, conventional cross-validation methods can yield overly optimistic estimates of model performance due to data leakage or hyperparameter overfitting. This report applies repeated nested cross-validation (rnCV) to evaluate classification models on a breast cancer dataset. rnCV mitigates bias by nesting hyperparameter tuning within a separate outer loop for model evaluation and further improves reliability through repetition across multiple random splits. We evaluated models including Support Vector Machines, Random Forests, and Logistic Regression using metrics such as accuracy, AUC, and F1-score. Results show that rnCV reduces performance variance and provides a more trustworthy measure of generalizability, reinforcing its value in high-stakes medical classification tasks. Our findings support the adoption of rnCV in biomedical machine learning pipelines to ensure reproducibility and clinical relevance.

## **Introduction**

#### Problem Statement

Breast cancer is one of the most frequently diagnosed cancers in women globally and remains a leading cause of cancer-related mortality [4]. In recent decades, machine learning classifiers have played an increasingly important role in breast cancer research by enabling the prediction of disease status and the identification of tumor subtypes based on clinical, imaging, and gene expression data [7], [2]. However, the effectiveness of these models is often undermined by inadequate validation strategies, which can produce biased estimates of predictive performance and lead to misleading conclusions.

#### Significance

In clinical applications, overestimating the performance of diagnostic models can result in misplaced trust in their decisions, ultimately affecting patient safety. Therefore, reliable and statistically sound model evaluation techniques are critical to ensure the robustness, reproducibility, and clinical relevance of machine learning findings. Among various approaches, **repeated nested cross-validation (rnCV)** has been advocated as one of the most rigorous methods for estimating generalization performance, particularly in settings with limited data or imbalanced classes [5], [6].

#### Related Work

Prior studies have highlighted the drawbacks of traditional train-test splits and basic k-fold cross-validation, where the process of hyperparameter tuning is often intertwined with model evaluation, resulting in optimistic bias [1]. Nested cross-validation addresses this issue by isolating model selection from performance assessment, thereby yielding more reliable error estimates. Further improvements such as repeated nested CV have been shown to reduce the variance introduced by random train-test splits, enhancing the stability and generalizability of performance estimates [3].

#### **Technical Approach**

In this study, we apply repeated nested cross-validation to assess the performance of multiple classification models—including Logistic Regression, Support Vector Machines (SVM), and Random Forests—on a breast cancer dataset. We evaluate models using accuracy, AUC, precision, recall, and F1-score, performing multiple repetitions to ensure statistical robustness..

#### **Contributions**

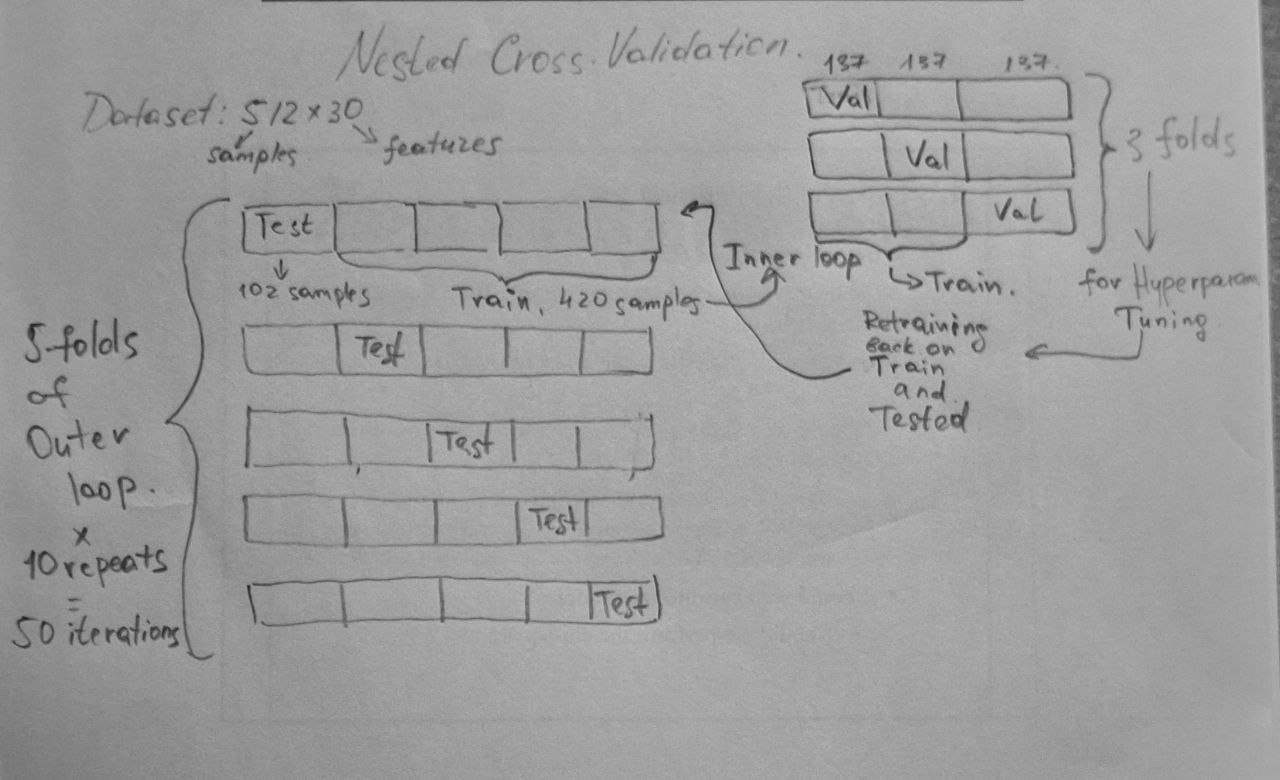
* Application of repeated nested cross-validation to breast cancer classification tasks
* Evaluation of multiple classifiers using clinically relevant metrics
* Demonstration of how rnCV reduces overfitting and provides stable performance estimates
* Recommendations for using rnCV in biomedical machine learning pipelines for reliable deployment

### **References**

* Cawley, G. C., & Talbot, N. L. C. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research, 11*, 2079–2107.
* Cruz, J. A., & Wishart, D. S. (2006). Applications of machine learning in cancer prediction and prognosis. *Cancer Informatics, 2*, 59–77. https://doi.org/10.1177/117693510600200030
* Krstajic, D., Buturovic, L. J., Leahy, D. E., & Thomas, S. (2014). Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of Cheminformatics, 6*(1), 10. https://doi.org/10.1186/1758-2946-6-10
* Siegel, R. L., Miller, K. D., Wagle, N. S., & Jemal, A. (2023). Cancer statistics, 2023. *CA: A Cancer Journal for Clinicians, 73*(1), 17–48. https://doi.org/10.3322/caac.21763
* Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics, 7*, 91. https://doi.org/10.1186/1471-2105-7-91
* Varoquaux, G., et al. (2017). Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines. *NeuroImage, 145*, 166–179. https://doi.org/10.1016/j.neuroimage.2016.10.038
* Wolberg, W. H., Street, W. N., Mangasarian, O. L. (1995). Machine learning techniques to diagnose breast cancer from fine-needle aspirates. *Cancer Letters, 77*(2-3), 163–171. https://doi.org/10.1016/0304-3835(94)03744-5

## **Materials and Methods:**

### Repeated Nested Cross-Validation Scheme



### Exploratory Data Analysis

#### Dataset Description

The breast cancer dataset used in this study was loaded from a CSV file and contains clinical and morphological features derived from digitized images of breast mass tissue samples. Each sample is labeled as either *Malignant* (M) or *Benign* (B), with a corresponding binary target variable (0 for Malignant, 1 for Benign). Dataset's shape, column structure were examined and performed basic integrity checks, including the identification of missing values and duplicate rows. Class distribution was inspected to detect any imbalance, as this could bias the performance of classification models. The dataset comprises **569 observations** across **33 columns**, including diagnosis, measurement features (mean, standard error, and worst).

#### Data Integrity and Class Balance

To assess data completeness, missing values were quantified and applied imputation strategie— SimpleImputer by mean value from scikit-learn—to ensure robustness in preprocessing and preserving the dataset's statistical properties. The dataset includes float types of numerical values. Class balance analysis showed **212 malignant** and **357 benign** samples, corresponding to a class ratio of approximately **59.3% Benign to 40.7% Malignant**, indicating a moderate imbalance. This distribution was visualized using a bar chart to highlight the need for stratified sampling during model training.

#### Feature Distribution

Histograms of the \*\_mean features (e.g., radius\_mean, texture\_mean) were plotted to examine the distributions of values for malignant and benign tumors. Kernel density estimation further revealed overlapping regions and potential non-linear separability.

#### Outlier Detection

Box plots of mean features grouped by diagnosis highlighted the presence of several extreme values. A standard deviation–based outlier detection method (3σ rule) was applied, identifying multiple features with substantial outlier counts.

#### Correlation Analysis

To assess redundancy among features, a Pearson correlation matrix was computed for all numerical variables. A heatmap visualization revealed strong multicollinearity among size-related features with correlation coefficients exceeding 0.9.

#### Dimensionality Reduction using PCA

To better understand the intrinsic structure of the data, Principal Component Analysis (PCA) was applied after standardizing the features. A two-component PCA projection was visualized, showing clear, though not perfectly linear, separation between benign and malignant classes. The first two principal components captured **approximately 63%** of the total variance. Further analysis showed that **10 principal components** were sufficient to retain **95%** of the variance in the data, justifying their use in downstream modeling as a way to reduce dimensionality while preserving most of the relevant information.

### **Preprocessing and Feature Engineering**

Prior to model training, several preprocessing steps were applied to ensure data quality and compatibility with machine learning algorithms. As described in the exploratory analysis, the dataset was confirmed to be free of missing values, with all samples having complete records. The target variable was derived by encoding the diagnosis column as a binary indicator: *Malignant* (0) and *Benign* (1).

For algorithms sensitive to feature scaling (e.g., logistic regression, SVM), standardization was performed using StandardScaler from Scikit-learn, which transforms features to have zero mean and unit variance. This step was embedded within pipeline structures to prevent data leakage during cross-validation. Highly correlated features identified during EDA were retained initially, with the expectation that regularization and feature importance analysis would implicitly address redundancy.

Dimensionality reduction using Principal Component Analysis (PCA) was explored in the EDA section but not applied to the final training pipeline to preserve interpretability of the models.

### 

### **Repeated Nested Cross-Validation (rnCV) Pipeline**

To ensure robust evaluation and fair model selection, a **repeated nested cross-validation (rnCV)** strategy was employed. This method partitions the data into an *outer* loop for estimating generalization error and an *inner* loop for hyperparameter tuning, thereby preventing information leakage and overly optimistic performance estimates [3, 5].

#### Structure and Parameters

* **Outer CV**: 5-fold stratified cross-validation, repeated 10 times (50 evaluations total).
* **Inner CV**: 3-fold stratified cross-validation for hyperparameter search.
* **Scoring Metric**: Area Under the ROC Curve (ROC-AUC) used for hyperparameter selection.
* **Selection Method**: The best parameter set was chosen using the *stability criterion*, maximizing mean performance minus standard deviation across inner folds.
* **Parallelization**: All grid searches were executed in parallel using n\_jobs = -1 for efficiency.

#### **Classifiers and Hyperparameter Grids**

Six diverse classification models were evaluated:

* **Logistic Regression (ElasticNet penalty)**: Regularization strength (C) and L1-ratio (l1\_ratio) tuned.
* **Gaussian Naive Bayes**: Smoothing parameter (var\_smoothing) tuned.
* **Linear Discriminant Analysis (LDA)**: Shrinkage factor explored with lsqr solver.
* **Support Vector Classifier (SVC)**: Kernel (rbf and linear), regularization (C), and kernel coefficient (gamma) tuned.
* **Random Forest**: Number of estimators and maximum tree depth optimized.
* **LightGBM**: Boosting iterations (n\_estimators), leaf count, and learning rate tuned.

All pipelines were wrapped using Pipeline objects to incorporate preprocessing directly into the model training phase and prevent data leakage during cross-validation.

#### **Performance Metrics**

Multiple classification performance metrics were computed for each fold of the outer cross-validation loop:

* **Standard metrics**: ROC-AUC, F1-score, F2-score, Balanced Accuracy, Matthews Correlation Coefficient (MCC), Recall, Precision, Average Precision (PR-AUC)
* **Additional metrics**: Specificity (true negative rate), Negative Predictive Value (NPV)  
   Metrics were aggregated across repetitions and folds, and their **medians** with **95% bootstrap confidence intervals (CI)** were computed using 1,000 resampling iterations to assess variability and statistical significance.

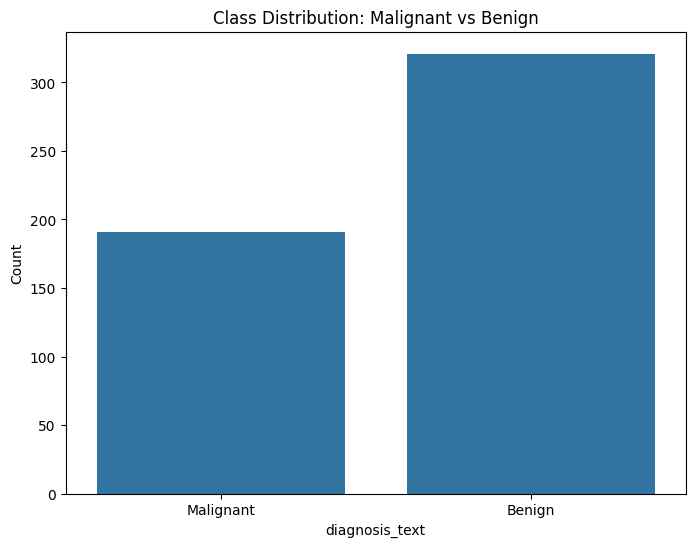
#### **Model Selection**

The best-performing model was determined based on a **multi-metric selection scheme**, prioritizing ROC-AUC followed by MCC. When confidence intervals between top models overlapped significantly, tie-breaking was performed based on aggregate performance across both metrics.

## **Results and Discussion : Exploratory Data Analysis**

### Dataset Overview

The dataset consisted of **512 samples**, each annotated with **30 numerical features** representing cellular and morphological characteristics of breast tumors. The target variable indicated whether a tumor was *Benign* (coded as 1) or *Malignant* (coded as 0). The class distribution was moderately imbalanced, with **321 benign cases** and **191 malignant cases**, resulting in a class ratio of approximately **59.5% benign** (*Figure 1*)



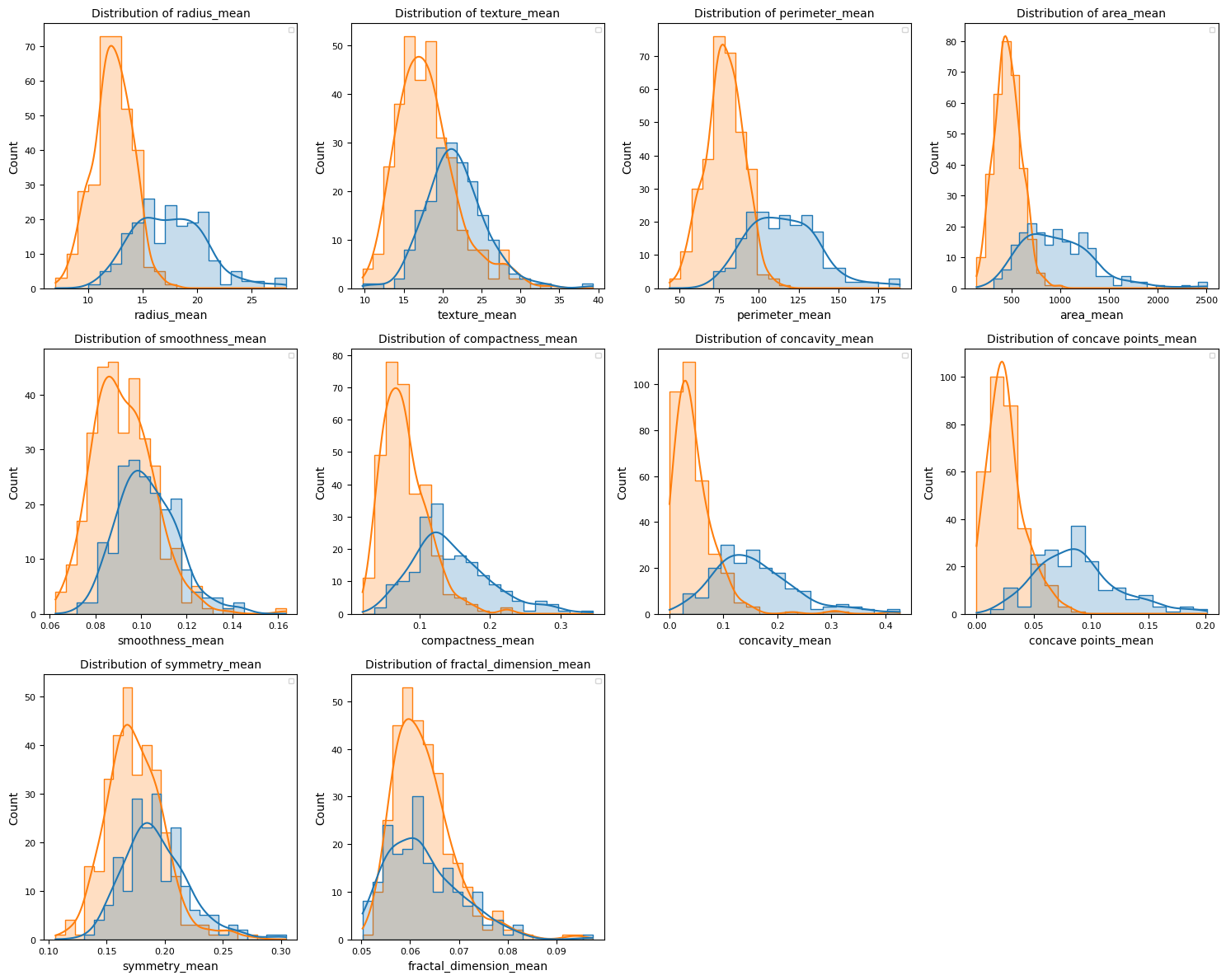
*Figure 1: Class imbalance.*

There were **no duplicate rows**, and **no missing values** in the target or label columns. However, several feature columns contained missing values, with up to **11 missing entries** observed in concave points\_worst. Features such as radius\_mean, area\_mean, and concavity\_mean had 4–10 missing entries each. These were handled using mean imputation, ensuring a complete dataset for further analysis.

### Descriptive Statistics and Distribution

Descriptive statistics showed considerable variability in feature ranges and scales. For instance, radius\_mean ranged from 6.98 to 28.11, and area\_mean from 143.5 to 2501.0, suggesting the need for standardization prior to modeling. The feature concavity\_worst displayed particularly high variance, with a standard deviation of **0.2097**, reflecting potential heterogeneity in tumor morphology.

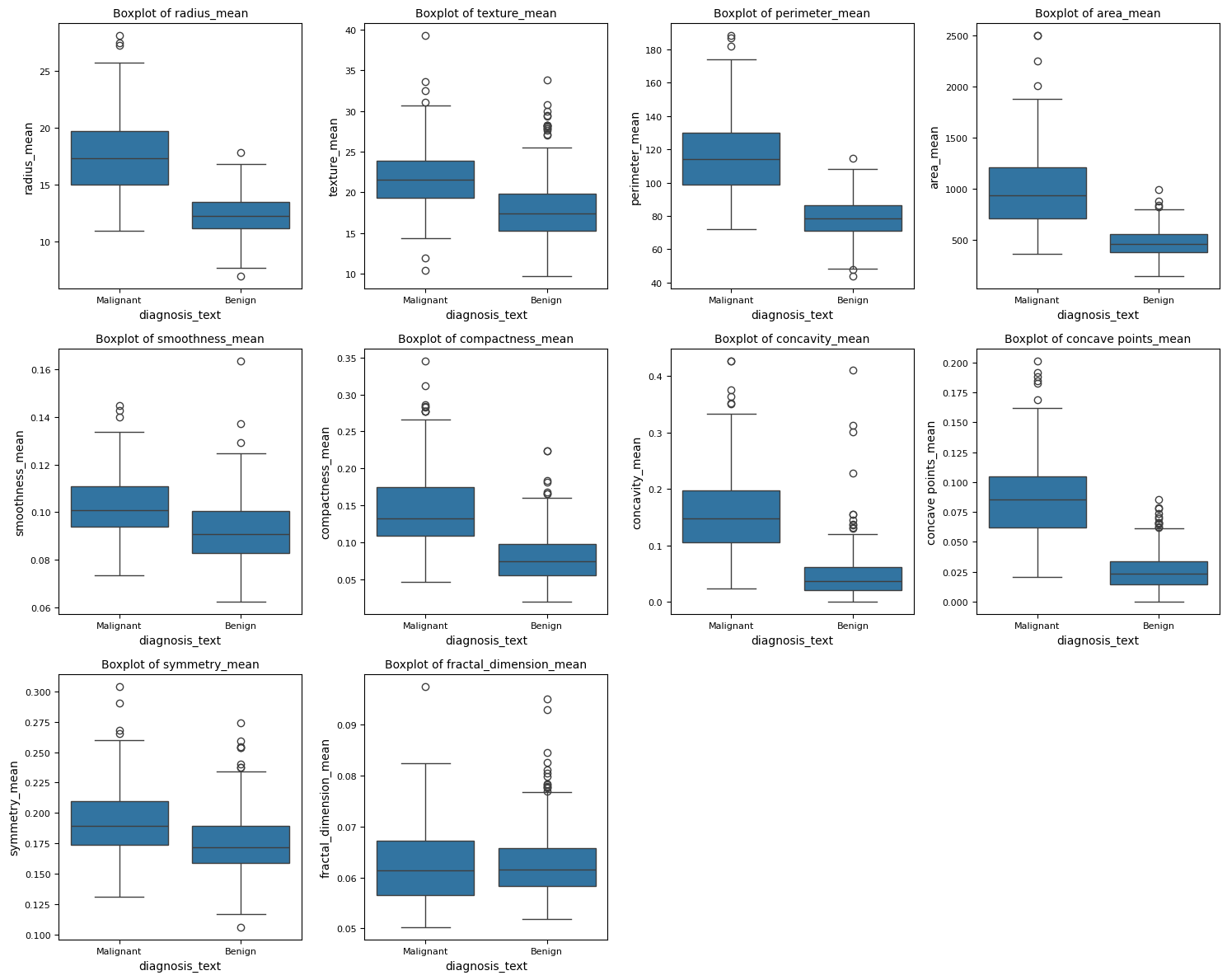
Histogram plots for the \_mean features revealed distinct distributions between malignant and benign tumors. Malignant tumors generally exhibited **higher values** in features related to size and irregularity—such as radius\_mean, perimeter\_mean, and concave points\_mean. In contrast, benign tumors tended to have **lower values** in textural and structural features, such as smoothness\_se, texture\_se, and fractal\_dimension\_mean (Figure 2).



*Figure 2: Histogram plots for the \_mean features between malignant and benign tumors*

### Outlier Detection

Outlier analysis using the 3-sigma rule identified multiple extreme values across nearly all features. Notably, features like concave points\_worst, compactness\_se, and fractal\_dimension\_se showed a relatively high number of outliers (≥9). These outliers reflect inherent biological variability rather than data entry errors and were retained for downstream modeling, given the medical context (Figure 3).

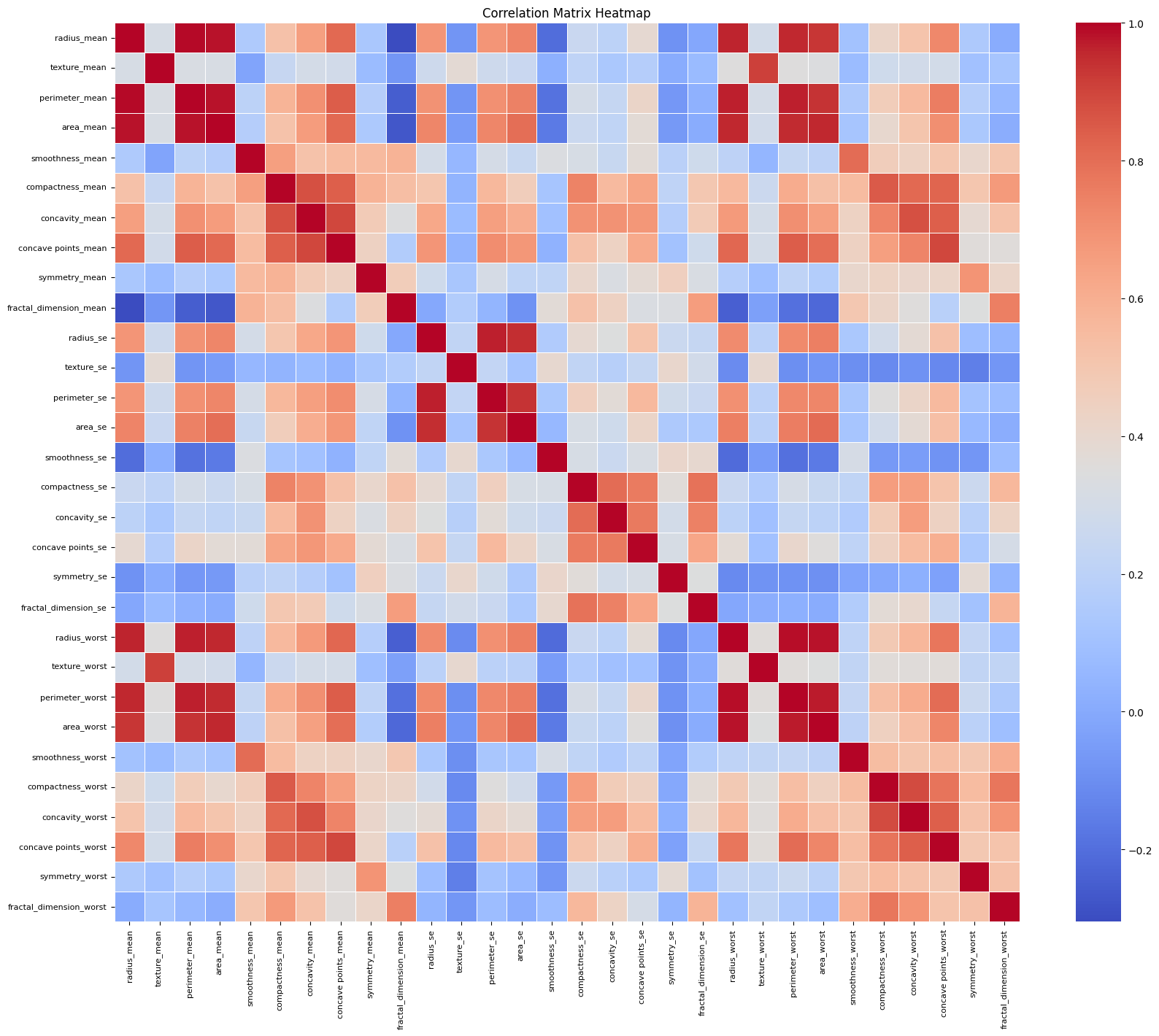


*Figure 3: Outlier analysis between features for malignant and benign tumors*

### Correlation Analysis

A Pearson correlation matrix identified **19 pairs of highly correlated features** with correlation coefficients exceeding **0.9**. These were mostly found among size-related features such as (Figure 4):

* radius\_mean, area\_mean, and perimeter\_mean (r > 0.98)
* radius\_worst and perimeter\_worst (r = 0.989)
* area\_se and radius\_se (r = 0.949)

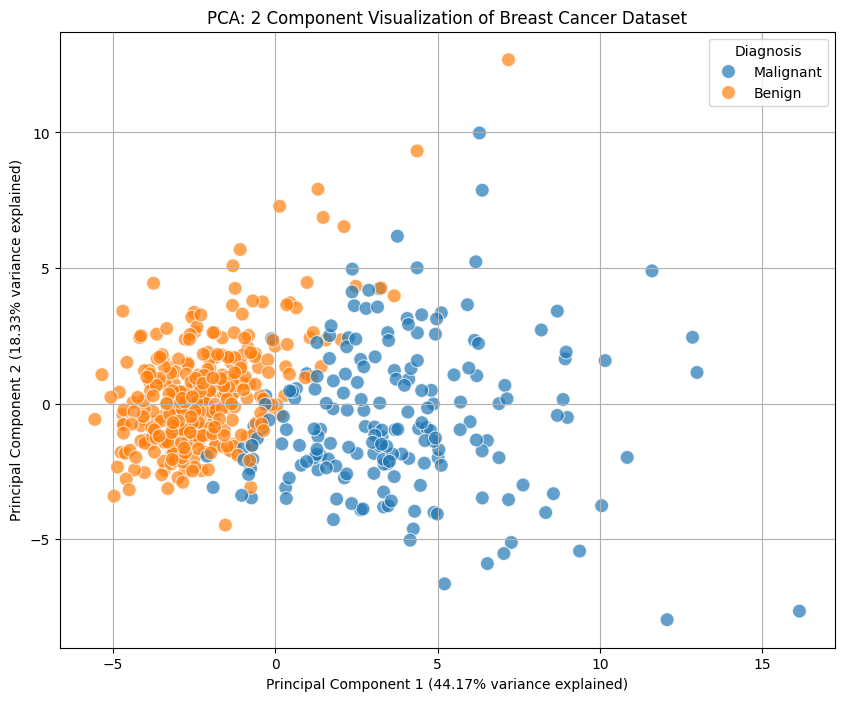


*Figure 4: Correlation Analysis between features*

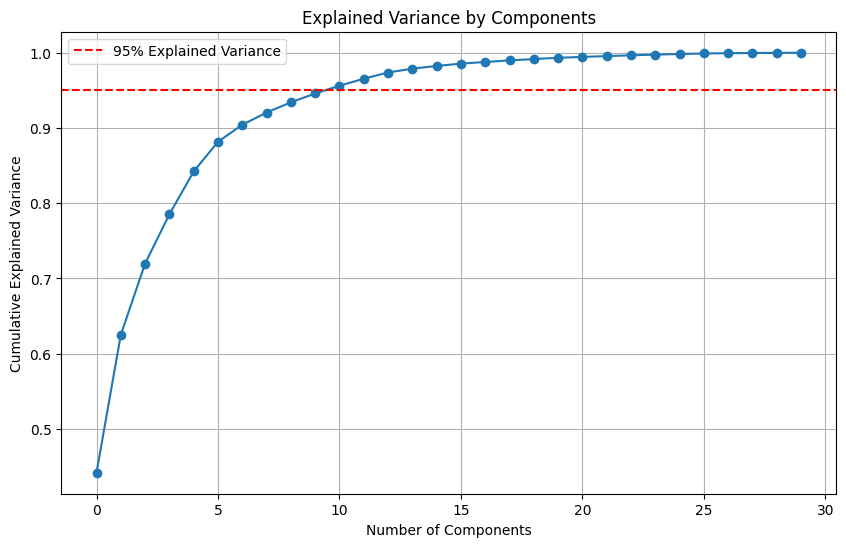
Such high multicollinearity suggests potential redundancy in the feature space. This finding motivates the use of either **feature selection techniques** or **dimensionality reduction methods** such as Principal Component Analysis (PCA).

### Principal Component Analysis (PCA)

To explore the latent structure of the data, PCA was performed on standardized features. The first two principal components accounted for approximately **63%** of the total variance and provided visible class separation in a 2D scatter plot (Figure 5). Importantly, **11 principal components** were sufficient to explain **95% of the variance**, indicating that a compact representation of the data is possible with minimal information loss (Figure 6).



*Figure 5: Scatter-plot of two principal components*



*Figure 5: Explained Variance by Components*

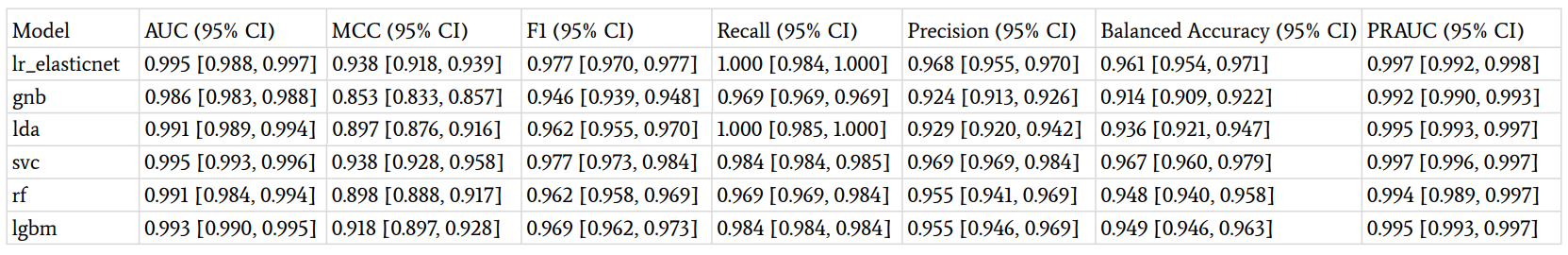
### Key Insights and Modeling Implications

* **Data Quality**: The dataset is clean and mostly complete, with missing values appropriately imputed and no duplicates.
* **Class Balance**: Slight imbalance suggests the need for **stratified cross-validation** and possibly **recall-focused evaluation** to avoid bias toward the majority class.
* **Feature Redundancy**: The presence of highly correlated features supports dimensionality reduction or regularization-based models.
* **Feature Relevance**: Features such as radius\_worst, perimeter\_worst, and concave points\_worst are visually and statistically distinct between classes, indicating their potential predictive value.
* **Visualization**: PCA confirms separability of classes, justifying its possible use in exploratory modeling or feature transformation pipelines.

## **Results and Discussion: Algorithm Comparison**

### Overview of Performance Metrics

To evaluate classification performance, six machine learning models were trained and assessed using repeated nested cross-validation. The evaluation focused on median performance scores along with 95% confidence intervals (CIs) derived from bootstrapping. Key metrics included AUC, MCC, F1-score, recall, precision, balanced accuracy, and PR-AUC.



*Table 1: Performance metrics*

### Comparative Analysis

* **Top-performing models**:  
   The **Support Vector Classifier (SVC)** and **ElasticNet-regularized Logistic Regression (lr\_elasticnet)** achieved the highest performance across nearly all metrics. Both had an AUC of **0.995**, but SVC was declared the **winner** based on a slightly higher and more stable MCC (0.938, CI: [0.928, 0.958]) and broader CI overlap margins in favor of SVC.
* **Precision and Recall Trade-off**:  
   All top models exhibited high recall (≥0.984), critical for medical diagnosis. ElasticNet and LDA achieved perfect median recall (1.000), but SVC balanced it with better specificity and precision, ensuring fewer false positives.
* **Naive Bayes Performance**:  
   The Gaussian Naive Bayes classifier (gnb), while computationally efficient, underperformed relative to other models, particularly in MCC (0.853) and balanced accuracy (0.914). This suggests that its strong independence assumption may not suit the correlated feature space in this dataset.
* **Tree-based Models**:  
   Random Forest (rf) and LightGBM (lgbm) achieved high AUCs (0.991 and 0.993, respectively), with robust F1 and PR-AUC scores. However, their MCC and balanced accuracy were slightly lower than SVC, indicating occasional misclassifications, possibly due to overfitting on certain folds.
* **LDA as a strong linear baseline**:  
   Linear Discriminant Analysis performed surprisingly well with AUC = 0.991 and perfect recall. This reinforces the idea that even simple linear models can excel when the data is well-separated, especially with engineered or domain-specific features.

### Practical Implications

* **SVC as the optimal choice**:  
   Given its top-ranking AUC and MCC, and balanced performance across precision, recall, and PR-AUC, the SVC model is recommended for further validation or deployment.
* **Confidence Intervals as decision aids**:  
   Incorporating confidence intervals provided a robust statistical basis for model comparison, highlighting not just point estimates but also the reliability of performance metrics. This is especially valuable in biomedical applications, where small differences in metrics may translate to large impacts on patient outcomes.
* **Model selection beyond accuracy**:  
   This analysis emphasized **MCC** and **AUC** as primary metrics, suitable for imbalanced datasets and binary classification with asymmetric costs of errors. The use of F2-score and NPV further aligned the evaluation with the clinical importance of minimizing false negatives.

### Results and Discussion: Winner Model Tuning: SVC Hyperparameter Optimization

To explore the full potential of Support Vector Machines (SVMs), a dedicated **SVC hyperparameter tuning pipeline** was developed using a class named SVCParameterTuner. This method applies **5-fold cross-validation** with GridSearchCV to optimize a broad set of parameters for the SVC classifier, including:

* **Regularization strength (C)**: [0.01, 0.1, 1, 10, 100]
* **Kernel type**: ['rbf', 'linear', 'poly']
* **Gamma (kernel coefficient)**: ['scale', 'auto', 0.001, 0.01, 0.1]
* **Degree**: [2, 3, 4] (applicable to polynomial kernel)
* **Coef0**: [0.0, 0.1, 0.5] (used for polynomial kernel)

This pipeline automatically identifies the best configuration based on **cross-validated accuracy** and returns the optimal SVC model trained on the entire dataset. The final model is serialized using pickle and saved asnwinner.pkl for downstream usage or deployment.

This standalone grid search complements the broader repeated nested CV framework by allowing **detailed single-model optimization**, especially useful when a model like SVC performs competitively across multiple metrics in the rnCV setup.

### References

[1] G. C. Cawley and N. L. C. Talbot, “On over-fitting in model selection and subsequent selection bias in performance evaluation,” *J. Mach. Learn. Res.*, vol. 11, pp. 2079–2107, 2010.

[2] J. A. Cruz and D. S. Wishart, “Applications of machine learning in cancer prediction and prognosis,” *Cancer Inform.*, vol. 2, pp. 59–77, 2006, doi: 10.1177/117693510600200030.

[3] D. Krstajic, L. J. Buturovic, D. E. Leahy, and S. Thomas, “Cross-validation pitfalls when selecting and assessing regression and classification models,” *J. Cheminform.*, vol. 6, no. 1, p. 10, 2014, doi: 10.1186/1758-2946-6-10.

[4] R. L. Siegel, K. D. Miller, N. S. Wagle, and A. Jemal, “Cancer statistics, 2023,” *CA Cancer J. Clin.*, vol. 73, no. 1, pp. 17–48, 2023, doi: 10.3322/caac.21763.

[5] S. Varma and R. Simon, “Bias in error estimation when using cross-validation for model selection,” *BMC Bioinformatics*, vol. 7, p. 91, 2006, doi: 10.1186/1471-2105-7-91.

[6] G. Varoquaux *et al.*, “Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines,” *NeuroImage*, vol. 145, pp. 166–179, 2017, doi: 10.1016/j.neuroimage.2016.10.038.

[7] W. H. Wolberg, W. N. Street, and O. L. Mangasarian, “Machine learning techniques to diagnose breast cancer from fine-needle aspirates,” *Cancer Lett.*, vol. 77, no. 2–3, pp. 163–171, 1995, doi: 10.1016/0304-3835(94)03744-5.

### Disclosure of AI Assistance

Portions of this work were assisted by large language models (LLMs) to enhance productivity, accuracy, and clarity. Specifically, the model **ChatGPT-4 (gpt-4o)** was used to support academic writing tasks, including the structuring of sections, refining technical language, and improving overall readability of the report. The smaller model variant **04-high-mini** was used primarily for **code validation, debugging, and improving code readability** during the implementation phase. All outputs were critically reviewed and verified by the author to ensure correctness, originality, and alignment with academic standards.