Data Preprocessing Methods with Mathematics Behind and Applications on Early Cancer Detection

Math 490 - Graduation Project

2024-2025 Spring Semester

Author
Mert BURSALIOĞLU

Supervisor
Assoc. Prof. Dr. Özlem DEFTERLİ

Department of Mathematics

Çankaya University

ABSTRACT

Data preprocessing is crucial before initiating machine learning and model creation steps. Raw datasets frequently consist of outliers, unnecessary features, and unscaled variables. This study contains the mathematical background and implementation of key data preprocessing techniques, such as outlier treatment, feature scaling, multicollinearity elimination, and feature selection. In order to assess the effectiveness of these techniques on model performance, a comparison-based evaluation was carried out. Applying them to real-world medical data indicates why appropriately operated preprocessing is necessary.

Keywords: data preprocessing, outlier treatment, feature scaling, multicollinearity, feature selection, machine learning

Contents

1 Introduction **1**

1.1 **Motivation** . . . 3

1.2 **Objectives** . . . 3

1.3 **Contributions** . . . 4

1.4 **Structure** . . . 4

2 Data Preprocessing Methods **6**

2.1 Missing Data Treatment . . . 6

2.2 Outlier Treatment . . . 7

2.3 Feature Scaling . . . 8

2.4 Multicollinearity Treatment . . . 10

2.5 Feature Encoding . . . 11

2.6 Feature Selection . . . 12

3 Mathematical Background of Preprocessing **14**

3.1 Missing Data Treatment . . . 14

3.1.1 Mean Imputation . . . 14

3.1.2 Median Imputation . . . 15

3.1.3 Mode Imputation . . . 15

3.2 Outlier Treatment . . . 15

3.2.1 IQR-based Winsorization . . . 15

3.3 Feature Scaling . . . 16

3.3.1 Z-score Standardization . . . 16

3.3.2 MinMax Normalization . . . 16

3.3.3 Robust Scaling . . . 17

3.4 Feature Selection . . . 18

3.4.1 Lasso Regression . . . 18

4 Conclusion **20**

List of Figures

1 Breast Cancer Death Rate in Women 2
2 Visual illustration of Tukey's approach for spotting outliers using a boxplot. 7
3 Confusion matrix before preprocessing and logistic regression modeling 21
4 Confusion matrix after preprocessing and logistic regression modeling 21

1 Introduction

Data quality expresses how appropriately a dataset fulfills the specific needs of its application. This concept is shaped by elements like accuracy, completeness, internal consistency, timeliness, reliability, and how easily the data can be understood, which refer to the condition of a dataset. In real-world applications, the raw data does not satisfy these quality standards. Raw data may contain missing information, inaccurate values, contradictory formats, and unnecessary entries for diverse reasons, such as human, environmental, and instrument-related errors.

Using low-quality data has serious consequences for data analysis and machine learning. Models created by noisy and inconsistent data are more likely to produce erroneous predictions. Data quality has significant importance in medical fields such as cancer malignancy detection. This project focuses on breast cancer data to illustrate how to apply preprocessing techniques and their influence on model quality and reliability.

Cancer is a worldwide medical issue that can dramatically shorten the lifespan of a human. It arises when cellular mechanisms responsible for regulating division and self-destruction malfunction, allowing defective cells to multiply uncontrollably and spread to adjacent areas. However, when a cell's genetic material is damaged or changed, it can prevent regular apoptosis(regular cell death). This can cause rapid and unregulated cell proliferation. These cells can turn into tumors, and when they colonize secondary locations within the body, it is called metastasis, which is the leading cause of death due to cancer.

Benign tumors usually stay in the place where they first develop and do not have the ability to spread to other parts of the body. They are not considered cancer since they do not move beyond their original area or invade nearby tissues. On the other hand, malignant tumors behave differently — they tend to grow quickly, push into nearby areas, and eventually reach distant parts of the body. This spreading ability is what makes them dangerous and defines them as cancerous.[1]

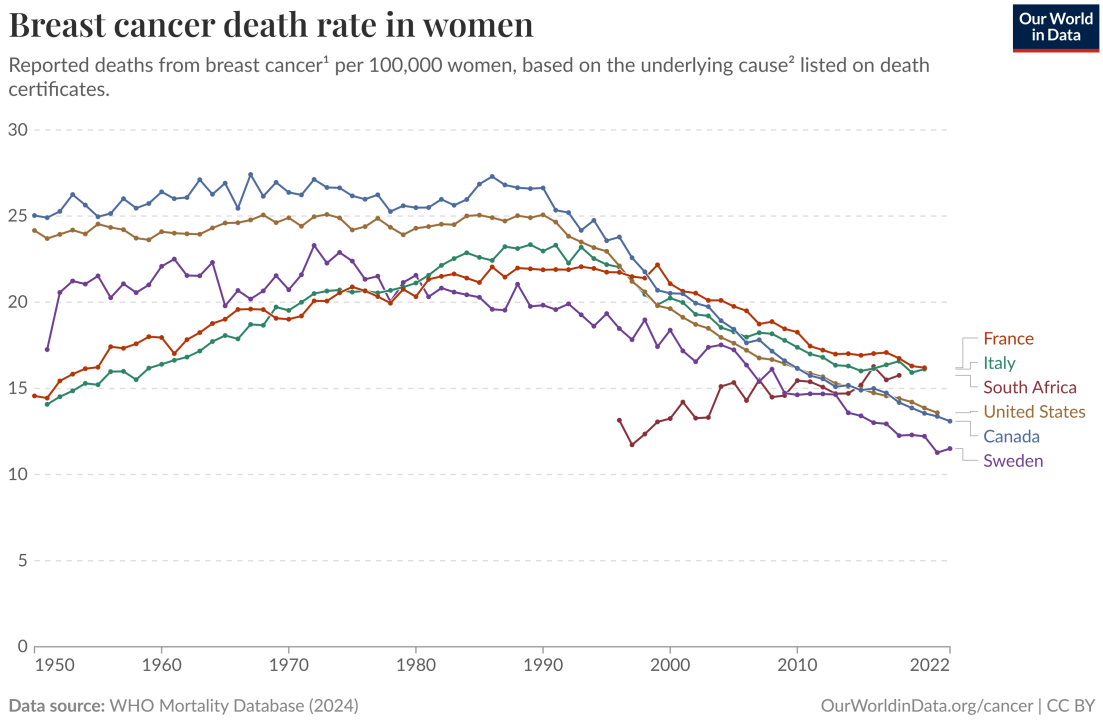

Figure 1: Breast Cancer Death Rate in Women

[2]

According to the World Health Organization, cancer was one of the primary causes of death worldwide in 2018, with different cancer types showing varying prevalence between genders. While men were more commonly diagnosed with lung and prostate cancers, women faced higher rates of breast and cervical cancers.[1]

Cancer is not supposed to be a death sentence. It progresses through four stages, ranging from mild to severe. So, early diagnosis can save lives. Biopsies are the traditional ways of gathering data about tumors. A biopsy is a medical procedure in which a sample—often a specimen—is taken from a suspicious lump, tumor, or area for laboratory examination. These samples, whether composed of cells or tissue, can be collected from nearly any body region and are commonly used to identify the presence of cancer.[3] However, tumors might be in an area that can comprise patient

health. In addition, tumors often spread non-homogeneously, so data collected from biopsies are limited to where biopics are applied.
Therefore, data preprocessing is key to improving the reliability and accuracy of biopsy analysis. It allows hidden patterns and important relationships in the data to be more easily identified.

1.1 Motivation

Raw data is non-usable without any process being applied to it. Since they contain missing values, noise, inconsistencies, and irrelevant features, making it harder to produce stable and trustworthy results, the model outputs should be precise since preprocessing techniques are used primarily in the medical field. The project aims to apply the preprocessing techniques to breast cancer data and comprehend the importance of preprocessing. Since cancer is a deathly and frequently encountered disease, there is a lower chance of survival when cancer treatment is delayed. However, preprocessing techniques can diagnose patients early, lowering care expenses and extinguishing inequalities.

1.2 Objectives

This project aims to explain the data preprocessing strategies, explain how these techniques deal with imbalanced data, and maximize the utility of obtained data for creating a prediction to characterize cancer.Moreover, the mathematical background of the preprocessing strategies will be mentioned. These will be applied to the Wisconsin Diagnostic Breast Cancer (WDBC) Dataset to comprehend sufficiently the role of mathematics in preprocessing techniques and their impact on cancer detection, which will be beneficial. This study will demonstrate the effects of preprocessing techniques on model accuracy and reliability.

1.3 Contributions

The contribution of this study:

- Describe the data pre-processing techniques that can substantially enhance the model accuracy.
- Describe the mathematical background of data preprocessing techniques.
- Benchmark their performance on real-world breast cancer data.

1.4 Structure

This document is divided into five sections. The present section introduces the project's motivation, objectives, and contributions. Section 2 describes a comprehensive overview of data preprocessing techniques, detailing their functionality, purpose, and significance. This chapter will address challenges such as noise, missing values, unscaled features, and so on. Section 3 explains the mathematical background of a preprocessing technique, focusing on its theoretical foundations. Section 4 presents the application of explained data preprocessing techniques on a Wisconsin Diagnostic Breast Cancer (WDBC) Dataset to create a stronger understanding, including a benchmark of their performance. Section 5 provides a conclusion regarding the work is concluded in this project.

2 Data Preprocessing Methods

2.1 Missing Data Treatment

For various reasons, missing data occurs frequently in data science, which refers to the absence of recorded features within the observation set. Before starting the treatment of missing entries, one should understand why they occurred. There are three main types of missing data: (1) Missing Completely at Random (MCAR), (2) Missing at Random (MAR), and (3) Missing Not at Random (MNAR).[14]

1. **Missing Completely at Random(MCAR):** It refers to a type of missingness where the chance of a value being missing is entirely unrelated to any part of the data, whether seen or unseen. In other words, the available data can be considered as a simple random sample of the entire dataset.[5] This mechanism is the least problematic trouble since the missingness is not due to structural reasons. However, it is not commonly seen in real-life data.

2. **Missing at Random(MAR):** It refers to cases where missing values in a variable depend on other observed variables but not the missing variables. After accounting for related variables, the missingness becomes unrelated to the outcome. For example, if candidates with low IQs tend to have missing performance ratings, regardless of actual performance, it is considered MAR.[5]. This mechanism is the most frequently seen missingness type in applied research since the missingness is related to known variables in the dataset.

3. **Missing Not at Random(MNAR):** It describes cases where the likelihood of a value being missing is directly influenced by the value itself, which remains unobserved. For example, if low-performing employees are more likely to have missed evaluations due to early termination, missingness is directly related to performance. [5]This mechanism is the most complicated among the types of mechanisms. Since the missingness occurs due to unknown values, it is challenging to address it statistically.

2.2 Outlier Treatment

Sometimes, a dataset contains a few values that stand out — they do not quite match what is typical for the rest of the data. These outliers can happen because of mistakes in collecting data, or they might simply reflect unusual but real events. If these values are not managed properly, they might throw the model off and lead to results that are not very reliable. There are several methods available for outlier detection, including visualization-based techniques such as boxplots and scatter plots, statistical approaches like Z-scores, Grubbs' test, and Tukey's approach, as well as machine learning-based methods like Local Outlier Factor One-Class SVM, and Isolation Forest.

Visualization-based outlier detection is one of the most intuitive detection techniques. It lets the analyst rapidly view the data's patterns, trends, and anomalies. One standard visualization tool for this purpose is the boxplot, which summarizes the distribution using five points — the smallest and largest values, the middle value (median), and the lower and upper quartiles. Figure 2 also helps identify outliers by applying Tukey's method, which marks any value as an outlier if it falls well below or far above the expected range based on the interquartile spread.[10, 9].

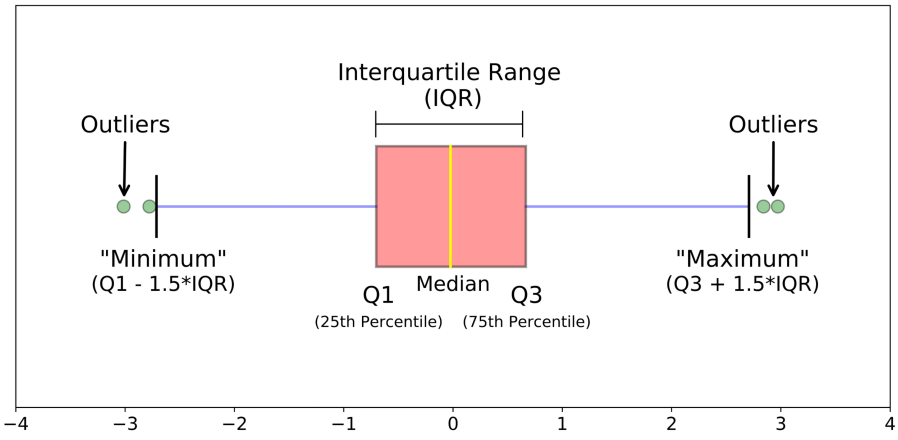

Figure 2: Visual illustration of Tukey's approach for spotting outliers using a boxplot.

Outliers can broadly be categorized into the following types:

- **Point Outliers:** A single observation that shows a substantial deviation from the overall pattern within the dataset.
- **Contextual Outliers:** An unusual observation when evaluated under specific conditions, such as temporal or spatial settings.
- **Collective Outliers:** A group of related entries that together form a pattern inconsistent with the dataset's general structure.

[11]

2.3 Feature Scaling

In machine learning applications, datasets generally contain multiple numerical attributes with differing scales and units. The model can assign too much importance to some features in this situation, especially in length- and magnitude-sensitive algorithms. For instance, a feature takes values between 0 and 1. At the same time, another may range between 0.0001 and 0.1, so the first feature will dominate the model, leading the model to generate unbalanced and unrelatable outcomes. Feature scaling transforms features to similar scale intervals to overcome this issue. In this manner, the algorithm gives equivalent importance to the features. Common strategies for scaling include techniques that normalize data based on its z-score, compress it into a fixed interval such as [0,1], or reduce the impact of extreme values.

- **StandartScaler:** In this approach, the values of each feature are shifted so that their mean becomes zero and then rescaled to reflect unit variance. This ensures that all variables contribute proportionally to models sensitive to feature magnitudes, such as those using Euclidean distance. However, when datasets contain extreme values, this method can become less effective due to the sensitivity of mean and variance to outliers. [6]

- **MinMaxScaler:** This method transforms features by mapping their values into a fixed interval—typically between 0 and 1—based on their minimum and maximum observed values. It is especially suitable when the dataset lacks extreme deviations and has a uniform distribution. Nonetheless, even a few significantly high or low values can distort the scaling process. [6]

- **RobustScaler:** Rather than applying averages and standard deviations, this strategy uses the median and the interquartile range (IQR) to perform scaling. Because these statistical measures are resistant to the influence of anomalies, this method is more effective for datasets that contain outliers. The IQR captures the middle 50% of the data by measuring the distance between the first and third quartiles. Given the irregular nature of real-world data, this scaling method is often considered more stable and dependable in practice. [6, 7]

Table 1: Comparison of Common Feature Scaling Methods

Property	**StandardScaler**	**MinMaxScaler**	**RobustScaler**
Centers data?	Yes (mean = 0)	No	Yes (median = 0)
Scales to fixed range?	No	Yes (default: [0, 1])	No
Affected by outliers?	Yes	Yes	No (uses IQR)
Best for normal distribution?	Yes	No	No
Best for data with outliers?	No	No	Yes
Scaling method	$(x - \mu)/\sigma$	$(x - x_{\min})/(x_{\max} - x_{\min})$	$(x - \text{median})/\text{IQR}$

2.4 Multicollinearity Treatment

In regression contexts, multicollinearity refers to the situation where predictor variables show substantial linear association, which can result in unreliable coefficient estimates and inflated standard errors. As a result, the model cannot isolate the distinct contribution of each explanatory variable, which affects its outcome in overestimating the variables' statistical significance. [12]

One practical way to examine multicollinearity is by using the Variance Inflation Factor (VIF), which reflects how much the variance of a model coefficient increases due to relationships among predictors. A higher VIF value usually indicates stronger multicollinearity and may signal instability in coefficient estimates.

$$\text{VIF} = \frac{1}{1 - R^2} = \frac{1}{\text{Tolerance}}$$

Where R^2 represents how much of the predictor's variance is explained by the other predictors. VIF values near one indicate that predictor variables function independently without notable linear association. As the value increases beyond this point but stays below five, mild dependency may exist, which is typically manageable. When the figure approaches or surpasses five, particularly nearing ten, it often signals that multicollinearity is present at a level that could undermine the stability of the regression model. If the VIF exceeds ten, it reflects a serious multicollinearity problem and suggests that the regression coefficients may be estimated with low precision.[13]

2.5 Feature Encoding

Feature encoding denotes altering categorical variables into numerical values since machine learning algorithms cannot operate on non-numerical inputs. There are seven feature encoding techniques:

- **Ordinal Encoding:** Assigns numerical values to categories based on order. For example: Basic → 0, Standard → 1, Premium → 2
- **One-Hot Encoding:** This approach represents each category as a separate binary feature. For example: HR → [1, 0, 0], IT → [0, 1, 0], Finance → [0, 0, 1].
- **Sum Encoding:** Encodes categorical variables such that the sum of the coding vectors equals zero. For example: A→[0, 1],B→[1, 0],C→[-1, -1]
- **Helmert Encoding:** Every variable level is compared against the average of the remaining levels after it. For a three-category variable A, B, and C, A is compared to the average of B and C, and B is compared to C.
- **Polynomial Encoding:** It encodes levels based on polynomial functions (e.g., linear, quadratic) to reflect systematic changes in categorical values. For example, levels 1, 2, and 3 could be encoded with linear contrasts of -1, 0, and 1, respectively.
- **Backward Encoding:** Each level is encoded by measuring its deviation from a reference level, commonly the final category of the variable. To illustrate, A → [1, 0], B → [0, 1], C → [0, 0]
- **Binary Encoding:** Each category is first assigned an integer, which is then converted to its binary form.
  Node1 (1) → 001, Node2 (2) → 010, Node3 (3) → 011, Node4 (4) → 100 [15, 14]

2.6 Feature Selection

Real-life datasets, especially medical records, can have various features. However, all these features cannot contribute equally to the model output. Some may involve redundant, noisy, or irrelevant information. The models trained by such features cannot be reliable. Feature selection helps mitigate these problems by keeping the features that contribute the most to the model output to reduce overfitting and enhance interpretability. This problem can be tackled using five well-known approaches Subsequently, various feature selection methods can be applied, including approaches based on low variance elimination, univariate statistical testing, recursive model-based ranking, importance-weighted model selection, and stepwise feature evaluation.

- **Varience Threshold:** It eliminates the features showing minimal variation across samples. It supposes that the features with low interpretation contribute slightly to model performance. After selecting a predefined threshold, any feature falling below this variance level is considered uninformative and removed from the dataset.

- **Select K Best:** It applies a univariate statistical test to rate separate features based on relevance, considering each feature acts independently. The method scores each feature individually and identifies and retains the k most relevant features based on their individual statistical contributions.

- **Recursive Feature Elimination:** It works by training a model using all known features, rating them according to their applicability, and then continually removing the least significant ones until a specific number of features remains.

- **Select From Model:** It performs feature selection utilizing an external estimator's importance scores or coefficients. It presents a straightforward approach by filtering out features by applying a threshold, where those falling below the set value are excluded from the dataset.

- **Sequential Feature Selector:** The Sequential Feature Selector (SFS) gradually adds or extracts features based on how well a chosen machine learning model performs. At each iteration, it evaluates subsets of features by updating the selection and measuring performance, eventually identifying the optimal feature set.

[16, 17]

3 Mathematical Background of Preprocessing

Each method used in data preprocessing is not just an algorithm. These processes are based on mathematical principles. It is crucial since understanding why is more important than how before undertaking preprocessing. In favor of mathematical insight, one can make reliable and precise decisions on when and how to implement these methodologies. Methods like Z-score, IQR, and VIF are intrinsically linked to core mathematical ideas such as distribution properties, range measures, and multicollinearity. In addition, a robust mathematical understanding will improve the data analysts' decision-making and help them make deeper analyses, improving the accuracy of the models created. Therefore, the practical and theoretical aspects of preprocessing methods are addressed in this study.

3.1 Missing Data Treatment

Notation

Let the dataset be denoted as a matrix $X = [x_i^{(j)}]$, with the following definitions:

- $\hat{x}_i^{(j)}$: estimated replacement for the missing entry located at index (i, j)
- $X_{\text{obs}}^{(j)}$: collection of available (non-missing) values in the j^{th} feature
- n_j: total count of known values within column j

3.1.1 Mean Imputation

$$\hat{x}_i^{(j)} = \mu_j = \frac{1}{n_j} \sum_{r=1}^{n_j} x_r^{(j)}$$

where μ_j is the arithmetic mean of the observed values. [18]

3.1.2 Median Imputation

$$\hat{x}_i^{(j)} = \text{median}(X_{\text{obs}}^{(j)})$$

The missing value is replaced by the median of the observed values. [18]

3.1.3 Mode Imputation

$$\hat{x}_i^{(j)} = \text{mode}(X_{\text{obs}}^{(j)})$$

Missing entries in categorical variables can be filled by assigning the most commonly occurring category in the data. [18]

3.2 Outlier Treatment

3.2.1 IQR-based Winsorization

This approach replaces fixed percentiles with outlier thresholds derived from the Interquartile Range (IQR). To handle extreme values in each feature, observations that fall significantly outside the typical range—defined as more than 1.5 times the interquartile spread beyond the lower or upper quartiles—are modified. This process, known as Winsorization, restricts such values to the nearest acceptable boundary:

$$\tilde{x}_i = \begin{cases} Q_1 - 1.5 \cdot \text{IQR} & \text{if } x_i < Q_1 - 1.5 \cdot \text{IQR} \\ x_i & \text{if } Q_1 - 1.5 \cdot \text{IQR} \leq x_i \leq Q_3 + 1.5 \cdot \text{IQR} \\ Q_3 + 1.5 \cdot \text{IQR} & \text{if } x_i > Q_3 + 1.5 \cdot \text{IQR} \end{cases}$$

Where:

- Q_1: the lower quartile, indicating the value below which roughly one-fourth of the data falls
- Q_3: the upper quartile, capturing the point above which the highest 25% of values are located

- IQR $= Q_3 - Q_1$: a measure of central dispersion, reflecting the spread of the central half of the distribution
- x_i: the original measurement observed in the dataset
- $\tilde{x}_i$: the final value after boundary-based adjustment to mitigate outliers

3.3 Feature Scaling

3.3.1 Z-score Standardization

This scaling method adjusts values so that the transformed distribution is centered at zero with a standard deviation of one. Each data point is rescaled relative to the feature's average and variability:

$$x'_i = \frac{x_i - \mu}{\sigma}$$

Notation:

- x_i: the unscaled value
- x'_i: the resulting standardized score
- μ: the expected value of the corresponding variable
- σ: the spread of the feature's values (standard deviation)

[19]

3.3.2 MinMax Normalization

This method adjusts the scale of features so that their values fall within a defined interval, commonly from 0 to 1.

$$x'_i = \frac{x_i - \min(X)}{\max(X) - \min(X)}$$

Where:

- x_i: original value
- x'_i: normalized value
- $\min(X)$: smallest observed value in the feature
- $\max(X)$: largest recorded value for that feature

[19]

3.3.3 Robust Scaling

In situations where datasets include extreme values, it is often useful to apply a scaling method that resists distortion caused by such anomalies. Robust scaling achieves this by realigning the data around its middle point, without being heavily influenced by unusually high or low values. The transformation is based on the distance between two internal cut-off points rather than the overall spread.

$$x'_i = \frac{x_i - \text{median}(X)}{\text{IQR}(X)}$$

Where:

- x_i: original value
- x'_i: scaled value
- $\text{median}(X)$: the central observation in the sorted version of the data
- $\text{IQR}(X)$: interquartile range

[19]

3.4 Feature Selection

3.4.1 Lasso Regression

This regression method applies a regularization strategy that not only prevents overfitting but also acts as a feature selector by shrinking some parameter estimates to zero. This approach is particularly advantageous when dealing with a large number of predictors or in the presence of multicollinearity.

From an optimization perspective, Lasso seeks a solution that minimizes prediction error while applying a constraint on the total absolute value of the coefficients.

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

where:

- y_i is the response variable,
- x_i is the predictor vector for the i-th observation,
- β is the vector of regression coefficients,
- $\lambda \geq 0$ is a tuning parameter controlling the strength of the penalty.

The penalty term $\sum |\beta_j|$ is the L1-norm of the coefficients. When λ is large, more coefficients are driven to exactly zero, effectively performing variable selection.[20]

4 Conclusion

This study presented a comprehensive data preprocessing pipeline to improve classification performance on a breast cancer dataset. The steps included exploratory data analysis (EDA), outlier detection and treatment using IQR-based Winsorization, feature scaling with `RobustScaler`, multicollinearity elimination using VIF analysis ,and statistical filtering via `SelectKBest` were applied as part of the feature selection strategy. These stages are used to enhance the model's reliability and robustness.

Preprocessing showed its impact frankly, especially on boxplots, which visualized the significant outliers in a feature like `area_mean`. Winsorization capped these outliers to stabilize feature distributions. Then, by feature scaling, data ranges are normalized, and the influence of extreme values is mitigated. These ensured an even contribution of each feature to make a model assign fair importance to them.

To handle multicollinearity, features with high VIF values (above 10) were iteratively removed. This helped reduce redundancy among predictors and enhanced the model's generalization ability. Feature selection with `SelectKBest` was applied on the cleaned dataset, selecting the top-performing features based on ANOVA F-scores. To refine the feature set, variables showing high correlation were eliminated based on VIF scores, while `SelectKBest` was used to retain the most informative predictors for different k values (5, 10, 15), and $k = 5$ yielded the best trade-off between simplicity and performance.

Using the five most relevant variables, a logistic regression model was constructed. The outcome of the evaluation revealed measurable gains in predictive accuracy.

The trained algorithm trained on raw data achieved only 95.6% accuracy, with four false negatives and one false positive. These errors are particularly critical in medical applications, as false negatives may result in undetected malignant cases.

In contrast, the confusion matrix of the preprocessed model demonstrated almost perfect prediction, with only one false negative and one false positive among 114 samples. This was a considerable enhancement over the unprocessed model. With an accuracy of 99.1% and an F1-score of 0.99, the resulting classification system validated the success of the preprocessing strategy.

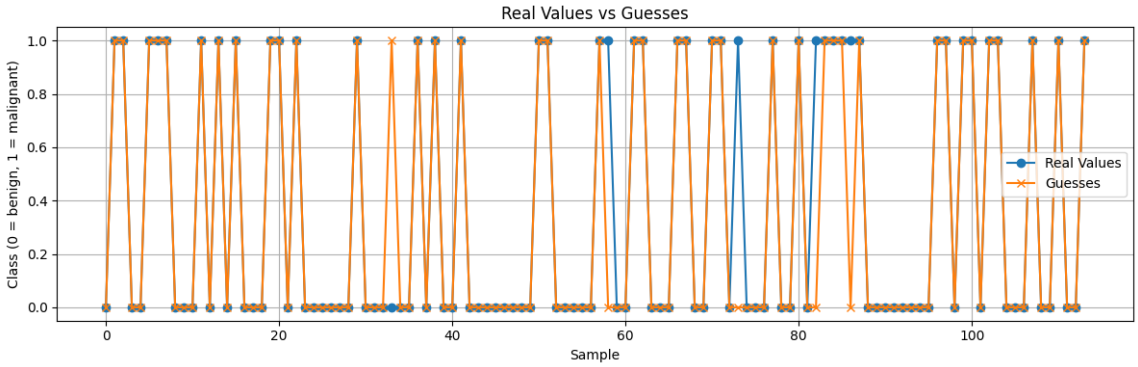

Figure 3: Confusion matrix before preprocessing and logistic regression modeling

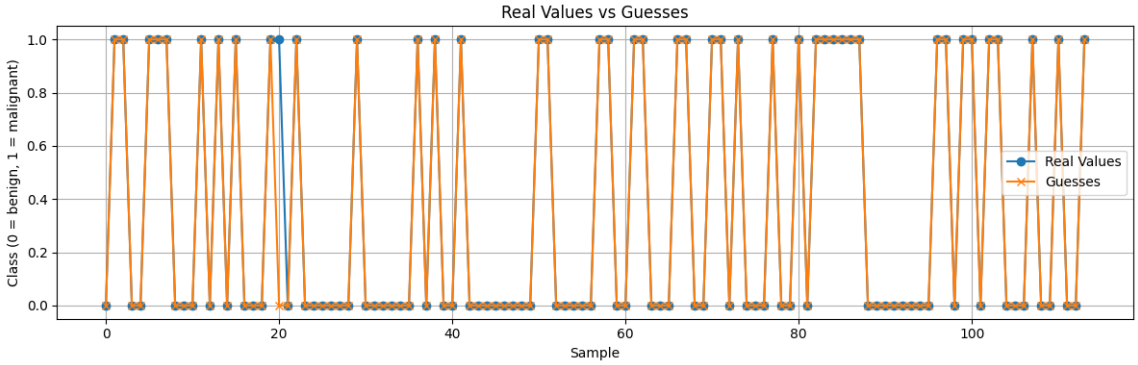

Figure 4: Confusion matrix after preprocessing and logistic regression modeling

In a nutshell, the pipeline significantly improved the model quality and classification. The project highlighted the role of data preprocessing, especially in medical applications where accuracy is vital.

Future work may involve testing alternative classifiers (e.g., Random Forest, SVM) or validating the approach on external datasets to assess robustness further.

References

[1] World Health Organization, "Cancer", 2023. Available: https://www.who.int/health-topics/cancer#tab=tab_1. [Accessed: 02-Mar-2025]

[2] Our World in Data, "Breast Cancer Figure", Available: https://ourworldindata.org/grapher/breast-cancer-death-rate-in-women. [Accessed: 02-Mar-2025]

[3] American Cancer Society, "How is a Biopsy Done?", Available: https://www.cancer.org/cancer/diagnosis-staging/tests/biopsy-and-cytology-tests/biopsy-types.html#:~:text=A%20biopsy%20is%20a%20procedure,be%20tested%20in%20a%20lab.&text=Biopsy%20samples%20of%20cells%20or,used%20to%20help%20diagnose%20cancer. [Accessed:02-Mar-2025]

[4] DataCamp. (2023). "Techniques to Handle Missing Data Values", Available: https://www.datacamp.com/tutorial/techniques-to-handle-missing-data-values

[5] Enders, C. K. (2010). "Applied Missing Data Analysis". New York: Guilford Press.

[6] Géron, A. (2019). "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow". O'Reilly Media.

[7] BYJU'S. (n.d.). *Interquartile Range (IQR)*. Available: https://byjus.com/maths/interquartile-range/

[8] Samira Alipour, "A Comprehensive Guide to Outliers in Machine Learning: Detection, Handling, and Impact," *Medium*, Available: https://medium.com/@samiraalipour/a-comprehensive-guide-to-outliers-in-machine-learning-detection-handling-and-impact-f7d965bba7 [Accessed: 09-May-2025].

[9] V. Agarwal, "Outlier detection with Boxplots," *Medium*, 2019. Available: https://medium.com/@agarwal.vishal819/outlier-detection-with-boxplots-1b6757fafa21 [Accessed: 09-May-2025].

[10] J. W. Tukey, *Exploratory Data Analysis*, Reading, MA: Addison-Wesley, 1977.

[11] D. Divya and S. S. Babu, "Methods to Detect Different Types of Outliers," in *Proceedings of the 2016 International Conference on ICT in Business, Industry, and Government (ICTBIG)*, IEEE, 2016, pp. 1–5. Available: https://www.researchgate.net/publication/311610830_Methods_to_detect_different_types_of_outliers

[12] Kim J. H. (2019). Multicollinearity and misleading statistical results. Korean journal of anesthesiology, 72(6), 558–569. https://doi.org/10.4097/kja.19087

[13] N. Shrestha, "Detecting Multicollinearity in Regression Analysis," *American Journal of Applied Mathematics and Statistics*, vol. 8, no. 2, pp. 39–42, 2020. Available: https://www.researchgate.net/publication/342413955_Detecting_Multicollinearity_in_Regression_Analysis

[14] DataCamp. (2024). What Is One Hot Encoding and How to Implement It in Python. Retrieved May 16, 2025, from https://www.datacamp.com/tutorial/one-hot-encoding-python-tutorial

[15] S. Kumar and R. S. Rajput, "A Comparative Study of Categorical Variable Encoding Techniques," *International Journal of Computer Applications*, vol. 175, no. 4, pp. 1–5, 2020. Available: https://ijcaonline.org/archives/volume175/number4/28474-2017915495

[16] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

[17] Scikit-learn developers. *Feature selection — scikit-learn 1.4.2 documentation*. Retrieved June 5, 2025, from https://scikit-learn.org/stable/modules/feature_selection.html

[18] Ribasanna, M. (2023). *Imputation Techniques in Machine Learning – Part 2: Handling Missing Data.* Retrieved from https://medium.com/@manukaribasanna/imputation-techniques-in-machine-learning-part-2-handling-missing-data-b62af123b8b1

[19] Yağcı, H. (2021). Feature Scaling with Scikit-Learn for Data Science. Retrieved from https://hersanyagci.medium.com/feature-scaling-with-scikit-learn-for-data-science-8c4cbcf2daff

[20] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1), 267–288.