Project Title

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXX

By

XXXXXXXX

Supervisor: XXXXXXXX

Dissertation Submitted to the University of Derby in Partial Fulfilment of the Requirements for the Award of Master of Science in Big Data Analytics

2023-2024

Abstract

Breast cancer remains one of the primary causes of mortality among women worldwide, emphasizing the critical need for innovative diagnostic and prognostic methodologies to improve treatment outcomes and patient survival rates. Existing studies mainly focused on developing machine learning (ML) models for cancer diagnosis with only little focus on the models' interpretability via Explainable AI (XAI) and identification of significant features that could aid accurate and robust diagnosis of cancer in clinical settings. To address this limitation, this project systematically investigated six state-of-the-art ML algorithms focusing on techniques that could enhance the models' predictive capability as well as their interpretability in the context of breast cancer diagnosis. Besides, the study used the Recursive Feature Elimination feature selection technique to construct a minimal feature vector that enhanced the overall performance of the models while an XAI tool (LIME: Local Interpretable Model-agnostic Explanations) was leveraged to demystify ML models decisions, enhancing transparency and fostering trust among medical practitioners and patients. The Wisconsin Breast Cancer Dataset (WBCD), a benchmark dataset was used and evaluation of the performances of the investigated models was done across four distinct metrics. Notable findings indicate that while all the models performed well, Logistic Regression yielded the highest precision and interpretability, making it the optimal classifier. Feature importance analysis highlights key predictors such as 'concave points_worst' and 'radius_worst', which are instrumental in distinguishing between malignant and benign cases. The outcomes further emphasized the potential of ML in revolutionizing breast cancer diagnostics, suggesting a shift towards more predictive, interpretable and personalized ML-based solution. This comprehensive approach not only improves the predictive accuracy of ML models in the context of breast cancer but also significantly contributes to the field of medical informatics by bridging the gap between advanced computational techniques and everyday clinical applications. The study advocates for ongoing enhancement of model interpretability and the ethical use of AI in healthcare to ensure these powerful tools are accessible and beneficial in real-world medical settings.

Keywords*: Breast Cancer Diagnosis, Machine Learning Algorithms, Predictive Modeling, Feature Selection Techniques, Explainable Artificial Intelligence (XAI), Model Interpretability.*

Acknowledgement

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

.

Table of Contents

List of Tables

List of Figures

1. Chapter One: Introduction

Cancer, a pervasive and intricate disease, continues to pose a significant global health challenge (Anisha et al. 2021). The ability to precisely predict and evaluate cancer is crucial for directing treatment choices and enhancing patient outcomes. In recent years, the integration of machine learning (ML) techniques has shown great promise in advancing the field of cancer research, particularly in enhancing cancer prediction and prognosis (Sharma & Rani, 2021).

Cancer's impact extends beyond the individual to affect families, communities, and healthcare systems on a global scale. The World Health Organization (WHO, 2020) highlights the urgency of addressing the rising incidence of cancer and implementing effective strategies for prevention, early detection, and treatment. Among the myriad forms of cancer, breast cancer stands out as one of the most prevalent and extensively studied (Wilkinson and Gathani, 2022). Breast cancer affects both men and women, although it is predominantly associated with the female population. According to global cancer statistics, breast cancer is the most common cancer among women, with an estimated 2.3 million new cases diagnosed annually (WHO, 2020). Its significance lies not only in its frequency but also in its diverse molecular subtypes, necessitating tailored approaches to diagnosis and prognosis.

Breast cancer, a complex and prevalent health concern, necessitates cutting-edge approaches for accurate prediction, prognosis, and subsequent policy decisions (Kourou et al., 2021). Within the breast, there are two distinct types of irregular cells: benign and malignant. Malignant cells, identified as cancer cells, pose a more serious threat, especially when they undergo replication or metastasize to other parts of the body (Saba, T., 2020). In contrast, benign cells exhibit a well-established structure. Diagnosing malignant tumors in the early stages of development is particularly challenging due to the constrained space within the breast and the prevalence of fatty and dense tissue. Even advanced automated or computerized systems face difficulties in accurately identifying and characterizing breast tumors.

1.1 Project Rationale

Breast cancer continues to be a major cause of death among women worldwide, emphasizing the critical need for advancements in diagnostic and prognostic methods to improve treatment outcomes. Although numerous studies have focused on developing machine learning (ML) models

for cancer diagnosis, there has been a notable lack of emphasis on the interpretability of these models using Explainable AI (XAI). Additionally, identifying key features that can support accurate and robust cancer diagnosis in clinical settings has not been sufficiently addressed.

This project aims to fill this gap by systematically investigating six advanced ML algorithms, focusing on techniques that enhance both the predictive capability and interpretability of these models in the context of breast cancer diagnosis. By incorporating XAI, the project seeks to make the decision-making processes of ML models more transparent and understandable to clinicians.

Furthermore, this project aspires to contribute significantly to cancer research by improving the accuracy, reliability, and utility of ML models for breast cancer diagnosis. The goal is to enhance patient care and outcomes by providing more accurate and interpretable diagnostic tools that can be effectively used in clinical settings.

1.2 Objectives of the Study

The main objectives of the study are to:

- **Optimal Feature Set Identification**: Construct an optimal feature vector for building a robust and efficient ML classifier for breast cancer prediction and prognosis.
- **Optimal Classifier Identification**: Examine how the constructed feature vector impact benchmark ML algorithms including Logistic Regression, Support Vector Machine, Random Forest, K-Nearest Neighbor, Decision Tree and Ensemble Method for breast cancer diagnosis.
- **Variable Significance Determination**: Identify the most significant variables in breast cancer prediction, providing insights for targeted intervention and policy formulation.
- **XAI (LIME) Analysis:** Evaluate the utility of XAI tool (LIME) in explaining the predictions made by ML models, enhancing their interpretability for clinical use.

1.3 Scope of Exploration

This study explores various machine learning techniques and explanatory methods aimed at advancing the diagnosis, classification, and prognosis of breast cancer. The techniques include Logistic Regression, which models binary outcomes based on predictor variables for classification tasks (Wibowo et al., 2021), and Support Vector Machine (SVM), which finds optimal hyperplanes to separate data into risk groups, handling complex dataset relationships effectively (Boeri et al.,

2020). K-Nearest Neighbors (KNN) classifies data by the majority class of its nearest neighbors, offering straightforward categorization (Shrivastava & Buri, 2023). Random Forest uses multiple decision trees to enhance predictive accuracy and patient risk profiling, while Decision Trees provide a clear, flowchart-like structure for making predictions (Anisha et al., 2021).

Ensemble methods combine different machine learning strategies to improve the accuracy of predictions by leveraging strengths from various algorithms (Yaqoob, Aziz, & Verma, 2023). Additionally, LIME (Local Interpretable Model-agnostic Explanations) helps clarify complex model predictions by training simpler models on locally perturbed data, making the predictions more accessible and interpretable, particularly useful in clinical settings (Garreau & Luxburg, 2020).

These integrated approaches underscore a commitment to enhancing breast cancer research through diverse and effective machine learning techniques.

1.4 Significance of the Study

This study stands to significantly impact breast cancer research and policy decision-making by:

- **Optimizing Predictive Accuracy**: Enhancing the precision of predictions and prognoses through the identification of optimal classifiers.
- **Enhancing Model Interpretability**: Improving understanding of ML model predictions with LIME, fostering trust and facilitating clinical integration.
- **Identifying Influential Variables**: Unveiling key variables that influence breast cancer outcomes, offering insights for personalized care and policy development.

1.5 Contribution to Knowledge

This thesis evaluates the effectiveness of machine learning models for breast cancer diagnosis using the Wisconsin Breast Cancer Dataset (WBCD). Key advancements include employing Recursive Feature Elimination (RFE) with logistic regression for improved interpretability and computational efficiency. Rigorous data preprocessing, coupled with cross-validation and Grid Search techniques, ensures model robustness and generalizability. Enhanced transparency in clinical decision-making is achieved through Local Interpretable Model-agnostic Explanations (LIME). The study utilizes a comprehensive evaluation framework, demonstrating the potential of

machine learning to enhance diagnostic accuracy and efficiency in breast cancer, with significant implications for patient outcomes.

1.6 Organization of Thesis

1.6.1 Chapter 1: Introduction

This segment of the research offers a comprehensive overview of the general context and justification for initiating the study. It also clearly specifies the scope and objectives of the investigation.

1.6.2 Chapter 2: Literature Review

Reviews existing literature on breast cancer diagnostics using machine learning approaches and identifying gaps. This review establishes the foundation for exploring innovative diagnostic improvements.

1.6.3 Chapter 3: Methodology

Describes the research methodology, including the machine learning algorithms selected and data collection processes. Details the design and implementation of these algorithms, explaining the structure of the datasets used and the rationale behind the choices made.

1.6.4 Chapter 4: Results ,Analysis and Discussion

Presents and discusses the results of algorithm performance, evaluating their effectiveness and interpretability in breast cancer diagnosis.

1.6.5 Chapter 5: Conclusion and Future Work

Summarizes the study's findings, discusses its implications for clinical practice, and suggests directions for future research in advanced machine learning applications.

Moving forward, this research aims to enhance machine learning applications in breast cancer diagnostics, focusing on improving accuracy and interpretability. It aims to refine treatment strategies and influence healthcare policies by optimizing diagnostic models and identifying key variables. Conclusively, the next chapter reviews existing literature to identify gaps that this thesis addresses, setting the stage for an in-depth examination of the methodologies and technologies involved in breast cancer diagnostics.

2. Chapter Two: Literature Review

2.1 Background and Importance of Breast Cancer Research

Breast cancer is a principal cause of cancer-related mortality among women worldwide, significantly impacting numerous individuals annually (Houghton and Hankinson, 2021). The significance of timely and accurate diagnosis and prognosis in managing the disease effectively and improving patient survival rates cannot be overstated (DeSantis et al., 2019). The World Health Organization (WHO) highlights breast cancer as a critical global health issue, with its increasing prevalence across different regions pointing to the pressing demand for advanced prevention techniques, early detection methodologies, and more effective treatments (World Health Organization, 2022). Beyond the physical implications, breast cancer places substantial emotional and financial strains on the affected individuals, their loved ones, and the broader healthcare infrastructure (Wilkinson and Gathani, 2022). This scenario highlights the imperative for continuous exploration and development in the fields of diagnostic and prognostic technologies, driving the necessity of the project focused on enhancing machine learning approaches for predicting and prognosing breast cancer.

2.2 Traditional Approaches to Breast Cancer Prediction and Prognosis

2.2.1 Clinical and Pathological Methods

Traditional methods for diagnosing and prognosticating breast cancer primarily involve clinical examinations, mammography, biopsy, and pathological staging (Petinrin et al., 2023). Mammography has been the cornerstone of breast cancer screening, helping in early detection and reducing mortality rates. Biopsies, followed by histopathological analysis, provide definitive diagnoses and insights into the cancer's aggressiveness and potential response to treatments (Roslidar et al., 2020). Pathological staging, based on the tumor size, lymph node involvement, and metastasis (TNM classification), further assists in prognosis and treatment planning.

2.2.2 Limitations and Challenges of Traditional Approaches

Despite advancements, these traditional approaches have limitations. Mammography, for instance, has variable sensitivity and specificity, particularly in women with dense breast tissue, leading to false positives and negatives. Biopsies, though definitive, are invasive procedures with associated risks and discomfort(Roslidar et al., 2020). Moreover, the pathological staging provides a broad

prognosis but may not capture the tumor's molecular complexity or predict individual responses to treatment accurately (Petinrin et al., 2023).These limitations underscore the necessity for non-invasive, precise, and personalized diagnostic and prognostic tools.

2.3 Scope of the Review

The review delved into various ML algorithms, such as Logistic Regression, Support Vector Machine, K-Nearest Neighbors, Random Forest, and Decision Tree, alongside ensemble methods and their applications in breast cancer research. Each algorithm's theoretical framework, advantages, limitations, and contributions to enhancing predictive accuracy and prognosis will be scrutinized. This analysis will include an evaluation of current studies and results that underscore the algorithms' effectiveness and areas requiring further investigation.

The review also focused on the interpretability of ML models, examining methodologies like Local Interpretable Model-agnostic Explanations (LIME). The aim is to underscore the importance of making complex ML predictions transparent and comprehensible for healthcare professionals, thus facilitating their integration into clinical practice. This section assessed how these interpretative techniques have been applied in existing research, evaluating their success in bridging the gap between data science and patient care. Therefore, this review aims to evaluate the efficacy of prevailing ML strategies in the realm of breast cancer care, pinpoint existing challenges and constraints, and propose directions for forthcoming investigations.

2.4 AI and Machine Learning in Cancer Research

2.4.1 Significance of Machine Learning in Cancer Prediction and Prognosis

Machine Learning (ML), an integral branch of artificial intelligence, employs sophisticated algorithms to analyze data, extract insights, and predict outcomes in various domains (Entezari et al., 2023). Machine learning (ML) focuses on creating algorithms and statistical models that empower computer systems to learn from experience and enhance their performance on specific tasks without direct programming. (Sharma & Rani, 2021). Essentially, it involves developing algorithms that empower machines to analyze data, identify patterns, and make informed predictions or decisions.

Within the realm of oncology, ML techniques meticulously examine extensive datasets encompassing patient records, imaging scans, genetic information, and other critical biomarkers.

This analysis enables the identification of subtle patterns and relationships that may not be immediately apparent to human researchers. By harnessing these capabilities, ML algorithms contribute significantly to enhancing the precision of cancer detection, understanding its progression, and tailoring prognosis assessments, thereby offering a promising avenue for advances in cancer care and treatment strategies.

2.4.2 Early Detection and Diagnostic Accuracy

Machine Learning (ML) significantly enhances oncology by improving early cancer detection and diagnostic accuracy (Gaur and Jagtap, 2022). ML algorithms analyze complex data from mammograms, MRIs, and genomic records, detecting subtle signs of early-stage cancers for timely and effective treatment. Particularly in breast cancer, ML improves mammogram accuracy, reduces false positives, and assists in identifying elusive cases. Additionally, by analyzing genetic data, ML models predict cancer malignancy, advancing personalized medicine and ultimately leading to better patient outcomes.

2.4.3 Personalized Treatment Plans

Machine learning (ML) enhances survival analysis by utilizing clinical data's 'worst' features to more accurately predict outcomes than traditional methods. These sophisticated techniques develop predictive models that handle the complexity of breast cancer, tailoring treatment strategies to individual patient profiles for optimized effectiveness (Gaur and Jagtap, 2022). Moreover, the use of interpretability tools like LIME and Shapley explanations increases transparency, aiding decision-making in personalized treatment planning.

2.4.4 Prognostic Value

ML models possess the ability to delve into historical treatment data and current research findings to forecast the progression of a patient's cancer. This capability provides critical information on survival probabilities, the risk of cancer returning, and the expected quality of life after treatment (Gaur and Jagtap, 2022). Such predictive insights are instrumental in assisting both patients and medical professionals in making well-informed choices concerning treatment pathways and, when necessary, end-of-life planning.

2.5 Types of Machine Learning

Machine Learning can be categorized into supervised , unsupervised and reinforcement learning.

2.5.1 Supervised Learning

Supervised machine learning uses labeled datasets to train algorithms, transforming oncology by enhancing cancer prediction and prognosis (Sarker, 2021). These models analyze patient data, including genetics, imaging, and treatment outcomes, to uncover patterns that improve the accuracy of predicting cancer progression and response to treatments (Kumar, Kaur, and Singh, 2020). This advancement enables personalized interventions, marking a significant shift towards precision medicine and potentially improving outcomes by customizing treatment to individual patient profiles.

2.5.2 Unsupervised Learning

Unsupervised machine learning discovers hidden patterns in data without predefined labels, playing a crucial role in oncology by offering novel insights into cancer biology and patient stratification (Kumar, Kaur, and Singh, 2020). Techniques like clustering and dimensionality reduction categorize patients into distinct groups based on genetic profiles, tumor characteristics, or treatment responses, uncovering cancer subtypes and potential biomarkers (Sarker, 2021). This approach processes extensive data from technologies like genomics and radiomics to identify complex patterns that inform prognosis and therapy responses, significantly advancing personalized medicine.

2.5.3 Reinforcement Learning (RL)

Reinforcement learning (RL) is a machine learning method where an agent learns to make decisions from the outcomes of its actions, receiving rewards or penalties (Sarker, 2021) (Kumar, Kaur, and Singh, 2020). In healthcare, particularly cancer treatment, RL is used to develop personalized treatment plans by analyzing a patient's medical history and responses to treatments, continuously optimizing recommendations to maximize outcomes and minimize side effects.

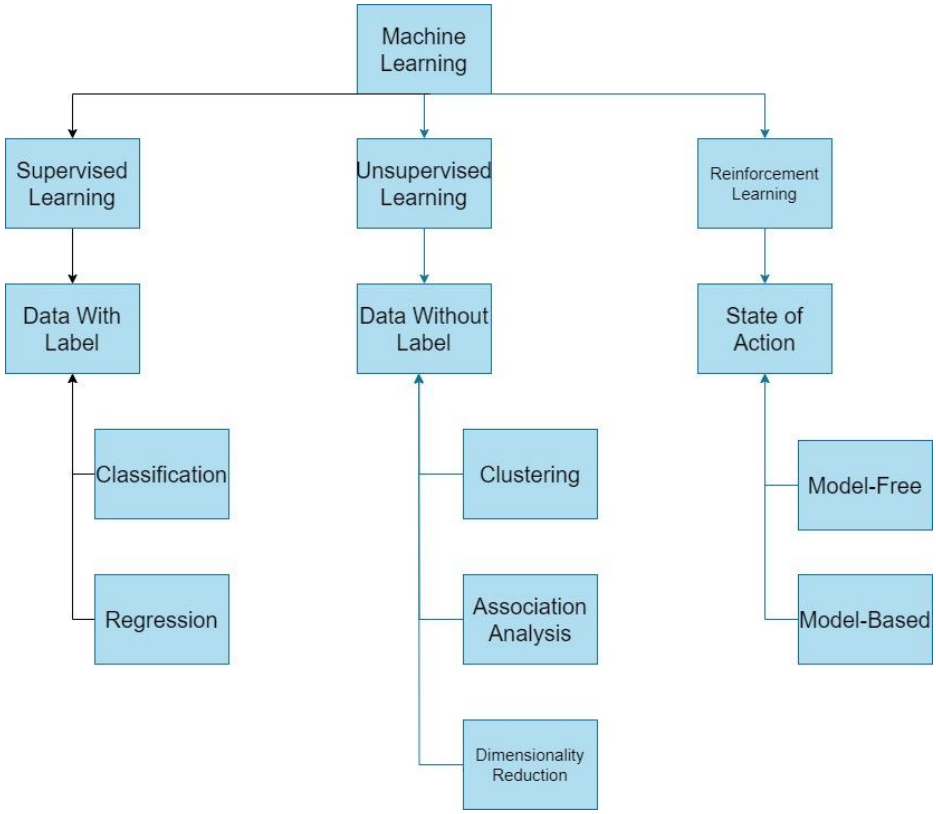


Figure 1: Categories of Machine Learning

2.6 Machine Learning Techniques for Breast Cancer: Enhanced Algorithms for Prediction and Prognosis

The incorporation of machine learning (ML) techniques into breast cancer research has significantly improved the capabilities for early detection, accurate diagnosis, and effective prognosis of this pervasive disease (Gaur and Jagtap, 2022). This section delves into the specifics of various ML techniques, including Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, Decision Tree, and Ensemble Methods, highlighting their application in breast cancer research.

2.6.1 Logistic Regression (LR)

Logistic Regression is a statistical method used for binary classification problems, making it highly suitable for distinguishing between two outcomes such as malignant or benign tumors in cancer detection (Kumar and Gota, 2023). This model calculates the probability of a particular outcome (e.g., malignancy) based on input features, which could include patient demographics, tumor

characteristics, genetic information, or any other relevant clinical markers. The essence of Logistic Regression in oncology lies in its ability to provide a straightforward, interpretable model that gives the odds of a patient having cancer based on the predictors included in the model. It is especially valued for its simplicity and the ease with which it can be explained to non-specialists, making it a staple in medical research for initial screening and diagnosis processes.

Studies on Cancer showcasing Logistic Regression

Logistic Regression is proving invaluable in oncology, highlighted by impactful studies on prostate and breast cancer diagnostics. Arash Hooshmand's research utilized logistic regression to diagnose prostate cancer with high precision, using RNA genomic data from 595 samples (Hooshmand, 2021). This demonstrated the method's accuracy and broader applicability for various cancer types. (Li et al. 2019) employed logistic regression to differentiate between benign and malignant breast cancer cases effectively, using clinical data like age and tumor size. In their 2023 study, Viswanatha et al. applied logistic regression to classify breast tumors using the Wisconsin Diagnostic Breast Cancer dataset, achieving high accuracy in distinguishing between benign and malignant tumors. This method proved effective in clinical settings, offering clear, interpretable results that aid in early cancer detection and informed decision-making (Viswanatha et al., 2023). These studies collectively emphasize logistic regression's potential to enhance cancer diagnostics, improving clinical decision-making and patient outcomes.

2.6.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful machine learning method that effectively classifies complex, high-dimensional datasets by identifying the optimal hyperplane to separate different classes (Akinnuwesi et al., 2023). SVM's versatility in using both linear and non-linear kernels, like the Radial Basis Function (RBF), allows it to manage both linearly separable and complex non-linear datasets, capturing intricate genetic patterns. This capability makes SVM highly valuable in cancer research, particularly for precise classification of cancerous versus non-cancerous cells and identifying cancer subtypes.

Studies and Findings on Cancer showcasing SVM.

The Support Vector Machine (SVM) algorithm has demonstrated significant efficacy in breast cancer diagnostics across multiple studies. (Liu and Li 2023) applied SVM to predict breast cancer, achieving an impressive 94.4% accuracy, showcasing its ability to distinguish between benign and malignant tumors efficiently. Similarly, (Deepika et al. 2021) utilized SVM to predict breast cancer stages, achieving a remarkable 99% accuracy, highlighting its robustness in handling complex classifications and large datasets. These findings collectively underscore SVM's potential to revolutionize early cancer detection and diagnosis, enhancing clinical decision-making and patient outcomes.

2.6.3 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) employs instance-based learning, categorizing new instances based on the consensus of the closest 'k' neighbors within the feature space (Kumar, Daniya, and Ajayan, 2020). This method, which uses measures like Euclidean distance, effectively diagnoses and classifies cancers, determining if a tumor is malignant or benign based on similar existing cases. KNN's flexibility to adapt and update with new data without retraining makes it well-suited for dynamic cancer data.

Studies and Findings on Cancer showcasing KNN.

In their study, (Maliha et al.,2019) utilized the application of the K-nearest neighbor (KNN) algorithm for cancer disease prediction. KNN employs the dataset by dividing it into different classes to facilitate the prediction of the classification for new data points. This method highlights the utility of KNN in analyzing datasets for cancer prediction, demonstrating its effectiveness in classifying cancer types based on the similarity measures of data points within the dataset.

2.6.4 Random Forest (RF)

Random Forest is an ensemble learning method that uses multiple decision trees to boost predictive accuracy and stability, reducing overfitting risks (Anisha et al., 2021). Each tree is built from a random subset of data and features, contributing to the model's effectiveness. Notably, Random Forest has been successful in distinguishing between malignant and benign tumors using gene expression data, analyzing complex interactions among thousands of genes to detect subtle patterns indicative of specific cancer types or subtypes.

Studies and Findings on Cancer showcasing Random Forest

The studies by (Anisha et al.,2021) and (Octaviani and Rustam, 2019) demonstrate the effectiveness of the Random Forest Classifier in breast cancer diagnostics. Anisha et al. achieved a 98% accuracy rate by using multiple decision trees to assess breast cancer risk, illustrating the method's utility in early detection. Similarly, Octaviani and Rustam achieved a perfect 100% classification accuracy, using Random Forest to analyze large cancer microarray datasets. Together, these studies underscore the significant potential of Random Forest in enhancing breast cancer prediction and transforming medical diagnostics.

2.6.5 Decision Tree (DT)

Decision Trees are supervised learning algorithms predominantly used for classification. They organize information in a hierarchical model, consisting of a root node, intermediate branches, and terminal leaves, each representing features, decisions, and outcomes respectively (Sheth, Tripathi, and Sharma, 2022). By strategically dividing data at each node to effectively separate classes, these algorithms offer a clear visual framework for exploring various decision outcomes based on current conditions.

Studies and Findings on Cancer showcasing Decision Tree

Decision Trees have demonstrated significant utility in oncology by identifying critical diagnostic and prognostic factors in cancer datasets. Several studies have showcased their effectiveness in this domain. The study "Breast Cancer Classification using Decision Tree Algorithms" by Omar Tarawneh, Mohammed Otair, and Moath Husni offers valuable insights into the application of decision tree algorithms for the diagnosis and treatment of breast cancer. This research highlights the use of decision trees, a form of machine learning model known for their simplicity and ease of interpretation, in representing actual breast cancer diagnoses to inform both local and systemic treatment strategies (Tarawneh et al., 2022).

2.6.6 Ensemble Methods

These techniques involve integrating multiple learning algorithms to achieve superior predictive performance compared to what could be achieved by any single model alone.

By leveraging the strengths of various models and mitigating their weaknesses, ensemble methods can significantly enhance the accuracy and reliability of cancer predictions (Jabbar, 2021).

Types of Ensemble Methods:

Bagging (Bootstrap Aggregating): Involves training identical models on different data subsets, then combining their outputs. Example: Random Forest, which integrates multiple decision trees.
Boosting: Sequentially trains models to correct predecessors' errors, improving overall accuracy. Examples include AdaBoost (Adaptive Boosting) and Gradient Boosting.
Stacking: Combines diverse models' predictions through a meta-model for final output.

2.7 Evaluating the Efficacy of Machine Learning Techniques in Breast Cancer Diagnosis: A Comparative Study

The advancement of machine learning (ML) algorithms in breast cancer diagnosis and prognosis is highlighted through comparative studies that assess their effectiveness across various metrics such as accuracy, sensitivity, and specificity:

The 2022 study by Vraj Sheth, Urvashi Tripathi, and Ankit stands out for its in-depth analysis of multiple classifiers including Decision Tree, SVM, Naive Bayesian, and K-Nearest Neighbor (KNN) across different datasets. This study found the Naive Bayesian algorithm to be the most effective, achieving superior performance with an accuracy of 94%, precision of 93%, recall of 92%, and an F1 score of 93%, demonstrating the critical influence of dataset characteristics on algorithm performance (Sheth, Tripathi, & Ankit, 2022).

Similarly, (Binsaif, 2022) evaluated six ML algorithms—Decision Tree, Random Forest, K-Nearest Neighbors, Artificial Neural Networks, Support Vector Machines, and Logistic Regression—using the Coimbra Breast Cancer Database. The study highlighted Random Forest as the best classifier, with a notable accuracy of 95% and an area under the curve (AUC) of 0.96, emphasizing its utility in improving early diagnosis and patient outcomes (Binsaif, 2022).

Rane and Kanade (2020) also contributed significantly by examining the efficacy of six machine learning algorithms including Naive Bayes, Random Forest, Artificial Neural Networks, K-Nearest Neighbor, Support Vector Machine, and Decision Tree. Their approach was crucial for

early detection, aiming to leverage computational models to increase the effectiveness of treatments and patient outcomes (Rane & Kanade, 2020).

Additionally, the research by Battineni, Chintalapudi, and Amenta (2020) focused on the effectiveness of SVM and Logistic Regression algorithms, documenting that SVM achieved an accuracy of 97.66% using all features, and enhancing SVM's accuracy to 98.25% with selective feature improvement. Logistic Regression showed a 100% true positive prediction with limited features, underscoring the importance of feature selection in diagnostic accuracy (Battineni, Chintalapudi, & Amenta, 2020).

Further, the study by Jaiswal, Suman, & Bisen (2023) introduced an enhanced XGBoost algorithm, which demonstrated superior accuracy at 98.24% compared to other classification techniques, providing a significant advancement in early breast cancer detection methods (Jaiswal, Suman, & Bisen, 2023).

These studies collectively underscore the transformative impact of ML in breast cancer research, offering insights into each algorithm's strengths and weaknesses and setting the stage for future advancements in personalized cancer care. As ML technology continues to evolve, its integration into breast cancer research promises to yield even more sophisticated tools for combating this disease, moving towards a future where personalized cancer care becomes the norm.

Table 1: Summary of Related Literature on ML Techniques on Cancer Prediction and Prognosis

Author	**Algorithm(s) Used**	**Dataset Used**	**Contribution**	**Limitation**
Binsaif, N., 2022	Decision Tree Random Forest K-Nearest Neighbors (KNN) Artificial Neural Networks (ANN) Support Vector Machines (SVM)	The Breast Cancer Database of Coimbra, made available by the UCI	The study's major contribution is the evaluation of six machine learning algorithms to classify breast cancer data. It highlights the random forest model's superiority	Challenge in significantly improving model performance through the reduction of input variables, indicating the

	Logistic Regression		when using nine variables, achieving 83.3% accuracy, 100% sensitivity, 64% specificity, and an AUC of 0.881	complexity of identifying a minimal set of features that can still yield high predictive performance.
(Rane et al., 2020)	Naive Bayes (NB), Random Forest (RF), Artificial Neural Networks (ANN), Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree (DT)	Wisconsin Diagnostic Breast Cancer (WDBC) dataset	This study provides a comprehensive comparison of six machine learning algorithms to classify and predict breast cancer as benign or malignant based on the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. The comparison aims to identify the most effective algorithm	There's a lack of discussion on the computational efficiency of these algorithms, which is essential where response time and resource utilization are critical.
(Sheth, Tripathi, and Sharma, 2022)	Decision Tree, Support Vector Machine (SVM), Naive Bayesian, K-nearest neighbor (K-NN)	The paper utilizes five distinct datasets to assess the performance of the algorithms, including Placement	The study undertakes a detailed comparative analysis of four significant classification algorithms, offering insights into their accuracy, precision, recall, and F1 score across different types of data.	The analysis does not consider the complexity of the generated models or their interpretability. Some algorithms, like decision trees or Naive Bayes, inherently provide

		Dataset, Heart Disease Dataset, Wine Quality Dataset, Glass Quality for the Classification.		more interpretable models compared to SVM or k-NN. An analysis of the trade-offs between model complexity, interpretability, and performance would have added depth to the study.
(Battineni, Chintalapudi, and Amenta, 2020)	Support Vector Machines (SVM), Logistic Regression (LR), and K-Nearest Neighbors (KNN)	The Wisconsin Breast Cancer Dataset (WBCD)	The paper presents ML models that leverage a limited set of features to achieve high classification accuracy in distinguishing between malignant and benign breast tumors.	Model Complexity and Explainability: The paper does not address the complexities, or the interpretability of the ML models developed. In medical applications, the ability to interpret and trust the decisions made by ML models is as crucial as their predictive accuracy.
(Mashudi et al., 2021)	k-Nearest Neighbors (k-NN), Random Forest,	Wisconsin Diagnostic Breast Cancer	The study investigates the performance of three machine learning	Consideration of Model Interpretability:

	Support Vector Machine (SVM), and AdaBoost ensemble methods	(WDBC) dataset with 23 selected attributes	algorithms and ensemble techniques for breast cancer classification into malignant and benign tumors, utilizing a comprehensive approach that includes 2-fold, 3-fold, 5-fold, and 10-fold cross-validation to optimize accuracy.	In medical applications, the interpretability of machine learning models is nearly as critical as their predictive accuracy. The study does not delve into the interpretability of the models explored.

2.8 Improving Interpretability of ML Models in Oncology: Focus on Breast Cancer

Anisha et al. (2021) identifies key challenges in selecting effective ML classifiers and enhancing model transparency, which are crucial for clinical adoption. To address the opacity of ML models and build trust among healthcare providers, interpretability tools like Local Interpretable Model-agnostic Explanations (LIME) and Shapley explanations have been developed (Rodríguez-Pérez & Bajorath, 2020; Garreau & Luxburg, 2020).

Separately, (Yaqoob, Aziz, and Verma, 2023) examine the evolution of ML applications in cancer classification, highlighting improvements in diagnostic accuracy and the push towards tailored treatment strategies. They point out the persisting issues with data quality and model interpretability, advocating for enhanced algorithms and better data management to optimize ML's application in cancer care. This approach is essential for integrating these advanced tools into clinical practice effectively, aiming to improve patient outcomes in oncology.

2.8.1 Need for Model Interpretability

In breast cancer care, the interpretability of machine learning (ML) models is crucial, serving as a foundation for ethical and patient-centered healthcare. It allows clinicians to trust and understand the models' predictions, supporting informed decision-making and clear communication with

patients. This is especially important in oncology, where treatment decisions significantly affect patient outcomes and quality of life. Interpretability aids in customizing treatment plans to align with patient preferences and the specific stage and subtype of cancer.

2.8.2 Techniques for Enhancing Interpretability- Local Interpretable Model-agnostic Explanations (LIME)

Local Interpretable Model-agnostic Explanations (LIME) is a revolutionary approach in the domain of machine learning interpretability, particularly significant in high-stakes fields like healthcare, where understanding the reasoning behind predictions is as crucial as the predictions themselves (Chudasama et al., 2023). In the context of enhancing machine learning for cancer prediction and prognosis, LIME plays a pivotal role in bridging the gap between complex model outputs and clinical decision-making.

2.8.3 The Advantages of LIME: Bridging AI and Clinical Practice with Transparency

Interpretable Insights: For cancer prediction and prognosis, where decisions directly impact patient treatment and outcomes, LIME can provide interpretable insights into the model's predictions. It allows healthcare professionals to understand why a model predicts a certain outcome, such as identifying cancer or predicting its progression, based on specific patient data features.

Trust and Transparency: By offering explanations for individual predictions, LIME enhances trust in machine learning models among clinicians and patients. It ensures transparency in how algorithmic decisions are made, crucial in medical settings where understanding the rationale behind a prognosis or diagnosis can influence treatment choices and patient consent.

Model Improvement: LIME can also aid in model development and refinement by highlighting areas where the model might be relying on irrelevant or biased features for its predictions. This feedback loop is invaluable for researchers and practitioners aiming to enhance the accuracy and fairness of machine learning models in cancer care.

Facilitating Clinical Integration: The interpretable explanations generated by LIME make it easier for healthcare professionals to integrate machine learning predictions into their clinical

practice, ensuring that these advanced tools complement rather than complicate medical decision-making processes.

2.9 Case Studies of LIME Application in Breast Cancer

Several case studies have demonstrated the effectiveness of LIME in improving the interpretability of breast cancer prediction models. In their study on enhancing the explainability of machine learning (ML) models for breast cancer survival prediction, (Jansen et al.,2020) employed LIME to demystify the predictions of an ML model developed using data from the Netherlands Cancer Registry. The study underscores the necessity of explainability in clinical ML applications, advocating for further research to integrate these techniques into oncological practice effectively. (Ladbury et al.,2022) provide a compelling narrative review on the use of Explainable Artificial Intelligence (XAI), particularly through LIME to enhance the interpretability of machine learning models in oncology. This review illuminates how these model-agnostic frameworks have been applied across various domains such as prognosis, diagnosis, and treatment selection, demonstrating their pivotal role in translating complex ML insights into actionable, clinician-friendly information.

By offering intuitive visualizations and interpretable rules, LIME significantly improve clinicians' trust and facilitate the integration of advanced ML models into clinical practice, marking a critical step towards leveraging AI to improve patient outcomes in oncology.

2.10 Variable Importance in Breast Cancer Prediction Models

The advancement of machine learning (ML) techniques in oncology, particularly in breast cancer prediction and prognosis, has underscored the critical role of identifying and selecting key predictive variables. These variables, which range from genetic markers to tumor characteristics, significantly influence the performance of predictive models. By carefully selecting these variables, researchers can develop more accurate and reliable ML models that contribute to personalized treatment and improved patient outcomes.

2.11 Feature Selection

Feature selection is a vital component of the machine learning workflow, crucial for enhancing model accuracy, computational efficiency, and interpretability, especially in complex areas like cancer prediction and prognosis. It involves distilling the dataset to retain only the most impactful variables, thus improving model performance and reducing overfitting risks. Key feature selection techniques relevant to cancer include:

2.11.1 Filter Methods

Feature selection techniques like filter methods use statistical measures to assess the significance of each feature relative to the target variable in cancer datasets. These methods quickly pinpoint biomarkers or clinical variables linked to cancer presence, subtype classification, or outcome prediction. Examples include correlation coefficient analysis, chi-squared tests, and mutual information (Pudjihartono et al., 2022).

2.11.2 Wrapper Methods

Wrapper methods for feature selection work by iteratively evaluating different subsets of features based on the performance of a specific machine learning algorithm (Pudjihartono et al., 2022). An example of this method is Recursive Feature Elimination (RFE). Wrapper methods consider the interactions between features within the context of the chosen machine learning algorithm, which can lead to better performance.

2.11.3 Embedded Methods

Embedded methods integrate feature selection directly into the model training process, making it a core part of the learning algorithm. These methods excel in identifying complex patterns within cancer prediction datasets (Pudjihartono et al., 2022). Examples include LASSO and Ridge regression, which apply regularization to minimize less critical features, and decision trees or ensemble methods like Random Forests, which determine feature importance from the trees' structure.

2.11.4 Hybrid Methods

Hybrid methods combine elements of filter, wrapper, and embedded approaches to leverage their respective strengths. In the context of cancer prediction and prognosis, hybrid methods can be designed to first reduce the feature space using filter methods and then apply wrapper or embedded techniques to fine-tune the selection (Pudjihartono et al., 2022).

2.12 Findings from the Literature Review on Machine Learning in Breast Cancer Prediction and Prognosis

The comprehensive literature review highlights the considerable promise of machine learning (ML) technologies to revolutionize breast cancer diagnosis and prognosis. These advanced ML models have demonstrated a marked improvement in diagnostic accuracy, personalized treatment planning, and prognostic assessments, highlighting their critical role in the evolution of oncological care. Notably, the application of interpretative tools such as LIME has proven instrumental in enhancing the transparency and usability of these models, making them more accessible and trustworthy for clinical practitioners.

However, the review also illuminates several challenges that persist in the broader adoption of ML in oncology. These include concerns regarding data privacy, the potential biases inherent in training data, and the necessity for robust, interpretable models that healthcare providers can trust and understand. Furthermore, the importance of effective feature selection has been consistently emphasized, revealing its impact on the performance and efficiency of predictive models.

Given these insights, the forthcoming methodology aims to build on the foundational knowledge established through the review. The primary focus will be on refining feature selection techniques to enhance model accuracy and computational efficiency. Additionally, the methodology will explore the development of more transparent and interpretable ML models, employing tools like LIME to ensure that these models can be seamlessly integrated into clinical settings. This approach will not only address the highlighted challenges but also leverage the strengths of ML to advance breast cancer diagnosis and prognosis further. By focusing on these strategic areas, the research aims to contribute significantly to the field, providing robust, ethical, and practical ML solutions that improve patient outcomes and streamline clinical workflows.

2.13 Refined Research Questions

To navigate the complexities and harness the full potential of machine learning (ML) in breast cancer prediction and prognosis, the following areas are crucial for exploration:

a) **Model Comparison and Optimization**

Which machine learning models most accurately predict breast cancer and its prognosis?

b) **Feature Selection and Importance**

What features most significantly predict breast cancer occurrence and outcomes?

c) **Explainability**

What is the efficacy of LIME technique in clarifying ML model predictions, and how can their interpretability be leveraged for clinical and policy decision-making?

d) **Risk Factor Analysis**

Which modifiable risk factors are most predictive of breast cancer, and how can ML models incorporate them?

By addressing these refined research questions, this study in no doubt will significantly contribute to the advancement of breast cancer care, leveraging machine learning to enhance prediction, diagnosis, prognosis, and personalized treatment planning.

3. Chapter Three: Methodology

This section outlines the methodology framework used in this study to investigate Machine Learning (ML) techniques for predicting and prognosing breast cancer. It details the research approach of the study, methods for data collection and preprocessing, selection and evaluation of ML classifiers, and the application of interpretability methods such as LIME to better understand model decisions.

3.1 Research Approach

The research conducted in this study adopts a quantitative approach, focusing on the systematic evaluation and implementation of various machine learning (ML) techniques to enhance the prediction and prognosis of breast cancer using the Wisconsin Breast Cancer Dataset (WBCD). The methodology encompasses several key steps, aimed at ensuring both the accuracy and interpretability of the ML models used.

3.1.1 Experimental Framework

Building on the shortcomings noted in Chapter 2, this report outlines a comprehensive framework designed to enhance the application of machine learning in predicting and prognosing breast cancer. It moves beyond basic performance metrics to emphasize the transparency of model decisions, the significance of variables, and their practical application in clinical environments. By evaluating different machine learning classifiers, incorporating interpretability methods like LIME to clarify results, and identifying key variables, this framework aims to make machine learning predictions not only accurate but also clear and usable for healthcare providers. The objective is to integrate these insights into clinical practice, thus improving the standard of care and outcomes for breast cancer patients through better-informed decision-making.

This methodology is visually represented by a flowchart in Figure 2, which outlines the detailed steps involved in the process:

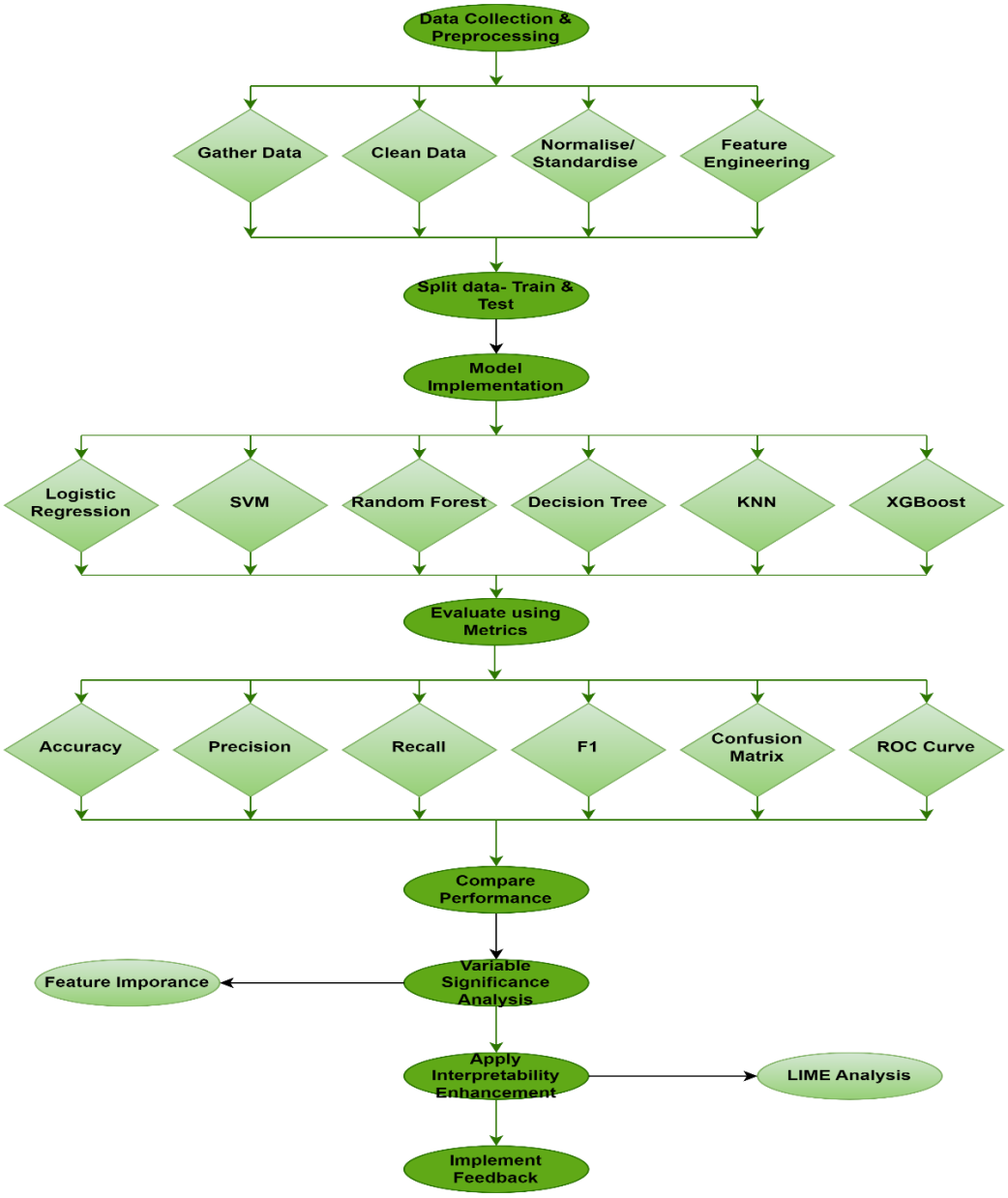


Figure 2: Comprehensive Workflow for ML Model Development in Cancer Research

This flowchart illustrates the comprehensive steps involved in the application of machine learning algorithms for predicting and diagnosing breast cancer, including data preprocessing, model training, performance evaluation, and interpretability enhancement.

3.2 Implementation Tools and Software

This project leveraged a comprehensive suite of Python libraries to manage all stages of the machine learning pipeline. Data manipulation and preprocessing were handled using tools like Pandas, NumPy, and Scikit-learn's preprocessing functionalities, including Label Encoder and MinMaxScaler. Visualization was facilitated through Matplotlib, Seaborn, and Plotly Express, supporting both static and interactive data exploration. Models were developed using Scikit-learn's and evaluated using metrics such as accuracy, precision, F1 score, and recall. LIME was integrated to ensure model interpretability, making the results both effective and understandable. The entire process was executed within Jupyter Notebooks, offering a flexible environment for coding, visualization, and analysis.

3.3 Expected Outcomes

The experimental design of this project is structured to methodically assess various machine learning classifiers for breast cancer prediction, focusing on pinpointing critical predictors and augmenting model interpretability. These models are expected to excel in generalizing to new datasets while maintaining high computational efficiency. Utilizing innovative visualization tools and interpretability frameworks such as LIME ensures that the derived insights are both comprehensible and actionable. Overall, the project aims to deliver efficient, scalable solutions that provide reliable, actionable insights, thereby facilitating improved patient outcomes and healthcare strategies.

3.4 Ethical Considerations

In advancing machine learning techniques for cancer prediction and prognosis, maintaining high ethical standards is essential, particularly in how stakeholders such as patients and healthcare providers are affected. Key ethical considerations in this research include:

3.4.1 Data Privacy and Confidentiality: Patient data sourced from Kaggle are anonymized in compliance with privacy standards akin to HIPAA and GDPR, ensuring no individual patient can be identified.

3.4.2 Informed Consent: It's assumed that original data collection involved informed consent, where patients were fully briefed on the use of their data and their rights.

3.4.3 Bias and Fairness: Measures are implemented to mitigate biases in the data and algorithms that could affect predictions based on demographics, ensuring equitable outcomes.

3.4.4 Transparency and Interpretability: Machine learning models are transparent and interpretable, utilizing tools like LIME to make the decision-making process understandable and trustworthy.

3.4.5 Validation and Accountability: Rigorous validation checks are performed to ensure model accuracy and applicability, with clear communication of any limitations to uphold accountability.

3.4.6 Beneficence and Non-maleficence: Research is guided by the principles of doing good and avoiding harm, aiming to improve patient outcomes and contribute positively to cancer care.

These ethical considerations ensure the research is conducted responsibly, with a focus on enhancing patient care and trust in new technological applications in healthcare.

3.5 Design and Implementation

To navigate and harness the potential of this data, a meticulous Exploratory Data Analysis (EDA) is conducted. This initial phase is crucial, as it involves visualizing and statistically summarizing the data to understand its characteristics, such as the distribution of diagnosis categories and the variability of features. Such insights are vital for guiding the subsequent design and implementation of predictive models.

This study focuses on the application of multiple ML models, exploring their effectiveness through a rigorous framework that includes data preprocessing, feature selection, model training, and evaluation. Each model's performance is critically assessed using a variety of metrics including accuracy, precision, recall, and F1 score, along with more detailed analyses such as ROC curves and confusion matrices. Furthermore, the research incorporates advanced techniques such as Recursive Feature Elimination (RFE) for feature selection and Local Interpretable Model-agnostic Explanations (LIME) for enhancing model transparency. These methods not only improve the predictive performance but also ensure that the models are interpretable and trustworthy.

By following this structured methodology, the study aims to develop reliable and interpretable models that significantly advance the predictive capabilities in breast cancer prognosis, contributing to more personalized and effective treatment strategies.

3.6 Exploratory Data Analysis (EDA) and Processing

3.6.1 Dataset Overview

The dataset comprises data from 569 instances, each representing individual breast cancer cases. It is structured across 32 columns, detailing a mixture of clinical, demographic, and morphological variables associated with the breast cancer samples. These variables are meticulously recorded as floating-point numbers, integers, and categorical data types, facilitating a multifaceted analysis approach in the machine learning models developed in this study.

The dataset encompasses a comprehensive set of features (as shown in Figure 3) derived from digitized images of breast mass, organized into statistical categories to capture various aspects of cell nuclei characteristics. These features are categorized into means, standard errors (SE), and worst-case scenarios of each measurement, reflecting the central tendency, variability, and extreme values that are pivotal for the precise diagnosis of breast cancer.

```
# Assuming df_cancer is your DataFrame
column_labels = df_cancer.columns.tolist()  # Get list of column names
```

	Column 1	Column 2	Column 3
1	id	diagnosis	radius_mean
2	texture_mean	perimeter_mean	area_mean
3	smoothness_mean	compactness_mean	concavity_mean
4	concave points_mean	symmetry_mean	fractal_dimension_mean
5	radius_se	texture_se	perimeter_se
6	area_se	smoothness_se	compactness_se
7	concavity_se	concave points_se	symmetry_se
8	fractal_dimension_se	radius_worst	texture_worst
9	perimeter_worst	area_worst	smoothness_worst
10	compactness_worst	concavity_worst	concave points_worst
11	symmetry_worst	fractal_dimension_worst	None

Figure 3: Column Structure of the Dataset

3.6.2 Variables Description

ID: Each entry is uniquely identified by an integer value, ensuring the discrete handling and analysis of data points.

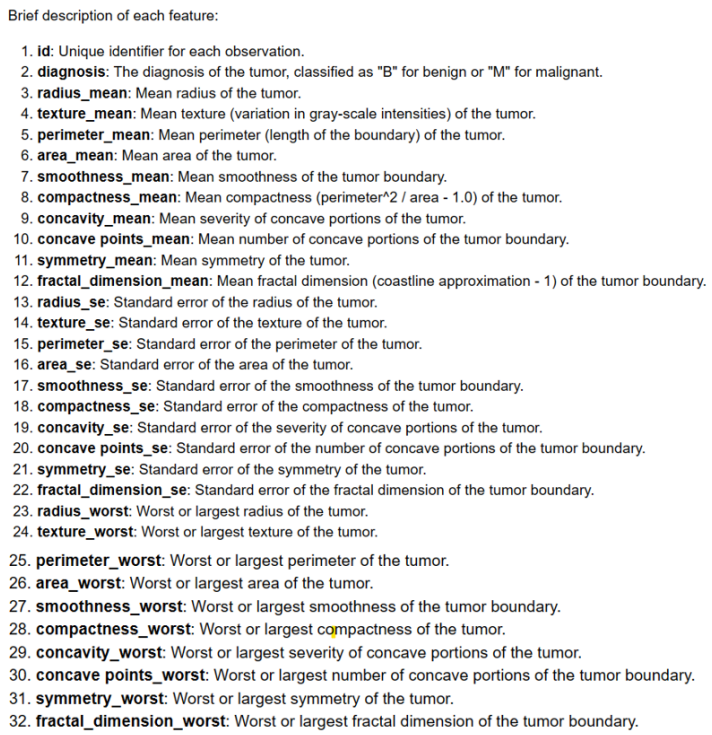
Diagnosis: Serving as the target variable, the diagnosis column categorizes the cancer tissues into Malignant (M) and Benign (B), providing the basis for supervised learning models.

Feature Variables: The dataset features 30 quantitative variables derived from the cell nuclei characteristics observed in digitized images of the breast mass. These variables are segmented into three distinct metrics:

Mean Features: Average values of the cell nuclei characteristics.

SE (Standard Error) Features: Variability measurements for the cell nuclei features.

Worst Features: Represent the most severe or largest values among the mean, standard error, and worst features for each characteristic, including radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension.



Brief description of each feature:

1. **id**: Unique identifier for each observation.
2. **diagnosis**: The diagnosis of the tumor, classified as "B" for benign or "M" for malignant.
3. **radius_mean**: Mean radius of the tumor.
4. **texture_mean**: Mean texture (variation in gray-scale intensities) of the tumor.
5. **perimeter_mean**: Mean perimeter (length of the boundary) of the tumor.
6. **area_mean**: Mean area of the tumor.
7. **smoothness_mean**: Mean smoothness of the tumor boundary.
8. **compactness_mean**: Mean compactness (perimeter^2 / area - 1.0) of the tumor.
9. **concavity_mean**: Mean severity of concave portions of the tumor.
10. **concave points_mean**: Mean number of concave portions of the tumor boundary.
11. **symmetry_mean**: Mean symmetry of the tumor.
12. **fractal_dimension_mean**: Mean fractal dimension (coastline approximation - 1) of the tumor boundary.
13. **radius_se**: Standard error of the radius of the tumor.
14. **texture_se**: Standard error of the texture of the tumor.
15. **perimeter_se**: Standard error of the perimeter of the tumor.
16. **area_se**: Standard error of the area of the tumor.
17. **smoothness_se**: Standard error of the smoothness of the tumor boundary.
18. **compactness_se**: Standard error of the compactness of the tumor.
19. **concavity_se**: Standard error of the severity of concave portions of the tumor.
20. **concave points_se**: Standard error of the number of concave portions of the tumor boundary.
21. **symmetry_se**: Standard error of the symmetry of the tumor.
22. **fractal_dimension_se**: Standard error of the fractal dimension of the tumor boundary.
23. **radius_worst**: Worst or largest radius of the tumor.
24. **texture_worst**: Worst or largest texture of the tumor.
25. **perimeter_worst**: Worst or largest perimeter of the tumor.
26. **area_worst**: Worst or largest area of the tumor.
27. **smoothness_worst**: Worst or largest smoothness of the tumor boundary.
28. **compactness_worst**: Worst or largest compactness of the tumor.
29. **concavity_worst**: Worst or largest severity of concave portions of the tumor.
30. **concave points_worst**: Worst or largest number of concave portions of the tumor boundary.
31. **symmetry_worst**: Worst or largest symmetry of the tumor.
32. **fractal_dimension_worst**: Worst or largest fractal dimension of the tumor boundary.

Figure 4: Screenshot-Feature Descriptions for Breast Cancer Dataset Used in Machine Learning Analysis

3.6.3 Descriptive Statistics

Descriptive statistics as shown in Figure 5 provide insights into feature distributions, variability, and data quality, aiding in understanding, comparison, and effective communication of key dataset characteristics.

```
# Drop the 'id' column from df_cancer
df_cancer_no_id = df_cancer.drop(columns='id')

# Generate descriptive statistics for df_cancer_no_id and transpose
descriptive_stats_transposed = df_cancer_no_id.describe().T
```

	count	mean	std	min	25%	50%	75%	max
radius_mean	569.000	14.127	3.524	6.981	11.700	13.370	15.780	28.110
texture_mean	569.000	19.290	4.301	9.710	16.170	18.840	21.800	39.280
perimeter_mean	569.000	91.969	24.299	43.790	75.170	86.240	104.100	188.500
area_mean	569.000	654.889	351.914	143.500	420.300	551.100	782.700	2501.000
smoothness_mean	569.000	0.096	0.014	0.053	0.086	0.096	0.105	0.163
compactness_mean	569.000	0.104	0.053	0.019	0.065	0.093	0.130	0.345
concavity_mean	569.000	0.089	0.080	0.000	0.030	0.062	0.131	0.427
concave points_mean	569.000	0.049	0.039	0.000	0.020	0.034	0.074	0.201
symmetry_mean	569.000	0.181	0.027	0.106	0.162	0.179	0.196	0.304
fractal_dimension_mean	569.000	0.063	0.007	0.050	0.058	0.062	0.066	0.097
radius_se	569.000	0.405	0.277	0.112	0.232	0.324	0.479	2.873
texture_se	569.000	1.217	0.552	0.360	0.834	1.108	1.474	4.885
perimeter_se	569.000	2.866	2.022	0.757	1.606	2.287	3.357	21.980
area_se	569.000	40.337	45.491	6.802	17.850	24.530	45.190	542.200
smoothness_se	569.000	0.007	0.003	0.002	0.005	0.006	0.008	0.031
compactness_se	569.000	0.025	0.018	0.002	0.013	0.020	0.032	0.135
concavity_se	569.000	0.032	0.030	0.000	0.015	0.026	0.042	0.396
concave points_se	569.000	0.012	0.006	0.000	0.008	0.011	0.015	0.053
symmetry_se	569.000	0.021	0.008	0.008	0.015	0.019	0.023	0.079
fractal_dimension_se	569.000	0.004	0.003	0.001	0.002	0.003	0.005	0.030
radius_worst	569.000	16.269	4.833	7.930	13.010	14.970	18.790	36.040
texture_worst	569.000	25.677	6.146	12.020	21.080	25.410	29.720	49.540
perimeter_worst	569.000	107.261	33.603	50.410	84.110	97.660	125.400	251.200
area_worst	569.000	880.583	569.357	185.200	515.300	686.500	1084.000	4254.000
smoothness_worst	569.000	0.132	0.023	0.071	0.117	0.131	0.146	0.223
compactness_worst	569.000	0.254	0.157	0.027	0.147	0.212	0.339	1.058
concavity_worst	569.000	0.272	0.209	0.000	0.114	0.227	0.383	1.252
concave points_worst	569.000	0.115	0.066	0.000	0.065	0.100	0.161	0.291
symmetry_worst	569.000	0.290	0.062	0.156	0.250	0.282	0.318	0.664
fractal_dimension_worst	569.000	0.084	0.018	0.055	0.071	0.080	0.092	0.208

Figure 5: Screenshot- Summary Statistics of Breast Cancer Dataset Features

3.6.4 Data Characteristics

The dataset's comprehensive coverage of cell nuclei features provides an in-depth examination of the morphological characteristics of breast cancer tissues. The statistical summary reveals varied scales and distributions across features, necessitating normalization steps to ensure models do not have bias towards variables with larger magnitudes.

3.6.5 Visualization of Target Variable and Features Distribution

The exploratory phase began with bar charts to display the frequency of each diagnosis category, providing a clear picture of the target variable's distribution. This was followed using histograms

and violin plots, which depicted the features' distributions and densities, shedding light on the data's behavior in relation to cancer prognosis. The seaborn and matplotlib libraries facilitated these visual explorations.

Figure 6 shows the bar chart that depicts the distribution of breast cancer diagnoses in the dataset, where 'B' stands for benign cases and 'M' for malignant cases. Significantly, 62.74% of the diagnoses are benign, whereas 37.26% are malignant. This visualization clearly emphasizes the higher incidence of benign cases, offering essential insights into the composition of the dataset.

Figure 7 presents a series of histograms with density curves, illustrating the distribution of various features categorized by diagnosis (benign: 0, malignant: 1). These features are grouped into mean, standard error (SE), and worst values.

The first row displays the distribution of mean values for radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. It is evident that malignant tumors (red) generally have higher mean values compared to benign tumors (blue) across most features.

The second row shows the standard error (SE) values for radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. Malignant tumors exhibit greater variability in these features, as indicated by the broader distribution and higher SE values compared to benign tumors.

The third row includes the worst values for radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. These plots demonstrate that the extreme values of these features are significantly higher in malignant tumors than in benign ones.

Overall, Figure 7 highlights the distinct differences in feature distributions between benign and malignant tumors. These insights are crucial for developing machine learning models aimed at accurately predicting and diagnosing breast cancer.

Figure 8 presents violin plots showing the distributions of various features by diagnosis, categorized into mean, standard error (SE), and worst values. Each plot illustrates the feature

values for benign (0) and malignant (1) diagnoses. These plots highlight differences in feature distributions between benign and malignant tumors, aiding in cancer prediction and diagnosis.

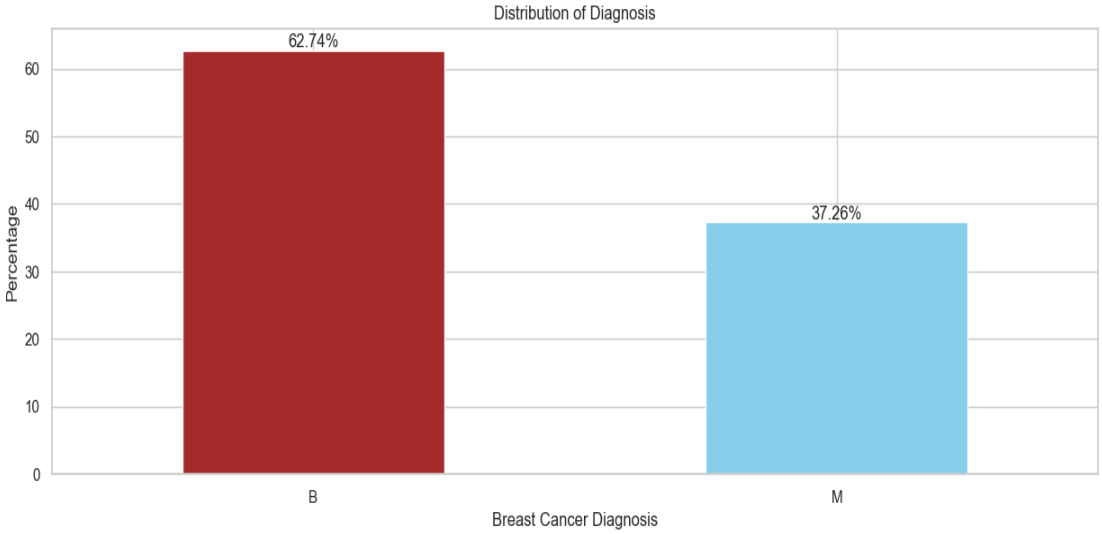


Figure 6: Bar Chart showing Distribution of Target Variable -Diagnosis

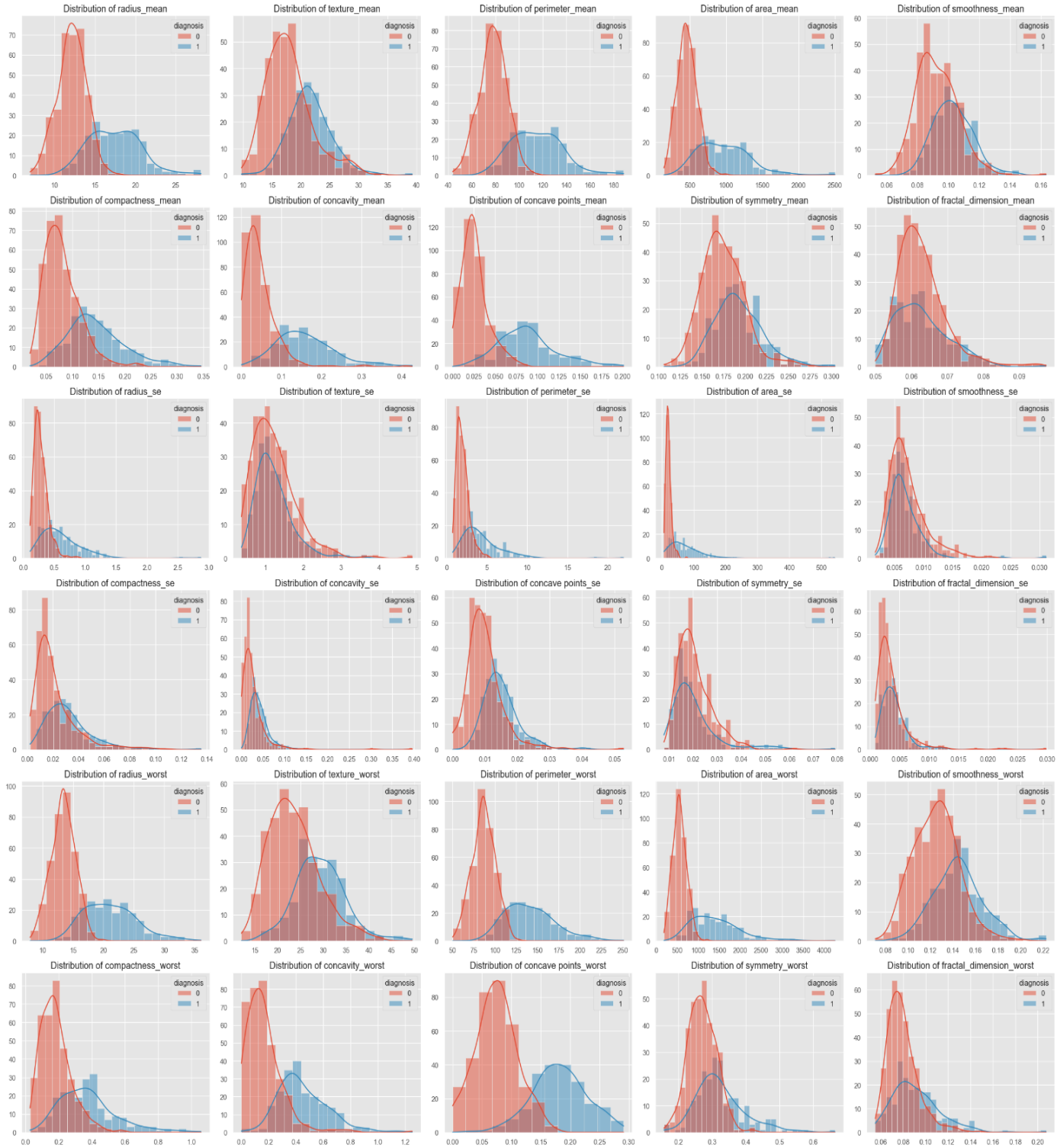


Figure 7 : Comparative Distribution of Breast Cancer Features by Diagnosis through Histogram

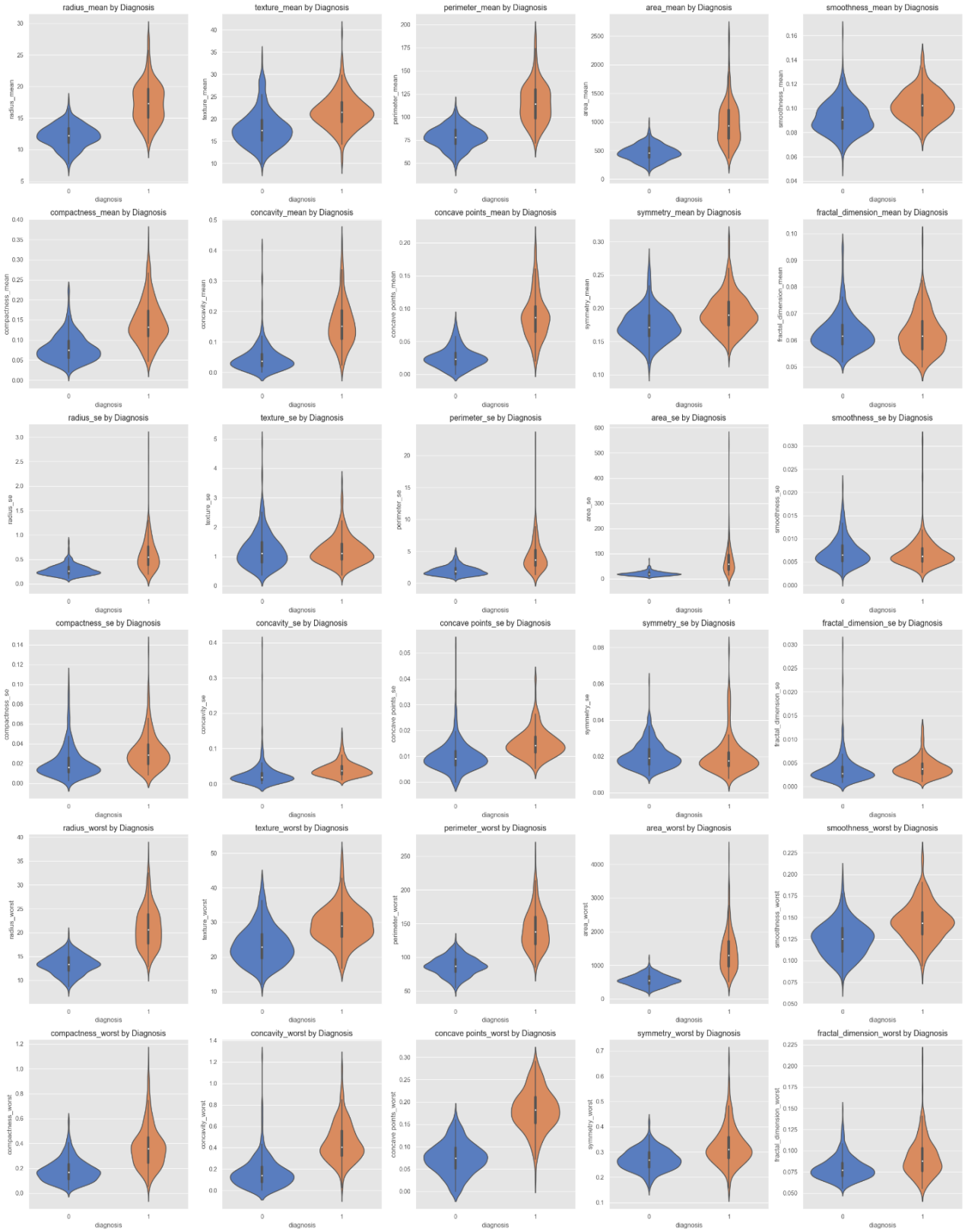


Figure 8 ：Violin Plot Showing Feature Distribution by Diagnosis in Breast Cancer Cases

3.6.6 Correlation Analysis

The correlation matrix in Figure 9 illustrates the degree of linear relationship between features, with values ranging from -1 to 1. Positive values indicate a direct relationship, while negative values indicate an inverse relationship. The heatmap uses a color scale to represent the strength of the correlations, where red indicates strong positive correlation, blue indicates strong negative correlation, and values near zero indicate weak or no correlation. This step was crucial in the feature selection process to exclude redundant predictors, ensuring that model performance was not adversely affected by highly correlated variables.

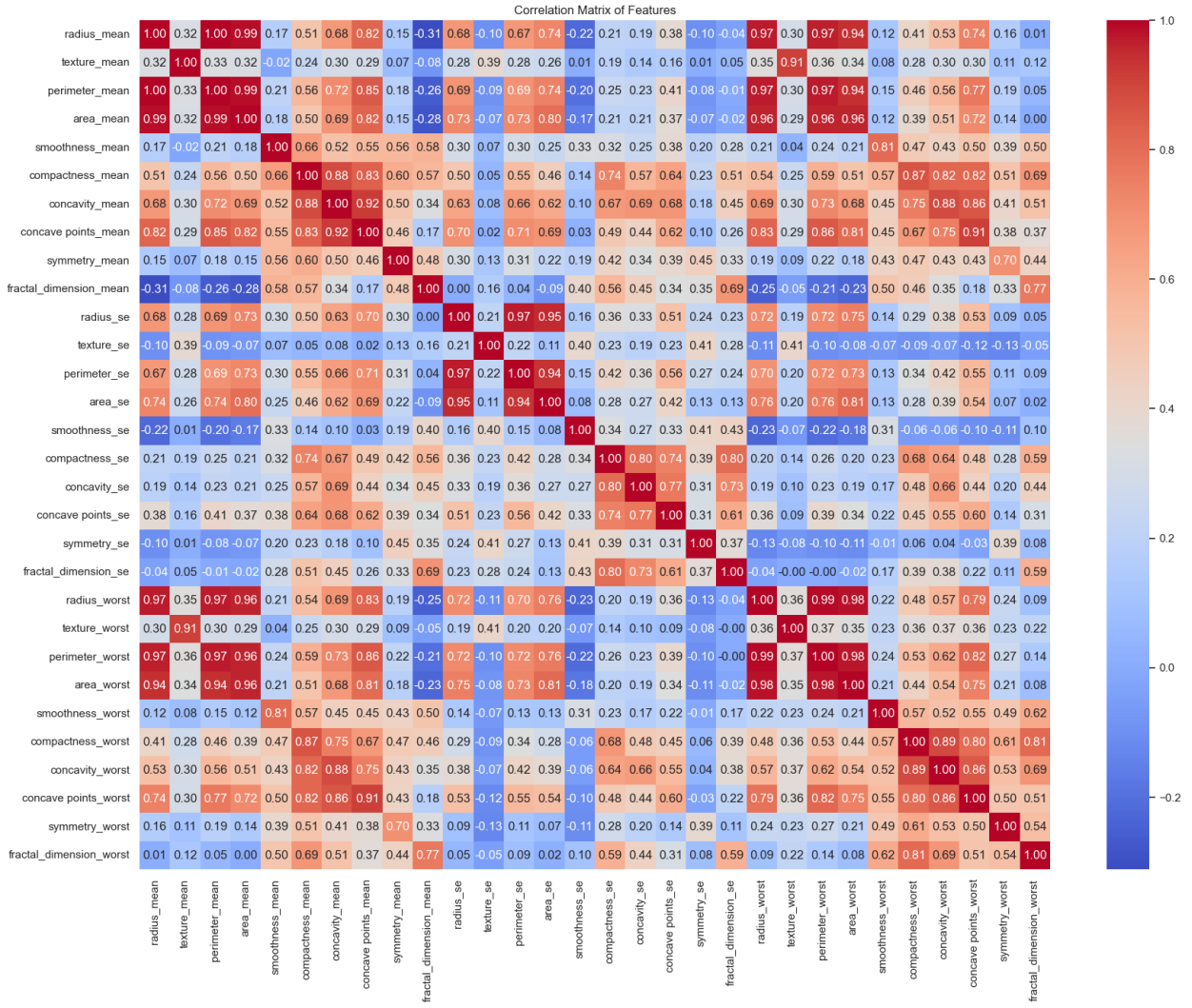


Figure 9 : Heatmap of Feature Correlation in Breast Cancer Dataset

3.7 Data Preprocessing

In preparation for the predictive modeling of breast cancer outcomes, the dataset underwent a rigorous preprocessing routine to ensure data quality and readiness for analysis. This multi-step process began with the cleansing of data to remove duplicates and address missing values, thereby preserving the integrity of the dataset. Numerical features were normalized or standardized to uniform scales to prevent model bias towards variables with larger scales.

Furthermore, the categorical target variable 'diagnosis' was encoded to facilitate its use in machine learning algorithms.

3.7.1 Data Cleaning

The dataset was subjected to a thorough data cleaning process to ensure the quality of the data for analysis (Figure 10). Upon inspection, it was found to be free of missing values, negating the need for imputation techniques. This lack of missing data points indicates a high level of completeness, which is beneficial for the reliability of the subsequent machine learning models.

```
# Checking for missing values
missing_values = df_cancer.isnull().sum()
# If there are no missing values, print a single statement
if not missing_values.any():
    print("No missing values found in the dataset.")
else:
    # If there are missing values, print them in a single row
    missing_values = missing_values[missing_values > 0]
    print("Missing values found in the following columns:")
    print(missing_values.to_frame().T)
```

```
No missing values found in the dataset.
```

Figure 10 :Code Snippet for Missing Value Analysis in the Breast Cancer Dataset

3.7.2 Categorical Variable Encoding

The 'diagnosis' column in the dataset was transformed from categorical to numeric format, mapping 'M' for malignant cases to 1 and 'B' for benign cases to 0, using the map function in pandas(Figure 11). This change is essential for facilitating the application of machine learning algorithms. Additionally, the 'id' column was removed from the dataset to eliminate non-informative data and focus the analysis on relevant features, ensuring that the model is not biased. The dataset was further organized by separating the independent variables (features, X) from the dependent variable (target, y). In this organization, 'X' includes all columns except 'diagnosis', and 'y' consists solely of the 'diagnosis' column, a crucial step for effectively training the model.

```
# Encode the 'diagnosis' column to numeric format, assuming 'M' for malignant and 'B' for benign
df_cancer['diagnosis'] = df_cancer['diagnosis'].map({'M': 1, 'B': 0})

# Ensure the 'id' column has been removed if it hasn't already
if 'id' in df_cancer.columns:
    df_cancer.drop('id', axis=1, inplace=True)

# Separate features and target
X = df_cancer.drop('diagnosis', axis=1)
y = df_cancer['diagnosis']
```

```
# Display the first few rows of the dataframe
df_cancer.head()
```

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean
0	1	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419
1	1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812
2	1	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069
3	1	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597
4	1	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809

Figure 11 : Initial Rows of Preprocessed Breast Cancer Dataset with Encoded Diagnosis

3.7.3 Feature Scaling

In the data preprocessing stage, feature scaling was conducted using the MinMaxScaler from the scikit-learn library. This normalization adjusted the dataset's features to a uniform scale, avoiding distortions in the value ranges. Each feature was scaled to a specified range, the default being 0 to 1, which is essential for algorithms sensitive to the magnitude of values. This step ensures unbiased treatment of all features during the predictive modeling process. Upon completion of scaling, the data was converted back to a pandas DataFrame format to maintain the convenient tabular structure for any further operations requiring DataFrame input.

```
from sklearn.preprocessing import MinMaxScaler

# Create a MinMaxScaler object
scaler = MinMaxScaler()

# Fit the scaler to the features and transform them
X_scaled = scaler.fit_transform(X)

# Convert the scaled numpy array back to a DataFrame
X_scaled = pd.DataFrame(X_scaled, columns=X.columns)
```

```
X_scaled.head()
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dim
0	0.521037	0.022658	0.545989	0.363733	0.593753	0.792037	0.703140	0.731113	0.686364	
1	0.643144	0.272574	0.615783	0.501591	0.289880	0.181768	0.203608	0.348757	0.379798	
2	0.601496	0.390260	0.595743	0.449417	0.514309	0.431017	0.462512	0.635686	0.509596	
3	0.210090	0.360839	0.233501	0.102906	0.811321	0.811361	0.565604	0.522863	0.776263	
4	0.629893	0.156578	0.630986	0.489290	0.430351	0.347893	0.463918	0.518390	0.378283	

Figure 12 :Normalizing Dataset Features with MinMaxScaler in Python

3.8 Establishing a Baseline for Model Performance with Full Feature Set: An Initial Train-Test Split

An initial model utilized the full feature set to establish a performance baseline. Subsequently, a streamlined model was crafted, employing feature selection to prioritize the most influential predictors. This strategic reduction aimed to bolster the model's generalizability beyond mere accuracy metrics. The train_test_split method from sklearn.model_selection segmented the data into an 80-20 train-test split, ensuring consistent and replicable divisions by fixing the random_state at 42. While this foundational split facilitated essential early assessments, the subsequent feature selection process was pivotal in enhancing the model's robustness, favoring long-term reliability and interpretability over a marginal reduction in accuracy. This trade-off reflects a preference for a more sustainable, efficient model operation conducive to real-world application, where simplicity and generalization hold paramount importance.

The corresponding code snippet is as shown in Figure 13:

```
# TrainTestSplit

from sklearn.model_selection import train_test_split

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)
```

```
print("Training set shapes:", X_train.shape, y_train.shape)
print("Testing set shapes:", X_test.shape, y_test.shape)
```

```
Training set shapes: (455, 30) (455,)
Testing set shapes: (114, 30) (114,)
```

Figure13: Data Division for Baseline Model Evaluation: Initial Train-Test Split Configuration

3.8.1 Model Training, Prediction and Evaluation Framework

The preliminary phase involved training models using the dataset's full range of features. These models were then systematically evaluated, not only through standard metrics such as accuracy, precision, recall, and F1 score but also by analyzing the confusion matrix for insights into true versus predicted classifications. Furthermore, the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) score were employed to assess the models' true positive rate against the false positive rate, providing a comprehensive understanding of model performance.

3.8.2 Logistic Regression Model Implementation

As shown in Figure 14, through the Scikit-learn LogisticRegression class, the model is trained on selected data samples. This phase adjusts the model's parameters to correlate the input features with the target classification—malignant or benign.

Post-training, the Logistic Regression model is evaluated against unseen data, using metrics such as accuracy, precision, recall, and the F1 score. These indicators collectively measure the model's predictive accuracy and its balance between sensitivity and specificity.

```
from sklearn.metrics import classification_report, confusion_matrix, roc_curve, roc_auc_score
import matplotlib.pyplot as plt
import seaborn as sns

def train_logistic_regression(X_train, y_train, X_test, y_test):
    # Build the model
    model = LogisticRegression()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)

   # Show Classification Report
   print("Classification Report for Logistic Regression:")
   print(classification_report(y_test, y_pred))

   # Visualize confusion matrix
   plt.figure(figsize=(8, 6))
   sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
   plt.xlabel('Predicted Labels')
   plt.ylabel('True Labels')
   plt.title('Confusion Matrix for Logistic Regression')
   plt.show()

   # Visualize the model using ROC Curve
   fpr_lr, tpr_lr, thresholds_lr = roc_curve(y_test, y_pred)
   auc_lr = roc_auc_score(y_test, y_pred)
```

Figure 14: Logistic Regression Model Training, Performance Metrics, and Visualization Code

3.8.3 Support Vector Machine Model Implementation

The SVM algorithm takes the feature vectors from the dataset. Each vector represents a set of measurements for a single sample, like a cell nucleus. During training, the SVM uses these vectors to determine the optimal separating hyperplane. It employs a kernel, like the linear or RBF(Radial Basis Function) kernel, to handle the non-linear nature of the data if necessary.

Key data points (support vectors) that are closest to the hyperplane influence its position and orientation. The SVM learns from these to maximize the margin between the classes.

Post-training, the model classifies new data points based on their position relative to the hyperplane. In the context of breast cancer data, this means identifying whether the measurements for a new cell nucleus sample are more like previous benign or malignant samples(Figure 15).

```
def train_svm(X_train, y_train, X_test, y_test):
    # Build the model
    model = SVC()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
```

```
    # Show Classification Report
    print("Classification Report for SVM:")
    print(classification_report(y_test, y_pred))

    # Visualize confusion matrix
    plt.figure(figsize=(8, 6))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title('Confusion Matrix for SVM')
    plt.show()

    # Visualize the model using ROC Curve
    fpr_svm, tpr_svm, thresholds_svm = roc_curve(y_test, y_pred)
    auc_svm = roc_auc_score(y_test, y_pred)
```

Figure 15: SVM Model Training, Performance Metrics, and Visualization Code

3.8.4 Random Forest Model Implementation

Specifically, the Random Forest algorithm is trained using a subset of the data (X_train, y_train), where it learns to correlate the input features with the target labels, either benign or malignant(Figure 16). Predictions are then made on a distinct test subset (X_test), providing insights into the model's potential accuracy on data it has not previously encountered. This process allows for a robust estimation of the model's performance in real-world scenarios.

```python
def train_random_forest(X_train, y_train, X_test, y_test):
    # Build the model
    model = RandomForestClassifier()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
```

```python
    # Show Classification Report
    print("Classification Report for Random Forest Classifier:")
    print(classification_report(y_test, y_pred))

    # Visualize confusion matrix
    plt.figure(figsize=(8, 6))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title('Confusion Matrix for Random Forest Classifier')
    plt.show()

    # Visualize the model using ROC Curve
    fpr_rf, tpr_rf, thresholds_rf = roc_curve(y_test, y_pred)
    auc_rf = roc_auc_score(y_test, y_pred)
```

Figure 16: Random Forest Classifier Training, Performance Metrics, and Visualization Code

3.8.5 Decision Tree Model Implementation

A Decision Tree Classifier from the sklearn.tree module is instantiated and trained on the training data (X_train, y_train). The decision tree learns by recursively splitting the training set into smaller subsets based on the feature that results in the highest information gain or the greatest reduction in impurity. This process continues until the nodes are pure (contain only one class) or until a stopping criterion is met (such as a maximum depth of the tree).

Once trained, the model uses the decision tree to predict the labels (y_pred) for the test dataset (X_test). It follows the decision rules derived from the training data to arrive at predictions. The code is as shown in Figure 17.

```python
def train_decision_tree(X_train, y_train, X_test, y_test):
    # Build the model
    model = DecisionTreeClassifier()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
```

```python
    # Show Classification Report
    print("Classification Report for Decision Tree Classifier:")
    print(classification_report(y_test, y_pred))

    # Visualize confusion matrix
    plt.figure(figsize=(8, 6))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title('Confusion Matrix for Decision Tree Classifier')
    plt.show()

    # Visualize the model using ROC Curve
    fpr_dt, tpr_dt, thresholds_dt = roc_curve(y_test, y_pred)
    auc_dt = roc_auc_score(y_test, y_pred)
```

Figure 17: Decision Tree Classifier Training, Performance Metrics, and Visualization Code

3.8.6 K- Nearest Neighbor Model Implementation

The KNN classifier was trained using the Scikit-learn library's KNeighborsClassifier. Unlike model-based learning, KNN functions by memorizing the entire training dataset. During training, the model stores the feature vectors and corresponding labels from X_train and y_train.

For predictions, KNN identifies the K closest neighbors to a query point based on a distance metric (typically Euclidean distance). The classification of a test sample in X_test is determined by a majority vote among its nearest neighbors, with the most frequent class label from the K-nearest neighbors being assigned to the test instance (Figure 18).

```python
def train_knn(X_train, y_train, X_test, y_test):
    # Build the model
    model = KNeighborsClassifier()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
```

```python
    # Show Classification Report
    print("Classification Report for KNN:")
    print(classification_report(y_test, y_pred))

    # Visualize confusion matrix
    plt.figure(figsize=(8, 6))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title('Confusion Matrix for KNN')
    plt.show()

    # Visualize the model using ROC Curve
    fpr_knn, tpr_knn, thresholds_knn = roc_curve(y_test, y_pred)
    auc_knn = roc_auc_score(y_test, y_pred)
```

Figure 18: KNN Classifier Training, Performance Metrics, and Visualization Code

3.8.7 XGBoost Neighbor Model Implementation

An XGB Classifier from the XGBoost library is initialized and trained using the training dataset (X_train, y_train). XGBoost optimizes both the computational speed and model performance, utilizing a gradient boosting framework that builds trees sequentially, where each new tree helps to correct errors made by previously built trees.

After training, the model employs the ensemble of trees to predict the outcomes for the test dataset (X_test). Each tree gives a prediction (vote) for each class, and XGBoost combines these predictions to give the final output based on the majority vote (Figure 19).

```python
def train_xgboost(X_train, y_train, X_test, y_test):
    # Build the model
    model = XGBClassifier()
    model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = model.predict(X_test)

    # Compute performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
```

```python
    # Show Classification Report
    print("Classification Report for XGBoost:")
    print(classification_report(y_test, y_pred))

    # Visualize confusion matrix
    plt.figure(figsize=(8, 6))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, cmap='Blues', fmt='g')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title('Confusion Matrix for XGBoost')
    plt.show()

    # Visualize the model using ROC Curve
    fpr_xgb, tpr_xgb, thresholds_xgb = roc_curve(y_test, y_pred)
    auc_xgb = roc_auc_score(y_test, y_pred)
```

Figure 19: XGBoost Classifier Training, Performance Metrics, and Visualization Code

3.8.8 Comparative Performance Report of Machine Learning Models

Each model was implemented using Python, with key snippets of code provided to demonstrate the setup and evaluation phases. The models were trained and tested on the dataset, with the following steps uniformly applied to each model:

a) **Model Initialization**: Setting up each model with appropriate hyperparameters.

b) **Model Training**: Fitting the model on the training set.

c) **Prediction**: Running the model against the test set to generate predictions.

d) **Performance Evaluation**: Calculating accuracy, precision, recall, and F1 score.

3.9 Evaluation Criteria

To effectively assess the performance of the predictive models developed for this breast cancer dataset, several evaluation metrics were employed, each chosen for its relevance to the classification task at hand. This dataset includes measurements from breast cancer diagnoses (benign or malignant) and features derived from digitized images of fine needle aspirates of breast masses. Understanding and predicting the diagnosis accurately is critical, hence the selection of the following metrics.

3.9.1 Accuracy

Measures the overall correctness of the model's predictions.

Mathematically, $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

Where:

TP (True Positives) is the number of positive samples correctly identified by the model.

TN (True Negatives) is the number of negative samples correctly identified by the model.

FP (False Positives) is the number of negative samples incorrectly identified as positive.

FN (False Negatives) is the number of positive samples incorrectly identified as negative.

3.9.2 Precision

Precision quantifies the accuracy of positive predictions.

Mathematically:

$$Precision = \frac{TP}{TP+FP}$$

3.9.3 Recall

Recall measures the model's ability to identify all relevant instances accurately.

Mathematically:

$$Recall = \frac{TP}{TP+FN}$$

3.9.4 F1 Score

Balances precision and recall, providing a single metric to assess a model's performance.

$$F1\ Score = \frac{2 \text{ x (precision x Recall)}}{\text{Recall} + \text{Precision}}$$

```
results = {
    'Model': ['Logistic Regression', 'SVM', 'Random Forest Classifier', 'Decision Tree Classifier', 'KNN', 'XG-Boost'],
    'Accuracy': [f"{accuracy_lr:.2f}", f"{accuracy_svm:.2f}", f"{accuracy_rf:.2f}", f"{accuracy_dt:.2f}", f"{accuracy_knn:.2f}",
    'Precision': [f"{precision_lr:.2f}", f"{precision_svm:.2f}", f"{precision_rf:.2f}", f"{precision_dt:.2f}", f"{precision_knn:.
    'Recall': [f"{recall_lr:.2f}", f"{recall_svm:.2f}", f"{recall_rf:.2f}", f"{recall_dt:.2f}", f"{recall_knn:.2f}", f"{xg_recall
    'F1 Score': [f"{f1_lr:.2f}", f"{f1_svm:.2f}", f"{f1_rf:.2f}", f"{f1_dt:.2f}", f"{f1_knn:.2f}", f"{xg_f1:.2f}"]
}
```

```
# Convert scores to a DataFrame
results_df = pd.DataFrame(results)

results_df
```

Figure 20: Screenshot Code-Comparison of Model Performance Metric

3.10 Visualization of Model Performance

The classification efficacy is conveyed visually through:

3.10.1 Confusion Matrix

A confusion matrix in machine learning is a table used to evaluate the performance of a classification algorithm. Rows represent actual classes, and columns represent predicted classes, allowing for the identification of errors like false positives and false negatives, as well as correct predictions. It helps in assessing the model's accuracy, precision, recall, and overall performance (Krstinić et al., 2020).

3.10.2 ROC Curve and AUC Score

The ROC Curve plots the true positive rate against the false positive rate, offering a graphical representation of the model's discriminative threshold. The AUC Score quantifies this discrimination, with a score of 1.0 denoting perfect classification and 0.5 suggesting no discriminative ability (Bhandari, 2024).

3.11 Feature Selection

Feature selection is a fundamental part of the machine learning workflow which helps in enhancing the efficiency and effectiveness of the predictive models. By intelligently reducing the number of input variables, it ensures that the computational complexity is managed while maximizing the performance of the model (Dhal and Azad, 2022).

The model development workflow was notably improved with the integration of feature selection, focused on honing the model's structure, minimizing its intricacy, and enhancing accuracy on future data. In this study, Recursive Feature Elimination (RFE), combined with a logistic regression classifier, was employed to determine the essential features for the model.

3.11.1 Justification for using Feature Selection

It improves model accuracy by focusing training on significant features and speeds up the training process. Additionally, it increases model interpretability, making it easier to understand and more reliable, particularly in critical decision-making areas like healthcare .

3.11.2 Recursive Feature Elimination (RFE)

RFE is an iterative process that begins with all available features and systematically removes the least significant feature at each step (Brownlee, 2020). The criterion for removal is based on the feature's weight magnitude in the model, with smaller weights indicating less importance.

3.11.3 Implementation Process for RFE

A Logistic Regression model was initialized to serve as the base estimator for RFE (Figure 21). The RFE algorithm was configured to select the top 15 features from the full set, removing one feature per iteration (step=1).

RFE fitting was performed using the training data (X_train, y_train).

```
from sklearn.feature_selection import RFE

# Initialize a Logistic Regression
lr = LogisticRegression()

# Initialize RFE with the Logistic Regression classifier
rfe = RFE(estimator=lr, n_features_to_select=15, step=1)

# Fit RFE to the training data
rfe.fit(X_train, y_train)

# Get the selected features
selected_features = X_train.columns[rfe.support_]

# Transform the training and testing data to include only the selected features
X_train_selected = rfe.transform(X_train)
X_test_selected = rfe.transform(X_test)
```

Figure 21: Feature Selection Using Recursive Feature Elimination with Logistic Regression

3.11.4 Result from RFE Implementation

Post-RFE, the algorithm identified 15 key features deemed most predictive of the target variable. The selected features were used to transform both training and testing datasets, yielding X_train_selected and X_test_selected. The selected features are listed in Figure 22.

Figure 23 presents the training set consisting of 455 samples with 15 features each, while the testing set consists of 114 samples with 15 features each. This division ensures that the model has adequate data for training while maintaining a separate set for evaluating its performance.

```
selected_features.tolist()
```

```
['radius_mean',
 'texture_mean',
 'perimeter_mean',
 'area_mean',
 'concavity_mean',
 'concave points_mean',
 'radius_se',
 'radius_worst',
 'texture_worst',
 'perimeter_worst',
 'area_worst',
 'smoothness_worst',
 'concavity_worst',
 'concave points_worst',
 'symmetry_worst']
```

Figure 22: Selected Feature Set After Recursive Feature Elimination

```
print("Training set shapes:", X_train_selected.shape, y_train.shape)
print("Testing set shapes:", X_test_selected.shape, y_test.shape)
```

```
Training set shapes: (455, 15) (455,)
Testing set shapes: (114, 15) (114,)
```

Figure 23: Displaying Shapes of Selected Features in Training and Testing Datasets

3.11.5 Implications for Model Building

The subset of features selected by RFE is not restricted to logistic regression and can be utilized for constructing various other predictive models. The rationale is that these refined datasets will allow any subsequent model to focus on the most informative attributes, enhancing generalization and interpretability while potentially reducing overfitting and computational expense.

3.11.6 Classifier Evaluation After Feature Selection

As part of the optimization process, the same array of machine learning classifiers was systematically trained and evaluated post-feature selection (Figure 24) to ensure model robustness and performance efficiency.

```
classifiers = {
     "Logistic Regression": LogisticRegression(),
     "SVM": SVC(probability=True),
     "RFC": RandomForestClassifier(),
     "DT": DecisionTreeClassifier(),
     "KNN": KNeighborsClassifier(),
     "XG-Boost": XGBClassifier(),
}
```

```
# Define a dictionary to store scores before and after feature selection
scores_after = {}
```

```
# Train and evaluate each model after feature selection
for name, classifier in classifiers.items():
    # Train the classifier
    classifier.fit(X_train_selected, y_train)

    # Predict on the test set
    y_pred = classifier.predict(X_test_selected)

    # Compute performance metrics after feature selection
    accuracy = round(accuracy_score(y_test, y_pred), 2)
    precision = round(precision_score(y_test, y_pred), 2)
    recall = round(recall_score(y_test, y_pred), 2)
    f1 = round(f1_score(y_test, y_pred), 2)

    # Store scores after feature selection
    scores_after[name] = {'Accuracy': accuracy, 'Precision': precision, 'Recall': recall, 'F1 Score': f1}
```

```
# Convert scores to DataFrames

scores_after_df = pd.DataFrame(scores_after).T
```

Figure 24: Evaluating Machine Learning Classifiers Post-Feature Selection

3.12 Cross Validation and Grid Search

A methodical approach was adopted for each classifier in employing GridSearchCV to perform grid search with 5-fold cross-validation. The evaluation metric set was 'accuracy'.

Grid Search: Each classifier was subjected to grid search to ascertain the best hyperparameters from the predefined param_grids.

Classifiers were trained using the full feature set (X_train) and process was repeated with classifiers trained on a feature-selected subset (X_train_selected).The trained models were used to make predictions on the corresponding test sets (X_test and X_test_selected). This methodology ensures that the models are not only tuned to the training data but also validated through cross-validation, resulting in models that are robust and likely to generalize well to new data.

In Figure 25, the first code snippet demonstrates the grid search with cross-validation using all features, while the second code snippet shows the same process using only the selected features.

```python
# Perform grid search with cross-validation for each model using all features
for name, classifier in classifiers.items():
    param_grid = param_grids[name]
    grid_search = GridSearchCV(classifier, param_grid, cv=5, scoring='accuracy')
    grid_search.fit(X_train, y_train)

    # Predict on the test set
    y_pred = grid_search.predict(X_test)

    # Compute performance metrics
    accuracy = round(accuracy_score(y_test, y_pred), 2)
```

```python
# Perform grid search with cross-validation for each model using selected features
for name, classifier in classifiers.items():
    param_grid = param_grids[name]
    grid_search = GridSearchCV(classifier, param_grid, cv=5, scoring='accuracy')
    grid_search.fit(X_train_selected, y_train)

    # Predict on the test set
    y_pred = grid_search.predict(X_test_selected)

    # Compute performance metrics
    accuracy = round(accuracy_score(y_test, y_pred), 2)
```

Figure 25: Model Optimization and Evaluation with Grid Search Cross-Validation

3.13 Variable Significance

3.13.1 Feature Importance Analysis

Feature importance analysis aims to identify and rank the importance of input variables (features) that contribute to the predictive power of a model across the entire dataset.

Each model is fitted to the training data (X_train, y_train). Depending on the model type, the relevant feature importance scores are extracted and visualized using a custom plotting function (plot_feature_importance). This function generates a bar chart depicting the weights or importance of the top features.

Post-extraction, feature importance metrics are cataloged, offering insights into which attributes most strongly influence the model's predictions and the common significant feature(s) across the models. Figure 26, the code snippet shows how feature importance is calculated for the models.

```python
# Perform feature importance analysis for each model using all features
for name, classifier in classifiers.items():
    classifier.fit(X_train, y_train)
    if name == 'SVM':  # SVM doesn't provide feature importance directly, so we skip it
        continue
    if hasattr(classifier, 'coef_'):  # If the model has 'coef_' attribute, it's a linear model
        feature_importance = classifier.coef_[0]
        plot_feature_importance(classifier, X_train.columns, f'{name} Feature Importance')
    elif hasattr(classifier, 'feature_importances_'):  # If the model has 'feature_importances_', it's a tree-based model
        feature_importance = classifier.feature_importances_
        plot_feature_importance(classifier, X_train.columns, f'{name} Feature Importance')

    # Store feature importance scores in the dictionary
    feature_importance_dict[name] = dict(zip(X_train.columns, feature_importance))
```

...

```python
# Find top common features
top_common_features = Counter()
```

Figure 26: Screenshot Feature Importance Across Multiple Machine Learning Models

3.14 Model Interpretability- LIME(Local Interpretable Model-agnostic Explanations) Analysis for Model Predictions

LIME provides explanations for individual predictions. It aims to explain why a model made a specific prediction for a single instance, making the model's operation transparent at a local level.

3.14.1 Implementation Framework

The project integrates XAI-LIME (Local Interpretable Model-agnostic Explanations), a technique pivotal in explicating individual predictions from a collection of classification models. The LIME framework is particularly adept at providing interpretability for models where global explanations may be too complex or opaque.

3.14.2 Process Description for LIME

An instance of LimeTabularExplainer is established, fine-tuned for tabular datasets used in classification tasks. This explainer utilizes the training data, X_train, along with the feature names, to provide detailed information throughout the explanation process.

3.14.3 Analytical Function for LIME

A specialized function, lime_analysis, is deployed to operationalize the LIME analysis for any classifier. The function is designed to:

- Employ the explainer to generate an explanation instance, which explains the rationale behind a specific prediction, denoting the influence of each feature.
- Present the explanation directly within the notebook, as well as in a pyplot figure format for enhanced visualization and interpretability.

3.14.4 LIME Execution Across Classifiers

The lime_analysis function is systematically applied across each classifier, with the process being reiterated for the same data point to ensure consistency in comparative analysis. This allows for a uniform assessment of feature impact across multiple models.

Incorporating LIME into the design implementation phase ensures that each model's decision process can be transparently communicated, meeting the requirements for accountable AI systems. This approach not only builds trust in the predictions but also aligns with best practices for model interpretability.

In Figure 27, the code initializes the LIME explainer and defines a function to perform LIME analysis for each model. The analysis is conducted on the test data to generate explanations for the model's predictions, enhancing the understanding of how each feature contributes to the final output.

```
from lime.lime_tabular import LimeTabularExplainer
```

```
# Initialize LIME explainer
explainer = LimeTabularExplainer(X_train.values, mode='classification', feature_names=X_train.columns, verbose=True)
```

```
# Function to perform LIME analysis for each model
def lime_analysis(model, name, X_instance):
    exp = explainer.explain_instance(X_instance, model.predict_proba)
    exp.show_in_notebook(show_table=True, show_all=False)
    exp.as_pyplot_figure()
```

```
# Perform LIME analysis for each model using all features
for name, classifier in classifiers.items():
    print(f"LIME ANALYSIS FOR: {name}")
    lime_analysis(classifier, name, X_test.iloc[1])
```

Figure 27: Implementation of LIME for Model Interpretability in Classification

4. Chapter Four: Results ,Analysis and Discussion

This study rigorously evaluates multiple machine learning classifiers—including Logistic Regression, Support Vector Machine, Random Forest, K-Nearest Neighbor, Decision Tree, and Ensemble Methods—for their effectiveness in breast cancer prediction. The analysis focuses on identifying the most accurate classifiers, determining key predictive variables through feature importance analysis, and assessing each model's performance in policy-making contexts for breast cancer management. Additionally, it examines the interpretability of these classifiers through LIME analysis, aiming to make their predictions more transparent and clinically actionable. The subsequent sections will delve into these aspects, presenting a synthesis of the findings.

Each model was evaluated using key performance metrics: Accuracy, Precision, Recall, and F1-Score to assess overall performance. The ROC-AUC Curve was used to analyze the sensitivity and specificity of the classifiers, while Confusion Matrices provided a detailed breakdown of each classifier's predictive capabilities.

a) Accuracy: The proportion of all predictions that are correct, measuring the model's overall correctness.
b) Precision: The proportion of positive identifications that were correct, important for minimizing false positives.
c) Recall: The proportion of actual positives that were correctly identified, crucial for capturing as many true cases as possible.
d) F1 Score: The harmonic mean of precision and recall, providing a single metric for balance between the two when both are equally important.

4.1 Result - Baseline for Model Performance with Full Features Set

Establishing a baseline performance using the full feature set from the breast cancer dataset is essential before exploring complex model configurations and feature reduction techniques. This initial analysis is critical for several reasons:

Benchmarking: It provides a reference point to compare against simplified feature sets, helping to quantify the effects of further optimizations.

Model Comparison: Applying each classifier to a comprehensive set of features ensures a fair assessment of their predictive abilities under uniform conditions. This process highlights the strengths and weaknesses of each model given the full information available.

Feature Efficacy: Using all available features helps evaluate the dataset's overall potential in predicting breast cancer outcomes. It also prepares the ground for later analyses to determine which features are most impactful to accuracy and which might be redundant.

This section presents the results from baseline classifiers This evaluation not only sets a foundation for the study but also points to opportunities for enhancing model performance through targeted tuning and feature selection in future phases of analysis.

4.1.1 Comparative Analysis of the Models' Performance on Full Features Set

Figure 28 provides a comprehensive comparison of the performance metrics for the six machine learning models:

Logistic Regression exhibited exceptional performance, achieving the highest Accuracy (0.98) and the perfect Precision (1.00) among all models. Its Recall was 0.95, indicating that it correctly identified 95% of all positive cases. The model also had the highest F1 Score (0.98), reflecting an excellent balance between Precision and Recall. SVM followed closely, with an Accuracy of 0.97 and Precision of 0.98. Its Recall, like Logistic Regression, was also 0.95, which contributed to a strong F1 Score of 0.96. SVM's consistency across metrics makes it a robust choice for various classification needs.

Random Forest Classifier showed strong Precision (0.98) but a slightly lower Recall of 0.93 compared to the top two models. It achieved an Accuracy of 0.96 and an F1 Score of 0.95, highlighting its effectiveness, particularly in handling complex data structures with multiple features .KNN also recorded an Accuracy of 0.96 but with Precision and Recall slightly lower at 0.95 each. Its F1 Score matched that of Random Forest at 0.95, indicating good but not optimal performance across both Precision and Recall.

XG-Boost mirrored KNN in Accuracy (0.96) and Recall (0.93) but had a slightly lower Precision of 0.95, resulting in an F1 Score of 0.94. This model is known for its efficiency in processing and might be preferred for large datasets or speed-critical applications. The Decision Tree Classifier

had the lowest metrics across the board with an Accuracy of 0.95, Precision of 0.93, and Recall of 0.93. The equal Recall and Precision resulted in an F1 Score of 0.93, suggesting that while effective to a degree, it might not perform as well under more rigorous or variable conditions compared to the other models.

The evaluation reveals Logistic Regression as the top performer in this specific dataset, with SVM and Random Forest also showing strong capabilities. Decision Tree, while less robust in this scenario, may still be useful with appropriate adjustments.

	Model	Accuracy	Precision	Recall	F1 Score
0	Logistic Regression	0.98	1.00	0.95	0.98
1	SVM	0.97	0.98	0.95	0.96
2	Random Forest Classifier	0.96	0.98	0.93	0.95
3	Decision Tree Classifier	0.95	0.93	0.93	0.93
4	KNN	0.96	0.95	0.95	0.95
5	XG-Boost	0.96	0.95	0.93	0.94

Figure 28: Screenshot- Summary of Models' Performance on Full Features Set

Figure 29 visually compares six machine learning models across Accuracy, Precision, Recall, and F1 Score. By presenting the data in this format, it becomes easier to discern which models are most reliable and balanced, aiding in the selection of the optimal model for specific applications.

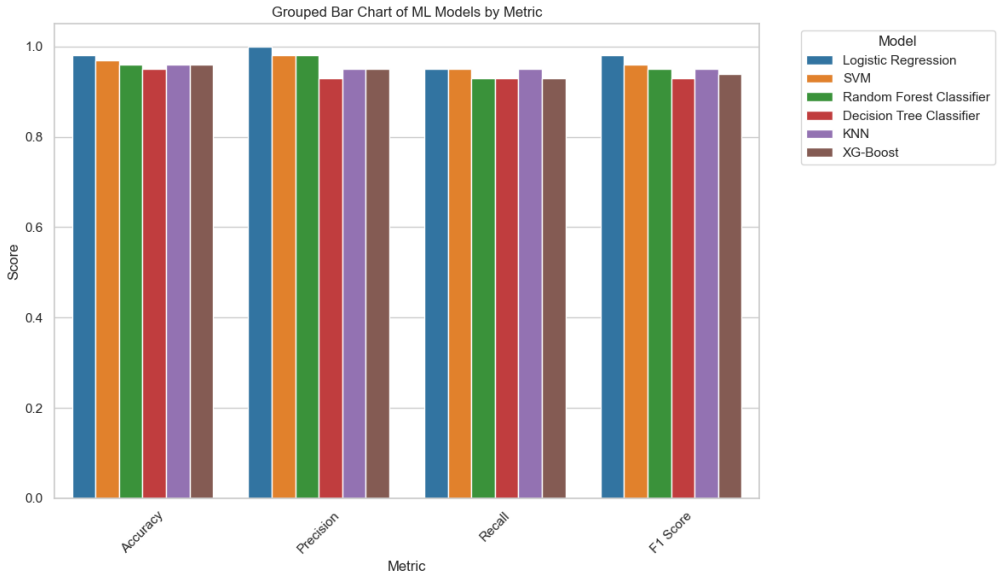


Figure 29: Representation of Machine Learning Model Efficacy Across Four Metrics

4.1.2 Comparative Presentation of Model Performance: Confusion Matrix Analysis

As represented by Figure 30 below, Logistic Regression performs exceptionally well with 71 true negatives and 41 true positives, achieving high accuracy with no false negatives and only 2 false positives. SVM reports similarly high accuracy with 70 true negatives and 41 true positives, slightly more false positives (1), and the same false negatives (2) as Logistic Regression.

Random Forest shows robust but slightly less sensitive results with 70 true negatives, 40 true positives, 3 false negatives, and 1 false positive. Decision Tree presents variability with 68 true negatives and 40 true positives, and increased error rates with 3 false positives and 3 false negatives. KNN maintains comparable performance with 69 true negatives and 41 true positives, alongside 2 false positives and 2 false negatives. XGBoost also delivers strong results with 69 true negatives and 40 true positives, 2 false positives, and 3 false negatives.

These confusion matrices underscore each model's diagnostic accuracy and highlight the crucial trade-offs between sensitivity (recall) and specificity (true negative rate). This comparative analysis aids in selecting an optimal model for clinical deployment, emphasizing the need for a balance in detecting and accurately classifying medical conditions to minimize potential harm in clinical decision-making.

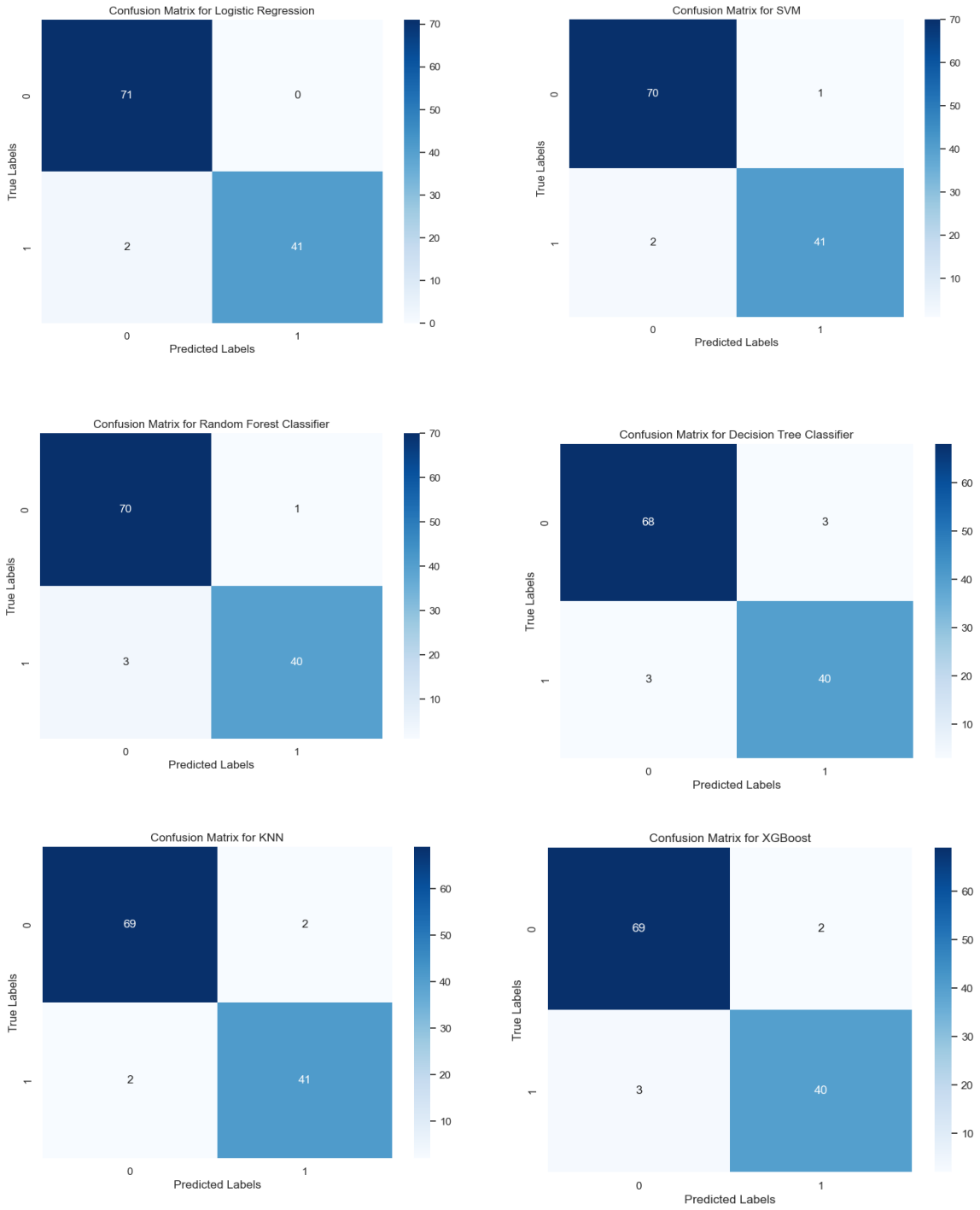


Figure 30: Performance of ML Classifiers in Breast Cancer Prediction: Confusion Matrices Overview

4.1.3 ROC Curve Analysis across Machine Learning Models

Figure 31 presents the ROC curves and AUC values for the six machine learning models. The ROC curves and AUC values indicate that Logistic Regression and SVM are the top-performing models, with AUCs of 0.98 and 0.97, respectively. Random Forest and KNN also perform well with AUCs of 0.96, while XGBoost follows closely with an AUC of 0.95. The Decision Tree model, with an AUC of 0.94, is effective but less robust compared to the others. These results highlight the importance of selecting models with high AUC values for precise and reliable classification tasks.

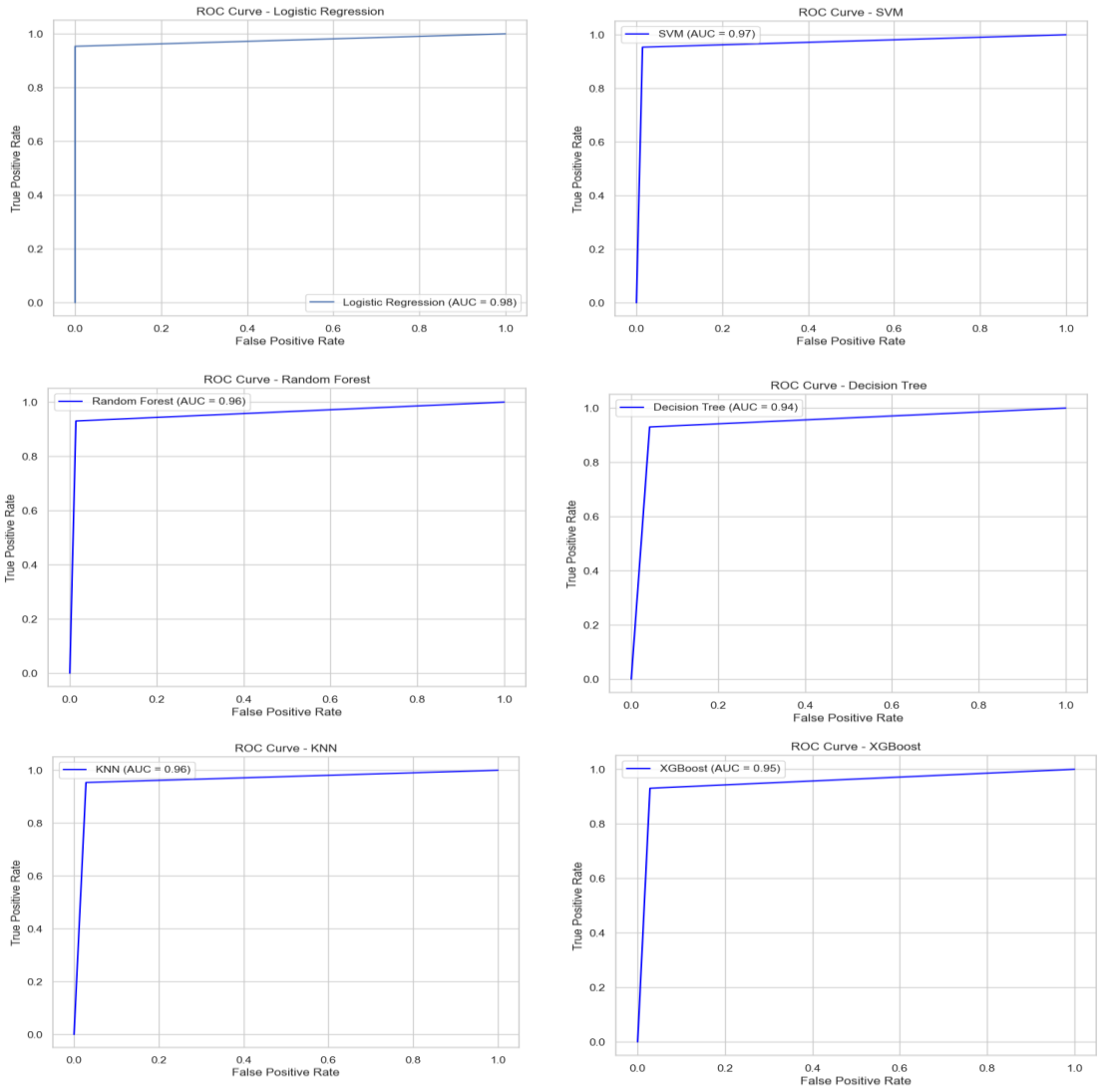


Figure 31: Evaluation of ROC Curves for Various Machine Learning Models

4.2 Result- Comparison of Performance of Models After Feature Selection

In the development of predictive models for cancer diagnosis, the precision and reliability of the predictions are paramount. To enhance these aspects, Recursive Feature Elimination (RFE) was applied as a strategic method to streamline the complexity of the models by focusing on the most impactful features.

This report detailed the outcomes of employing RFE across various machine learning models, including Logistic Regression, SVM, Random Forest, Decision Tree, KNN, and XG-Boost.

4.2.1 Summary of the Models' Performance on Post-Feature Selection Using Recursive Feature Elimination (RFE)

Figure 32 gives the summary of performance after implementing Recursive Feature Elimination (RFE) on the predictive models, here is an analysis of the metrics:

Logistic Regression stands out with a precision rate of 100%, ensuring that all its positive predictions are accurate, which is especially critical in clinical settings to avoid unnecessary interventions. It also boasts an impressive accuracy of 97% and a recall of 93%, resulting in an F1 score of 0.96, making it the top-performing model in the analysis.

Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) also show strong performances, with both models achieving an F1 score of 0.95 and 0.96 respectively. SVM has balanced precision and recall at 95%, while KNN slightly edges ahead with precision at 98% and recall at 95%.

Random Forest Classifier (RFC) and XG-Boost maintain robust metrics, with RFC showing a precision of 98% and XG-Boost matching this precision but with slightly lower recall at 93%. Both models achieve an F1 score of 0.95, highlighting their reliability.

Decision Tree (DT), while slightly trailing behind, posts decent figures with an accuracy of 92%, a precision of 90%, and a recall of 88%, leading to an F1 score of 0.89. It remains a viable option, albeit less efficient than the others.

In conclusion, the application of RFE has effectively concentrated model training on the most impactful features, enhancing their precision and overall performance. Logistic Regression emerges as the best model due to its unmatched precision and high scores across all metrics.

Figure 32 : Screenshot Comparison of the Models' Performance on Full Features Set

	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.97	1.00	0.93	0.96
SVM	0.96	0.95	0.95	0.95
RFC	0.96	0.98	0.93	0.95
DT	0.92	0.90	0.88	0.89
KNN	0.97	0.98	0.95	0.96
XG-Boost	0.96	0.98	0.93	0.95

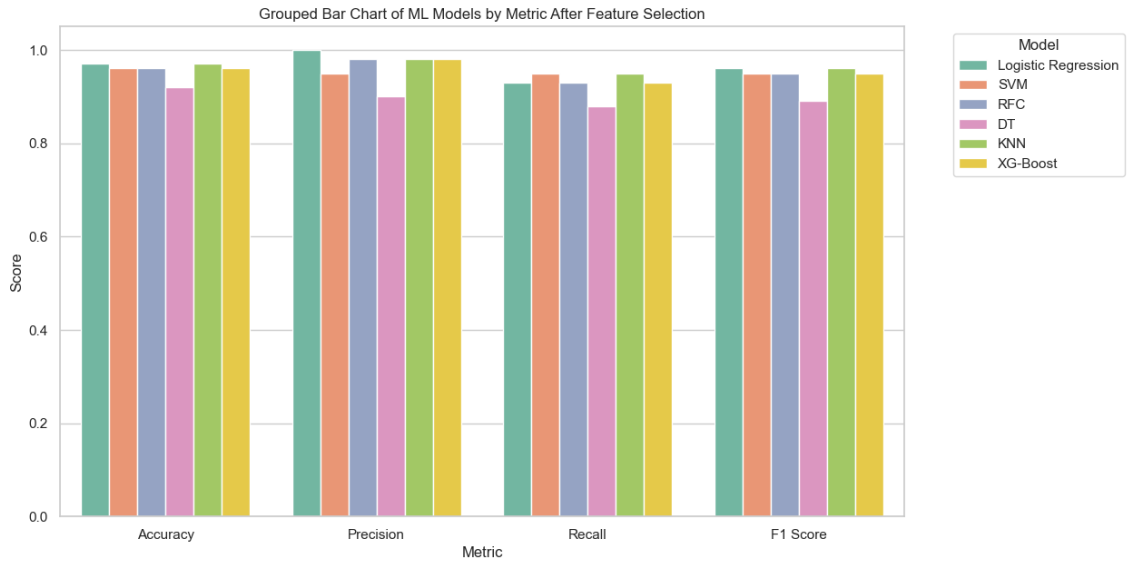


Figure 33: Bar Chart Representation of Machine Learning Model Efficacy Across Four Metrics After Feature Selection

4.2.2 Comparative Performance of Machine Learning Models Post-Feature Selection

Confusion Matrices and ROC Curves Analysis

(a)Figure 34: Confusion Matrices for Machine Learning Models

Figure 34 displays the confusion matrices for the six machine learning models. Logistic Regression achieves 71 true negatives (TN), 40 true positives (TP), 0 false positives (FP), and 3 false negatives (FN), demonstrating high accuracy with no false positives. SVM shows 69 TN, 41 TP, 2 FP, and 2 FN, indicating strong performance with slightly more false positives. Random Forest presents 69 TN, 40 TP, 2 FP, and 3 FN, reflecting robust performance but slightly lower sensitivity. Decision Tree records 67 TN, 38 TP, 4 FP, and 5 FN, indicating lower precision and sensitivity. KNN shows 70 TN, 41 TP, 1 FP, and 2 FN, achieving minimal false positives and false negatives. XG-Boost has 70 TN, 40 TP, 1 FP, and 3 FN, showing balanced performance.

(b)Figure 35: ROC Curves and AUC Values for Machine Learning Models

Figure 35 presents ROC curves and AUC values, highlighting each model's classification performance. Logistic Regression achieves an AUC of 1.00, indicating perfect classification capability. SVM exhibits an AUC of 0.99, showing excellent performance. Random Forest also achieves an AUC of 1.00, indicating robust classification. Decision Tree shows an AUC of 0.991, reflecting strong but slightly lower performance. KNN records an AUC of 0.98, indicating high accuracy, and XG-Boost has an AUC of 0.99, demonstrating strong performance.

Logistic Regression is the best performer, with perfect AUC and high accuracy, followed closely by Random Forest and SVM, both showing near-perfect AUC values and strong performance. KNN and XG-Boost also demonstrate robust results, while the Decision Tree model has comparatively lower performance. These findings highlight the importance of high sensitivity, precision, and overall accuracy in selecting reliable models for classification tasks.

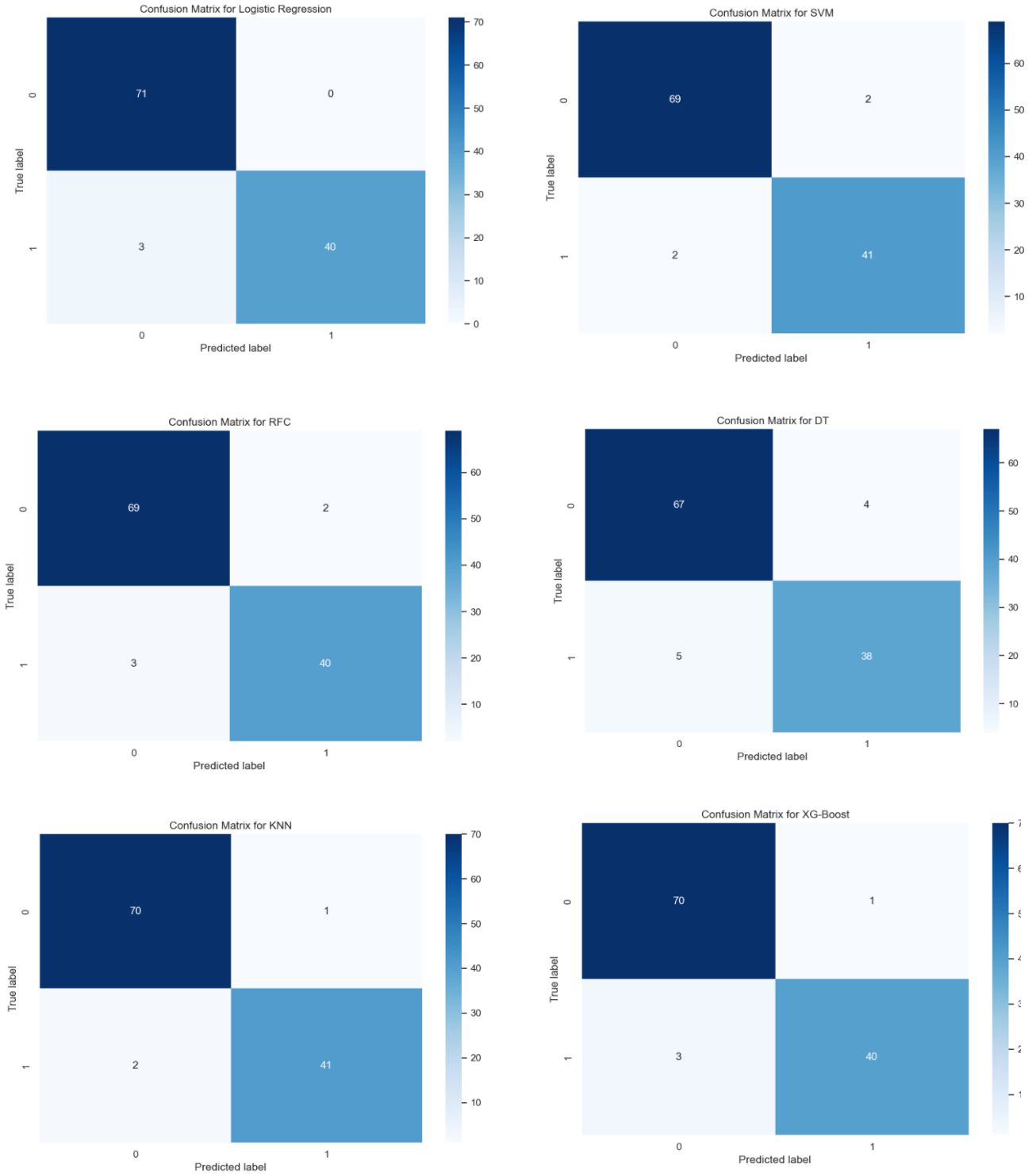


Figure 34: Confusion Matrices for Machine Learning Models

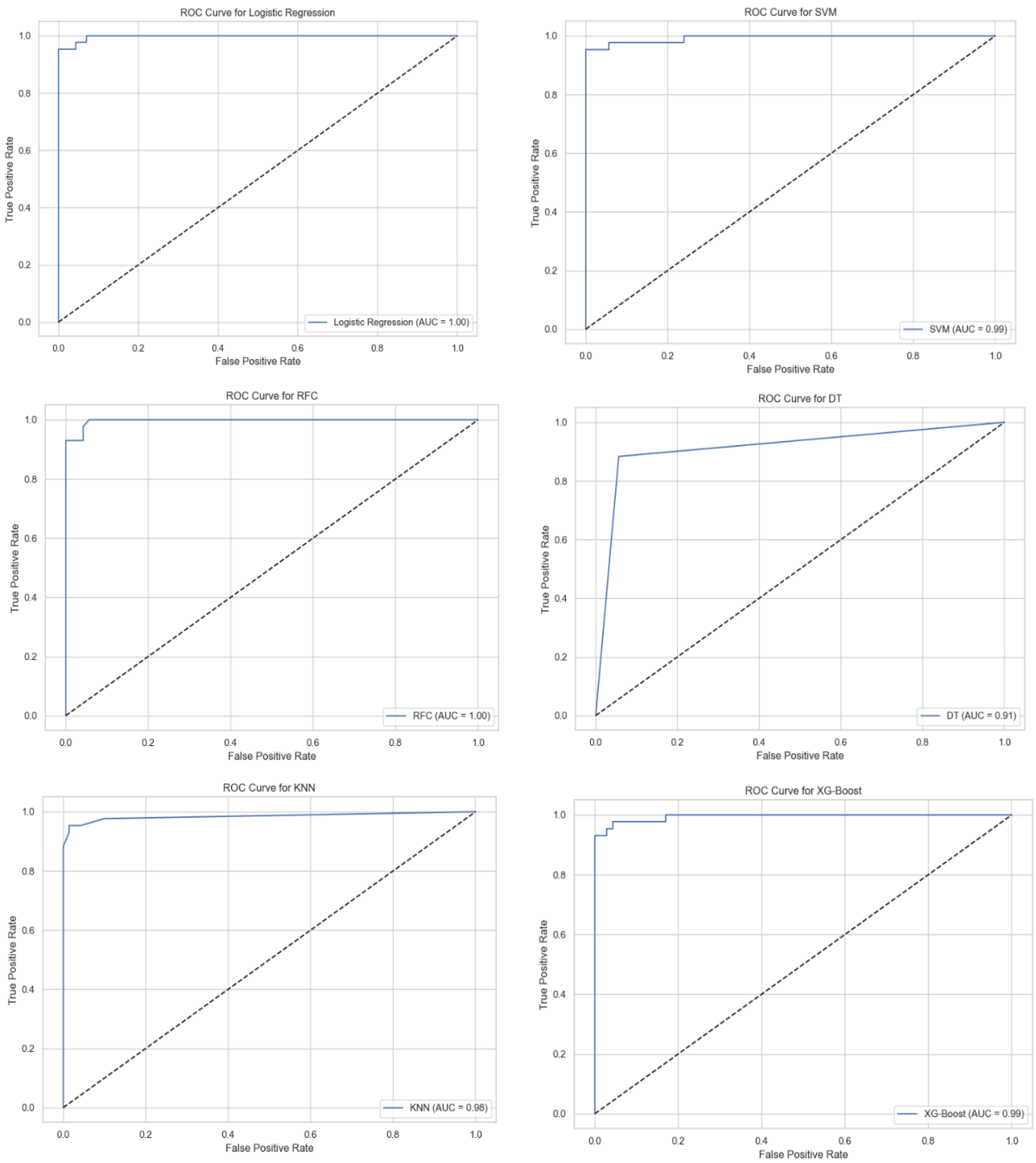


Figure 35: ROC Curves and AUC values for Machine Learning Models

4.3 Result: Cross Validation and Grid Search

Following feature selection, cross-validation and grid search were employed to optimize the hyperparameters of various machine learning models, aiming to enhance their predictive accuracies.

4.3.1 Summary of Results from Grid Search and Cross Validation

Figure 36 provides the summary of the GridSearch and Cross Validation.

Logistic Regression and RFC (Random Forest Classifier) maintained high stability in their performance with accuracies of 0.97 and 0.96, respectively, even after hyperparameter tuning and feature selection.

SVM (Support Vector Machine) showed a slight decline in accuracy from 0.97 to 0.96 after adjusting parameters post-feature selection.

DT (Decision Tree) experienced a decrease in performance, dropping from 0.94 to 0.91, indicating potential sensitivity to the reduced feature set.

KNN (K-Nearest Neighbors) improved slightly from 0.96 to 0.97 by adjusting the number of neighbors.

XG-Boost remained consistent with an accuracy of 0.96 before and after tuning.

The grid search and cross-validation processes have highlighted that while some models like KNN improved slightly, others like the Decision Tree saw a decrease in performance. Logistic Regression and RFC demonstrated excellent stability in their performance, affirming their robustness across different configurations.

	Model	Parameters	Accuracy
0	Logistic Regression	{'C': 10, 'solver': 'lbfgs'}	0.97
1	SVM	{'C': 1, 'gamma': 1, 'kernel': 'rbf'}	0.97
2	RFC	{'max_depth': 20, 'n_estimators': 50}	0.96
3	DT	{'max_depth': 10, 'min_samples_split': 10}	0.94
4	KNN	{'n_neighbors': 3}	0.96
5	XG-Boost	{'learning_rate': 0.1, 'max_depth': 3, 'n_esti...	0.96
6	Logistic Regression (Selected Features)	{'C': 10, 'solver': 'lbfgs'}	0.97
7	SVM (Selected Features)	{'C': 10, 'gamma': 1, 'kernel': 'rbf'}	0.96
8	RFC (Selected Features)	{'max_depth': 10, 'n_estimators': 50}	0.96
9	DT (Selected Features)	{'max_depth': None, 'min_samples_split': 5}	0.91
10	KNN (Selected Features)	{'n_neighbors': 7}	0.97
11	XG-Boost (Selected Features)	{'learning_rate': 0.1, 'max_depth': 3, 'n_esti...	0.96

Figure 36: Screenshot- Model Performance Comparison Before and After Feature Selection with Hyperparameter Tuning

4.4 Result : Feature Importance Analysis

The feature importance analysis for Logistic Regression, Random Forest Classifier (RFC), Decision Tree (DT), and XG-Boost reveal key insights into the features most influential in predicting cancer outcomes as shown in Figure 37.

Logistic Regression and XG-Boost emphasize the importance of concave points_worst and concave points_mean, highlighting their strong association with the target variable.

RFC and DT focus significantly on geometric features of the tumor such as perimeter_worst and texture_worst, along with concave points_mean, illustrating their critical roles in the models' predictive processes.

This analysis highlights the consistency in critical features across models, particularly those detailing the tumor's concavity, which aids in enhancing the models' predictive reliability and efficiency for clinical applications. This streamlined insight into feature importance is crucial for refining future model development and ensuring effective real-world deployment.

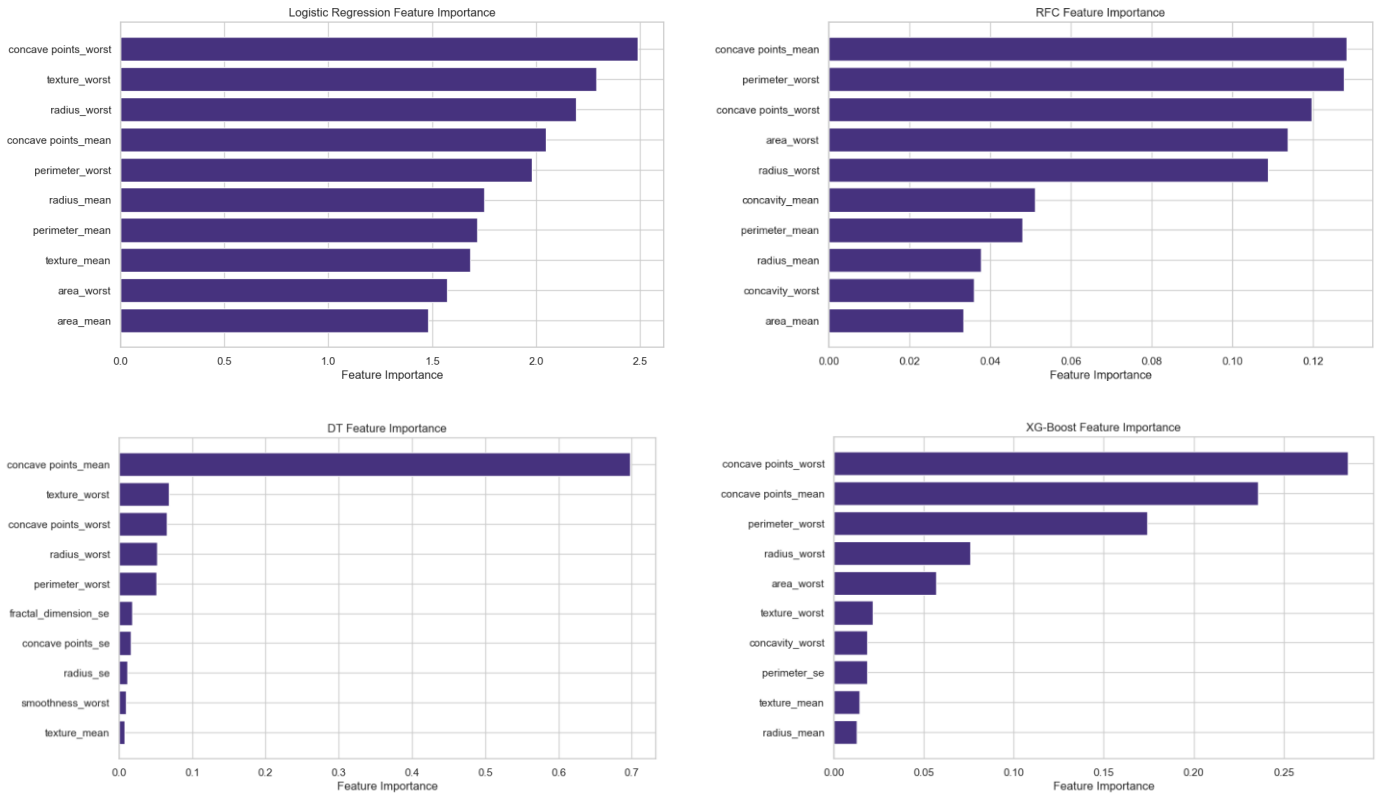


Figure** 37**: Feature Importance Analysis Across Machine Learning Models

4.5 Result: Key Feature Significance Across Machine Learning Models in Breast Cancer Diagnosis

4.5.1 Variable Significance Analysis

This analysis highlights the significance of certain features that consistently emerge as critical across various machine learning models used in cancer diagnosis. The visualization in Figure 38 employs frequency counts to illustrate the prominence of each feature's importance in these models.

Key Predictors: The features concave points_worst, radius_worst, perimeter_worst and concave points_mean are key predictors across various models, suggesting their strong relevance in the context being analyzed.

Implication: When selecting features for a model, prioritizing those that appear frequently across multiple models can enhance predictive performance and model robustness.

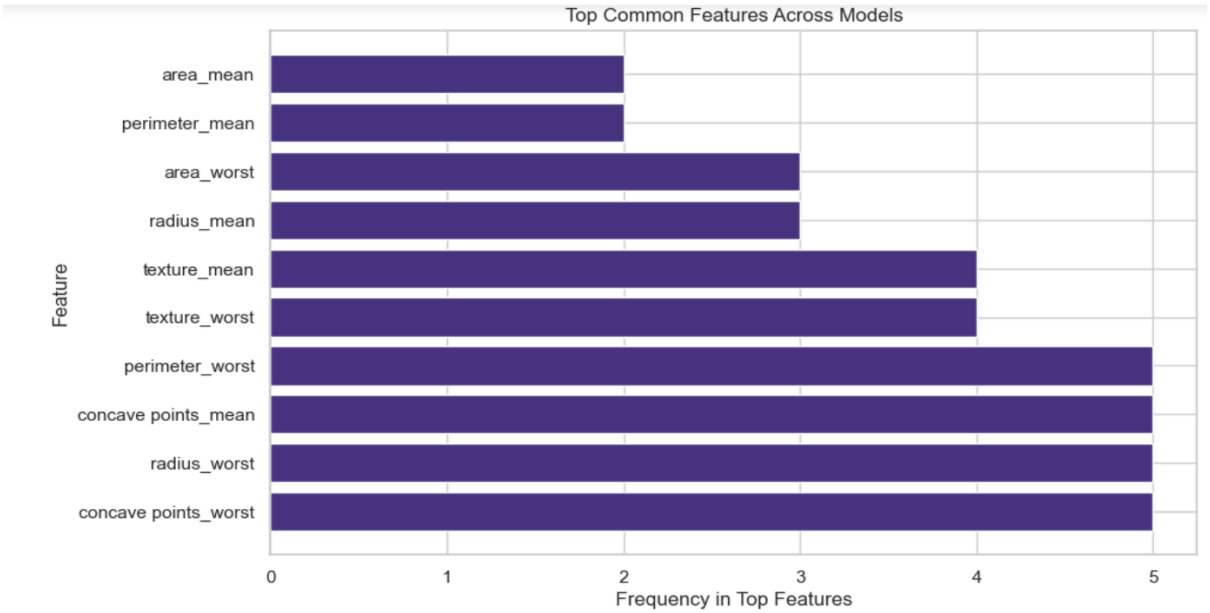


Figure 38: Frequency Distribution of Feature Importance Across Multiple Machine Learning Models

4.6 Result: Local Interpretable Model-agnostic Explanations(LIME) Analysis for the Models

4.6.1 Result: LIME Analysis Overview for Logistic Regression

Figures 39 and 40 demonstrate the Logistic Regression model's analysis for predicting breast cancer as malignant.

Detailed Components of the Visual

Prediction Probabilities:

Malignant (1): The model predicts with a high confidence of 97% that the cancer is malignant, indicating a strong likelihood based on the analyzed features.

Benign (0): In contrast, there is only a 3% probability that the cancer is benign, underscoring the model's certainty towards a malignant classification.

Key Features Influencing the Prediction:

Concave Points (Worst) with a value of 0.61 and the highest impact weight of 0.20, is critical in the model's prediction. High values typically indicate more pronounced irregularities in the tumor's cell structure, which are strongly associated with malignant tumors. This feature's significant weight and high value strongly drive the model's decision towards a malignant

diagnosis. Radius (Worst) with a value of 0.60 and an impact weight of 0.14, this feature indicates the extent of the tumor's radius. A larger radius is commonly correlated with more aggressive cancer forms, influencing the model to lean towards a malignancy prediction.

Perimeter (Worst) with value of 0.58 and a corresponding weight of 0.13 also plays a substantial role. It reflects the outer boundary measurement of the tumor, where larger and more irregular perimeters are indicative of malignancy.

This LIME visualization illustrates how the Logistic Regression model determines its predictions by detailing the impact of each feature. For example, features like 'concave points_worst' and 'radius_worst' are crucial, as their elevated values and substantial weights correlate with common traits of malignant tumors, such as irregular contours and greater dimensions. This comprehensive analysis of feature contributions helps ensure that the model's outputs can be interpreted and corroborated within a medical setting.

Figures 39 and 40 collectively demonstrate the interpretability of the Logistic Regression model using LIME. Figure 39 highlights the prediction probabilities and key contributing features, while Figure 40 visually represents the importance of these features in classifying an instance as class 1.

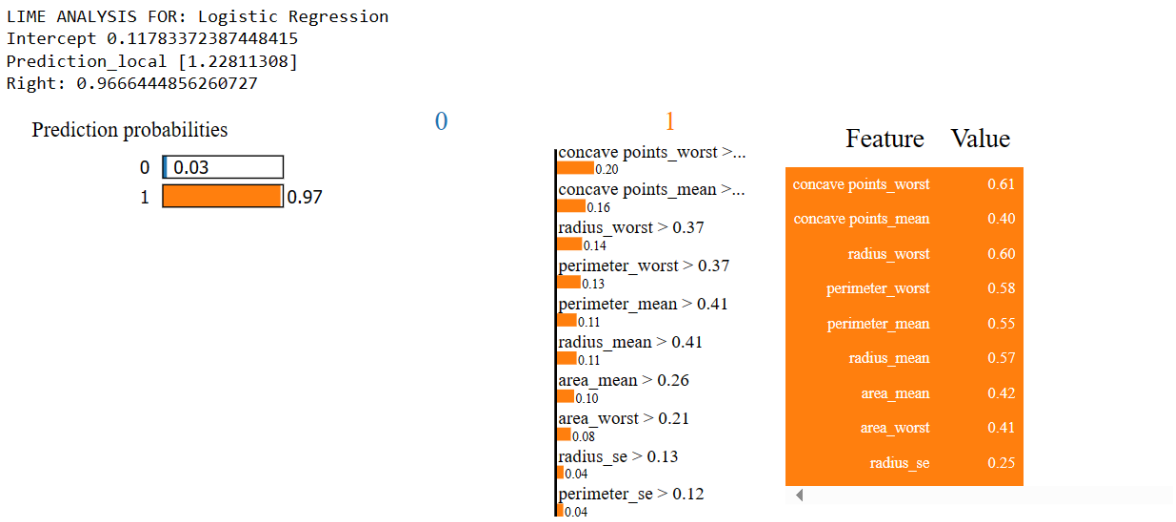


Figure 39 : Local Interpretation of Predictive Factors for Malignancy Using LIME in Logistic Regression

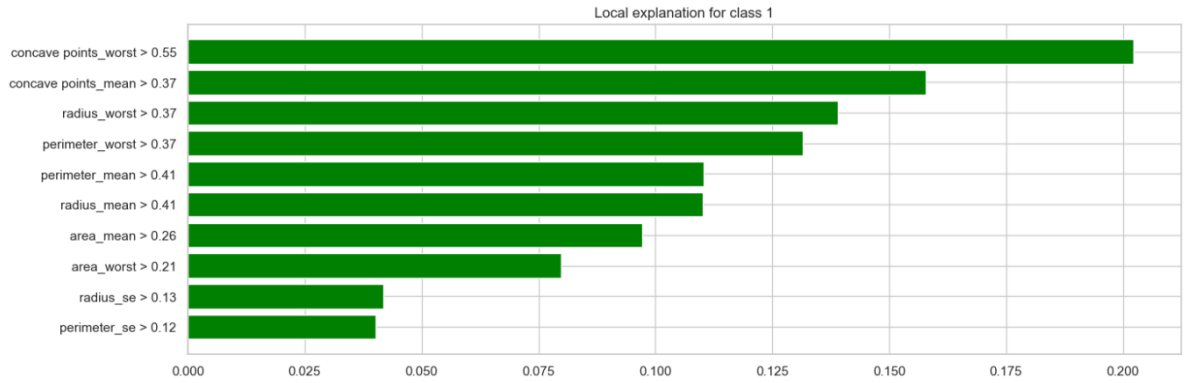


Figure 40: Bar Chart showing Key Feature Contributions to Malignancy Prediction in Logistic Regression Model

4.6.2 Result: LIME Analysis Overview for SVM

The LIME analysis for the SVM model reveals significant feature influences on its decision to predict malignancy:

Key Features: Concave Points (Worst) is the most impactful with a weight of 0.23, indicating its major role in the model's decision-making. Radius (Worst) and Concave Points (Mean) follow closely, with weights of 0.22 and 0.19, respectively, emphasizing their importance in diagnosing malignancy.

Supporting Features: Perimeter (Worst), Area (Worst), and Radius (Mean) also contribute significantly, reflecting the model's sensitivity to tumor dimensions and shape.

The accompanying bar graph visually confirms the dominance of these features, with Concave Points (Worst) leading in influence. This concise analysis helps clarify the SVM's reliance on specific tumor characteristics to predict outcomes, enhancing the model's interpretability and utility in clinical settings.

Figures 41 and 42 collectively clarify the SVM model's reliance on specific tumor characteristics to predict outcomes.

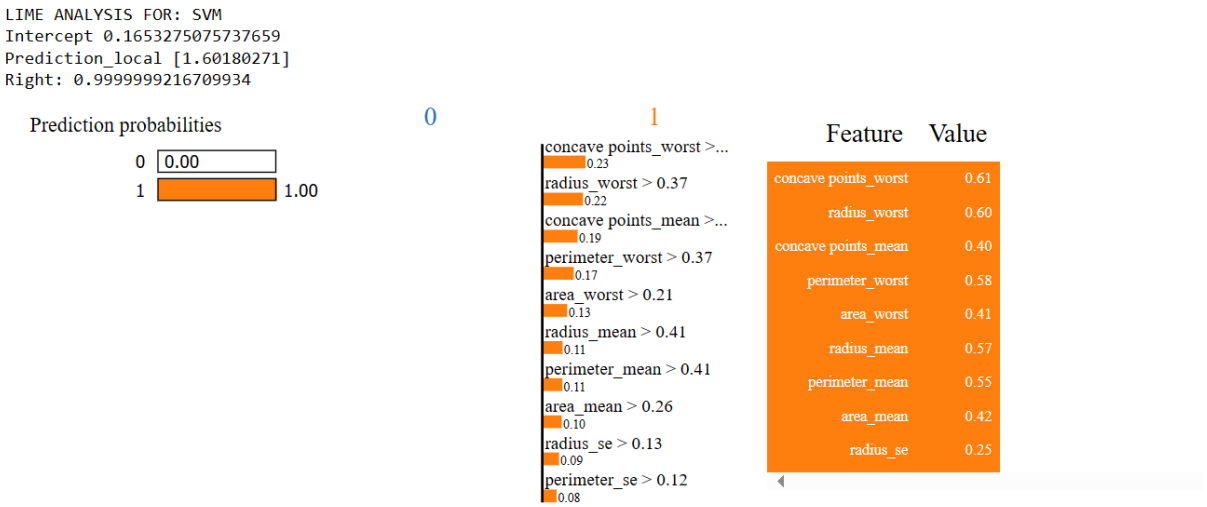


Figure 41: Local Interpretation of Predictive Factors for Malignancy Using SVM

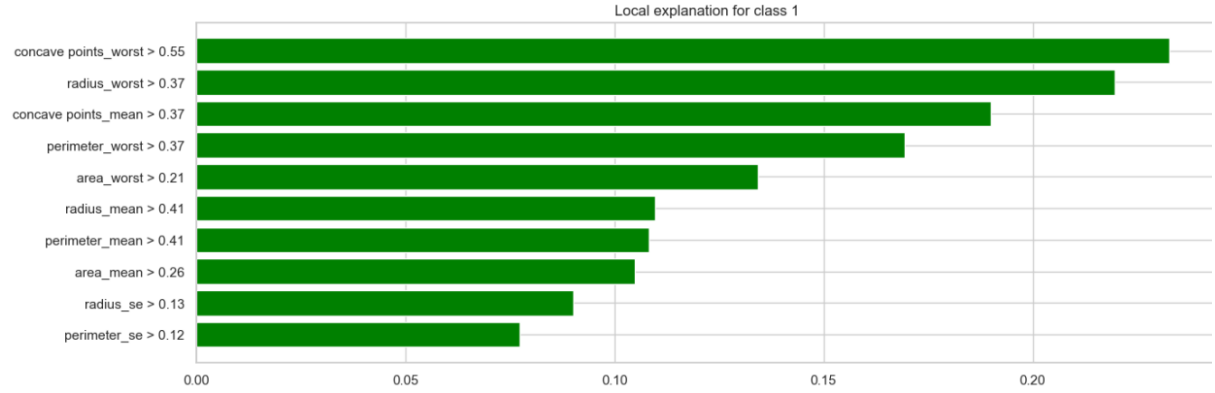


Figure 42: Bar Chart showing Key Feature Contributions to Malignancy Prediction in SVM

4.6.3 Result: LIME Analysis Overview for Random Forest

The LIME analysis for the Random Forest Classifier (RFC) illustrates how different features contribute to a prediction of malignancy with complete certainty (probability of 1.00):

Key Influencers: The model is heavily influenced by Radius Worst (0.13) and Area Worst (0.12), which suggests the size of the tumor is critical in the prediction. Concave Points Worst also plays a significant role with a weight of 0.10, indicating the severity of concave deformations as a crucial predictor.

Supporting Features: Perimeter Worst and Concave Points Mean contribute weights of 0.09 and 0.08, respectively, emphasizing the model's sensitivity to tumor shape and internal characteristics.

The visualization aligns these findings, marking the size and physical attributes of the tumor as primary diagnostic indicators in the RFC model. This concise interpretation offers a clear understanding of feature impacts, enhancing trust in the model's use for clinical diagnostics.

Figures 43 and 44 collectively clarify the Random Forest Classifier's reliance on specific tumor characteristics to predict outcomes.

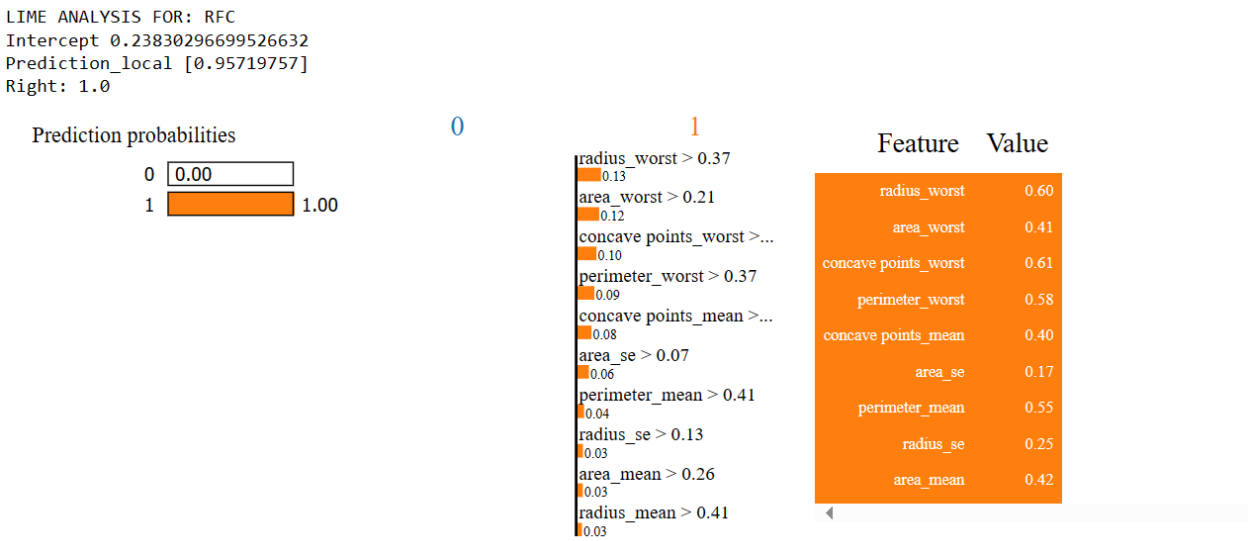


Figure 43 : Local Interpretation of Predictive Factors for Malignancy Using RFC

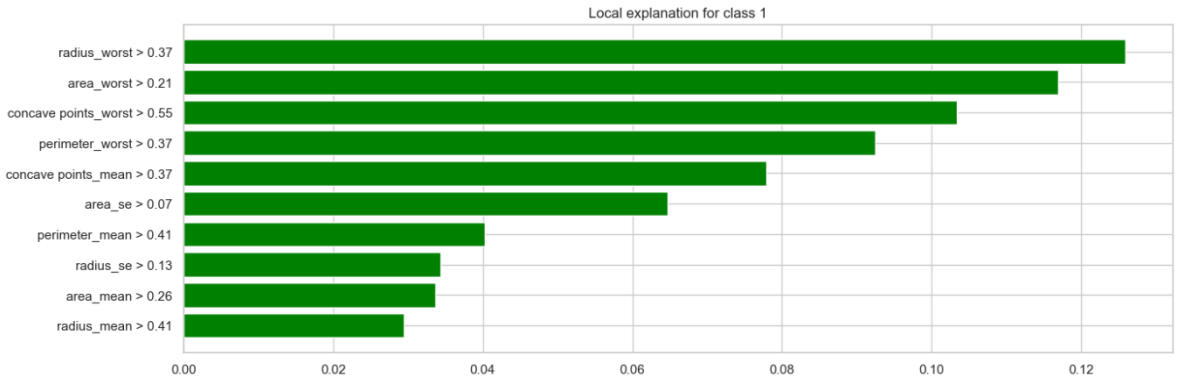


Figure 44: Bar Chart showing Key Feature Contributions to Malignancy Prediction in RFC

4.6.4 Result: LIME Analysis Overview for Decision Tree

Figure 45 shows the LIME analysis for the Decision Tree (DT) model, highlighting key features influencing its malignancy predictions. The most impactful features are Concave Points (Mean)

with a weight of 0.40, followed by Concave Points (Worst) at 0.17, and Perimeter (Worst) and Radius (Worst) both at 0.15. Supporting features include Concave Points (SE), Radius (SE), and Texture (Worst).

Figure 46 provides a bar graph confirming the dominance of these features, with Concave Points (Mean) and Concave Points (Worst) leading, followed by Perimeter (Worst) and Radius (Worst). It also illustrates negative influences like Concave Points (SE) between 0.21 and 0.28.

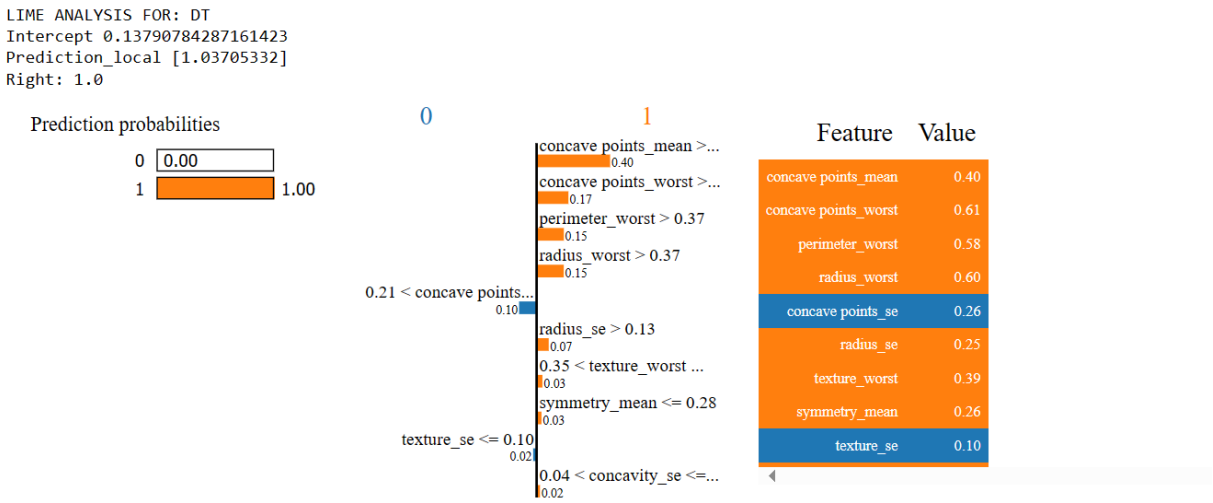


Figure 45 : Local Interpretation of Predictive Factors for Malignancy Using DT

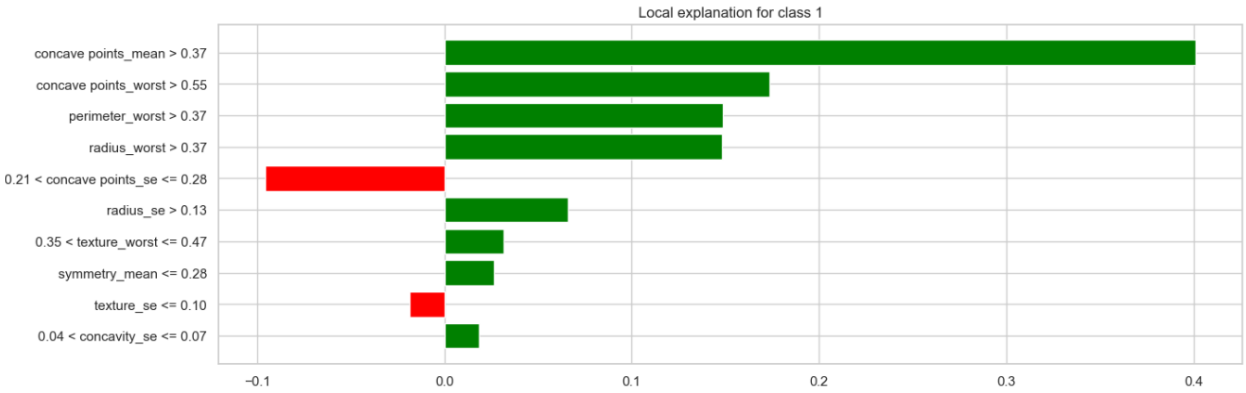


Figure 46: Bar Chart showing Key Feature Contributions to Malignancy Prediction in DT

4.6.5 Result: LIME Analysis Overview for KNN

The LIME analysis for the KNN model indicates that the classification decision is heavily influenced by a set of key features. The feature 'concave points_worst' with a weight of 0.22 is the

most significant predictor, followed closely by 'radius_worst' and 'perimeter_worst' both having an impact value of 0.19 and 0.17, respectively. These features collectively drive the high prediction confidence of class 1 with a probability of 1.00. The visualization also highlights the positive contributions of 'concave points_mean' and 'area_worst', emphasizing their relevance in the model's decision-making process. Figures 47 and 48 demonstrate the KNN model's reliance on tumor characteristics like Concave Points and Radius, enhancing its interpretability.

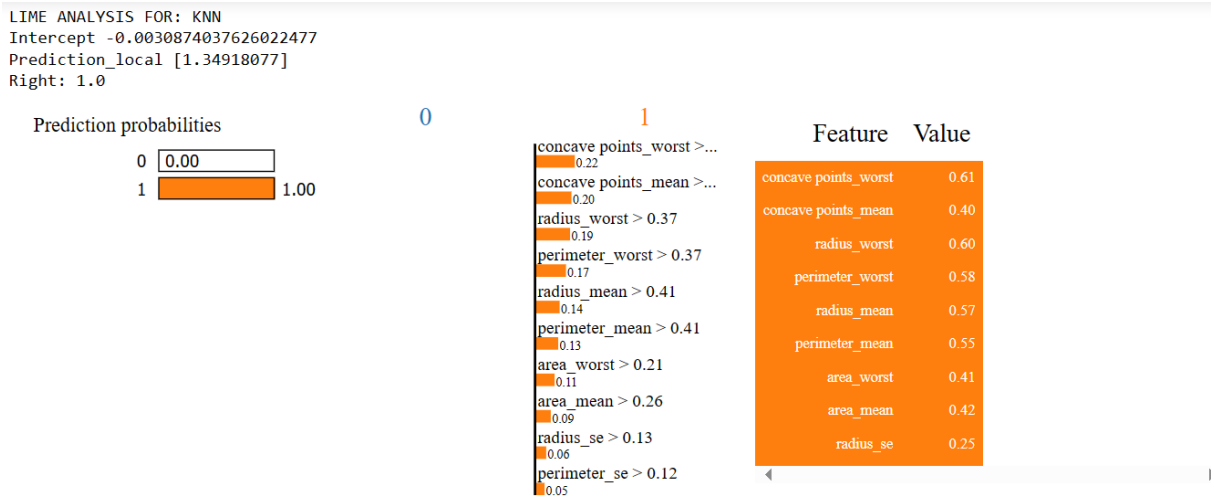


Figure 47 : Local Interpretation of Predictive Factors for Malignancy Using KNN

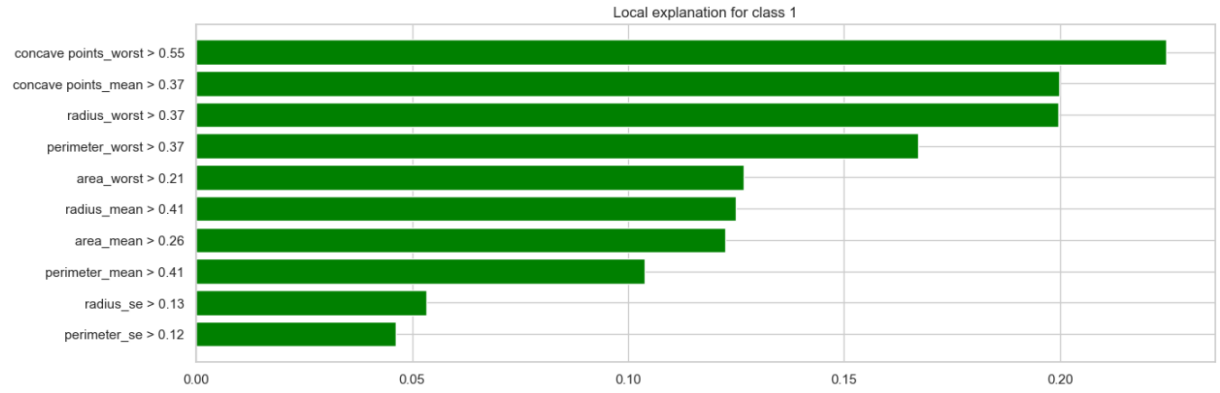


Figure 48: Bar Chart showing Key Feature Contributions to Malignancy Prediction in KNN

4.6.6 Result: LIME Analysis Overview for XGBoost

Figure 49 shows the LIME analysis for the XG-Boost model, highlighting key features influencing its malignancy predictions. The most impactful features are Area (Worst) with a weight of 0.26,

Concave Points (Worst) at 0.25, and Concave Points (Mean) at 0.20. Supporting features include Area (SE), Concavity (Worst), and Texture (Worst).

Figure 50 provides a bar graph confirming the dominance of these features, with Area (Worst) and Concave Points (Worst) leading, followed by Concave Points (Mean) and Area (SE). Negative influences, such as Symmetry (Worst) between 0.18 and 0.25, are also shown.

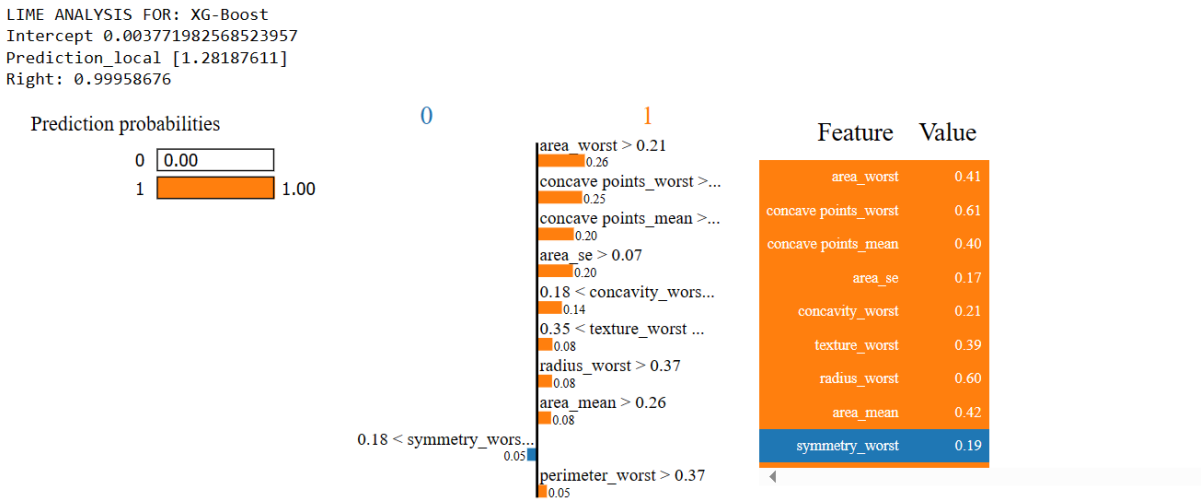


Figure 49 : Local Interpretation of Predictive Factors for Malignancy Using XGBoost

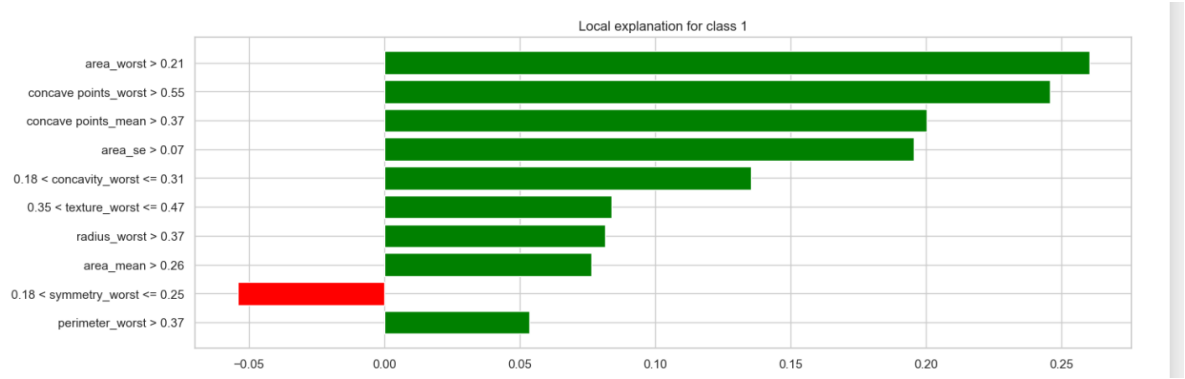


Figure 50: Bar Chart showing Key Feature Contributions to Malignancy Prediction in XGBoost

4.7 Discussions

This thesis evaluates various machine learning models for their effectiveness in predicting breast cancer using the Wisconsin Breast Cancer Dataset. The models examined include Logistic

Regression, SVM, Random Forest, K-Nearest Neighbors, Decision Tree, and XGBoost, with an emphasis on their performance before and after feature selection. The discussion incorporates extensive analysis of model accuracy, parameter settings, feature importance, and individual prediction explanations through LIME analysis, providing a thorough understanding of each model's capabilities. A key aspect of the investigation was not only to improve the accuracy of predictions but also to enhance the interpretability of the models, ensuring that they can be understood and trusted by medical professionals.

4.7.1 Baseline Performance:

Initially, all models were evaluated using the full set of features to establish a performance baseline. Logistic Regression emerged as the leading model with an accuracy of 0.98, precision of 1.00, recall of 0.95, and an F1 score of 0.98. SVM and other models also demonstrated high performance but were slightly outperformed by Logistic Regression.

4.7.2 Impact of Feature Selection:

Feature selection was implemented to enhance model interpretability and reduce computational complexity. Notably, Logistic Regression maintained high precision (1.00) and achieved an accuracy of 0.97, slightly lower than the baseline but still highly efficient. The consistency in performance emphasizes the robustness of these models even when the number of input features is reduced. The Decision Tree Classifier showed a decrease in performance metrics, indicating sensitivity to reduced feature sets.

4.7.3 Comparative Analysis of Performance Before and After Feature Selection

The comparative analysis before and after feature selection highlights that while most models maintained high performance metrics, some like the Decision Tree experienced performance drops. This indicates a trade-off between model simplicity and accuracy, which is crucial for applications requiring rapid and reliable diagnostics.

4.7.4 Variable Significance Determination- Feature Importance Analysis

Significant predictors such as "concave points_worst", "radius_worst", "concave points_mean" and "perimeter_worst" were consistently identified across models, indicating their critical role in predicting breast cancer.

Decision Tree and XG-Boost highlighted additional features like 'area_worst' and 'concavity_worst', which are indicative of more complex interactions within the models that focus on specific tumor characteristics.

4.7.5 LIME Analysis Insights

Logistic Regression and SVM focus on features like 'concave points_worst' and 'radius_worst', important for identifying malignancy due to their association with aggressive tumor characteristics.

Random Forest and XG-Boost emphasize 'area_worst' and 'concavity_worst', which relate to structural disruptions in malignant tumors.

Decision Tree leverages a diverse set of features from shape to texture, indicating its rule-based decision-making approach.

KNN uses 'radius_mean' and 'perimeter_mean', reflecting its method of averaging outcomes from nearest neighbors to assess tumor size and boundary.

LIME analysis provides a clear view into the decision-making process of each model, with a common emphasis on features indicative of malignancy while also highlighting unique attributes used by different models. This not only enhances trust in model predictions but also helps in understanding the biological underpinnings of breast cancer indicators.

4.7.6 Optimum Classifier Identification

Considering both accuracy and interpretability, Logistic Regression emerged as the optimum classifier due to its high accuracy, consistency across feature sets, and clear interpretability provided by LIME analysis. It effectively balances the simplicity of the model with the complexity needed to capture essential diagnostic features.

4.8 Limitation of the Study

Generalization to Other Cancer Types: While this study has provided valuable insights into breast cancer prediction using machine learning models, its applicability to other types of cancer has not been examined. The effectiveness and reliability of these models for diagnosing and

prognosing different forms of cancer, which may have distinct biological characteristics and diagnostic criteria, remain untested. This limitation suggests a need for further research to validate and possibly adapt these models for broader oncological applications.

4.9 Key Findings and Response to Research questions

4.9.1 Research Questions and Answers

RQ1. What machine learning model is most effective for predicting breast cancer?

Answer: Logistic Regression emerged as the optimum classifier in this study. The high performance of the Logistic Regression model, in terms of accuracy , precision and recall and interpretability emphasized its reliability for clinical applications.

RQ2.How does feature selection impact the performance of predictive models in breast cancer prognosis?

Answer: Feature selection, particularly through RFE, played a pivotal role in enhancing model accuracy and computational efficiency. By reducing the feature space, the models focused on the most informative attributes and less overfitting . This also speeds up training, simplifies model complexity, and aids in clearer interpretability and more informed clinical decision-making.

RQ3. Which features are most critical for accurate breast cancer prediction?

Answer: The study identified "concave points_worst", "radius_worst", "concave points_mean" and "perimeter_worst" as key features. Their significance across multiple models validated their role in enhancing predictive accuracy and informed targeted feature selection for future model improvements .

RQ4. What role does model interpretability play in the clinical applicability of predictive models?

Answer: LIME analysis provided detailed insights into individual predictions for each model, revealing the specific features that influence decisions. Interpretability, achieved through techniques like LIME (Local Interpretable Model-agnostic Explanations), was crucial for clinical acceptance. By providing insights into the decision-making process of each model, clinicians can better understand and trust the predictions, which is essential for practical deployment in medical diagnostics .

4.9.2 Implications of Study for Clinical Application

The research provides compelling evidence of the potential for machine learning models to enhance breast cancer diagnosis. The findings suggest that incorporating advanced machine learning techniques can significantly improve diagnostic accuracy and efficiency. For clinical applications, models that combine high predictive accuracy with interpretability, such as Logistic Regression and Random Forest, are particularly valuable as they aid in early detection and treatment planning while ensuring that the decision-making process is transparent and understandable.

4.9.3 Comparison with Other Studies

While this study aligns with several studies in terms of predictive accuracy, it particularly excels in model interpretability, an aspect often overlooked in similar research. The use of LIME (Local Interpretable Model-agnostic Explanations) to clarify the decision-making process of each model sets the approach of this study apart, aligning with the recent push towards XAI(Explainable Artificial Intelligence) in healthcare.

This discussion synthesizes the research findings, answering the core research questions and highlighting the implications and future directions for using enhanced machine learning in cancer prediction and prognosis.

5. Chapter Five: Conclusion, Recommendation and Future Works

5.1 Conclusion

This study has conducted a comprehensive investigation into the application of various machine learning (ML) techniques for enhancing breast cancer prediction and prognosis, using the Wisconsin Breast Cancer Dataset (WBCD). The primary aim was to identify an optimal feature set and the most effective ML classifier that balances high predictive accuracy with interpretability, which is essential for clinical use.

Among the evaluated models, Logistic Regression emerged as the most effective classifier. It achieved the highest precision and interpretability, maintaining robust performance metrics across both full and reduced feature sets. Logistic Regression's simplicity and clarity in decision-making processes make it particularly suitable for clinical applications where transparency is crucial.

The implementation of Recursive Feature Elimination (RFE) significantly enhanced model performance by concentrating on the most informative features. This process improved computational efficiency, reduced model complexity, and ensured that the predictions were more transparent and actionable. The RFE technique proved instrumental in refining the models to focus on key predictive attributes, enhancing their overall efficacy.

The study consistently identified "concave points_worst," "radius_worst," "concave points_mean," and "perimeter_worst" as the most critical features for accurate breast cancer prediction. These features were pivotal across multiple models, validating their importance in distinguishing between malignant and benign cases. The prominence of these features underscores their role in enhancing the predictive accuracy and reliability of the ML models.

The use of Local Interpretable Model-agnostic Explanations (LIME) provided deep insights into the decision-making processes of each model. LIME's ability to explain individual predictions made the ML models more transparent, fostering trust among medical practitioners. This interpretability is vital for the integration of ML models into clinical practice, ensuring that healthcare providers can understand and rely on the model outputs.

The findings of this study highlight the transformative potential of ML in breast cancer diagnostics. The ability of ML models, particularly Logistic Regression, to deliver precise, reliable, and interpretable predictions can significantly improve diagnostic accuracy and efficiency. These

models can aid in early detection and better treatment planning, ultimately enhancing patient outcomes. The high interpretability achieved through LIME ensures that the predictions are not only accurate but also understandable to clinicians, facilitating informed decision-making.

This study not only aligns with existing research in terms of predictive accuracy but also stands out for its emphasis on model interpretability, an often-neglected aspect. The application of LIME (Local Interpretable Model-agnostic Explanations) to elucidate the decision-making process of each model distinguishes this study, aligning it with the recent movement towards Explainable Artificial Intelligence (XAI) in healthcare.

5.2 Recommendation

To enhance the transparency and trust in predictive models used in clinical oncology, it is recommended that LIME (Local Interpretable Model-agnostic Explanations) be integrated into regular model evaluations. This integration should be complemented by exploring a variety of explainability tools to deepen insights and refine approaches within the field. By adopting these tools more broadly, the predictive models will become more interpretable and trustworthy to clinicians, ultimately facilitating their adoption in clinical settings.

5.3 Future Works

The following directions for future research are critical for advancing the efficacy and relevance of machine learning models in cancer prognosis and diagnosis:

5.3.1 Advancement in Model Interpretability: Future studies should focus on developing hybrid explainability frameworks that integrate multiple methods such as SHAP (SHapley Additive exPlanations), and counterfactual explanations. These frameworks are intended to provide a more comprehensive understanding of model decisions, thereby enhancing their interpretability and usability in clinical applications.

5.3.2 Expand Model Validation: To ensure the generalizability and reliability of the developed machine learning models, it is crucial to validate them across a wide range of cancer types, stages, and molecular subtypes using diverse datasets. Conducting multicentric studies with data from various geographical and demographic backgrounds is essential to assess the models' robustness

and applicability, accommodating variations due to ethnic, cultural, and environmental factors. This comprehensive validation is necessary for reliable deployment in diverse clinical settings worldwide.

By addressing these recommendations and pursuing the outlined future research directions, the gap between technical model development and clinical application can be bridged effectively. This approach will promote the wider adoption of AI in healthcare, particularly in enhancing the diagnosis and treatment planning in oncology.

6. References

1. Akinnuwesi, B.A., Olayanju, K.A., Aribisala, B.S., Fashoto, S.G., Mbunge, E., Okpeku, M. and Owate, P., 2023. Application of support vector machine algorithm for early differential diagnosis of prostate cancer. Data Science and Management, 6(1), pp.1-12.
2. Anisha, P.R. et al., 2021. 'Early diagnosis of breast cancer prediction using random forest classifier', IOP Conference Series: Materials Science and Engineering, 1116(1), p. 012187. doi:10.1088/1757-899x/1116/1/012187.
3. Battineni, G., Chintalapudi, N. and Amenta, F., 2020. Performance analysis of different machine learning algorithms in breast cancer predictions. EAI Endorsed Transactions on Pervasive Health and Technology, 6(23).
4. Bhandari, A., 2024 Guide to AUC ROC curve in machine learning : What is specificity?, Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/ (Accessed: 21 May 2024).
5. Binsaif, N., 2022. Application of machine learning models to the detection of breast cancer. Mobile Information Systems, 2022.
6. Boeri, C. et al., 2020. 'Machine learning techniques in breast cancer prognosis prediction: A primary evaluation', Cancer Medicine, 9(9), pp. 3234–3243. doi:10.1002/cam4.2811.
7. Brownlee, J. (2020) Recursive feature elimination (RFE) for feature selection in Python, MachineLearningMastery.com. Available at: https://machinelearningmastery.com/rfe-feature-selection-in-python/ (Accessed: 21 May 2024).
8. Chudasama, Y., Purohit, D., Rohde, P.D., Gercke, J. and Vidal, M.E., 2023. InterpretME: A tool for interpretations of machine learning models over knowledge graphs. Semantic Web, (Preprint), pp.1-21.
9. Deepika, S., Ramanathan, K. and Devi, N., 2021 'Prediction of breast cancer using SVM algorithm', International Journal of Applied Engineering Research, 16(4), pp. 316-320. Available at: http://www.ripublication.com (Accessed: [05/05/2024]).
10. DeSantis, C.E., Ma, J., Gaudet, M.M., Newman, L.A., Miller, K.D., Goding Sauer, A., Jemal, A. and Siegel, R.L., 2019. Breast cancer statistics, 2019. CA: a cancer journal for clinicians, 69(6), pp.438-451.
11. Dhal, P. and Azad, C., 2022. A comprehensive survey on feature selection in the various fields of machine learning. Applied Intelligence, 52(4), pp.4543-4581.

12. Entezari, A., Aslani, A., Zahedi, R. and Noorollahi, Y., 2023. Artificial intelligence and machine learning in energy systems: A bibliographic perspective. Energy Strategy Reviews, 45, p.101017.
13. Garreau, D. and Luxburg, U., 2020, June. Explaining the explainer: A first theoretical analysis of LIME. In International conference on artificial intelligence and statistics (pp. 1287-1296). PMLR.
14. Gaur, K. and Jagtap, M.M., 2022. Role of artificial intelligence and machine learning in prediction, diagnosis, and prognosis of cancer. Cureus, 14(11).
15. Hooshmand, A., 2021. Accurate diagnosis of prostate cancer using logistic regression, Open Medicine, 16, pp. 459-463. Available at: https://doi.org/10.1515/med-2021-0238 (Accessed: [05/05/2024]).
16. Houghton, S.C. and Hankinson, S.E., 2021. Cancer progress and priorities: breast cancer. Cancer epidemiology, biomarkers & prevention, 30(5), pp.822-844.
17. Jabbar, M.A., 2021. Breast cancer data classification using ensemble machine learning. Engineering & Applied Science Research, 48(1).
18. Jaiswal, V., Suman, P. and Bisen, D. (2023) 'An improved ensembling techniques for prediction of breast cancer tissues', Multimedia Tools and Applications [Preprint]. doi:10.1007/s11042-023-16949-8.
19. Jansen, T., Geleijnse, G., Van Maaren, M., Hendriks, M.P., Ten Teije, A. and Moncada-Torres, A., 2020. Machine learning explainability in breast cancer survival. In Digital Personalized Health and Medicine (pp. 307-311). IOS Press.
20. K. Maliha, R. R. Ema, S. K. Ghosh, H. Ahmed, M. R. J. Mollick and T. Islam, "Cancer Disease Prediction Using Naive Bayes,K-Nearest Neighbor and J48 algorithm," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019, pp. 1-7, doi: 10.1109/ICCCNT45670.2019.8944686.
21. Kourou, K., Exarchos, K.P., Papaloukas, C., Sakaloglou, P., Exarchos, T. and Fotiadis, D.I., 2021. Applied machine learning in cancer research: A systematic review for patient diagnosis, classification and prognosis. Computational and Structural Biotechnology Journal, 19, pp.5546-5555.

22. Krstinić, D., Braović, M., Šerić, L. and Božić-Štulić, D., 2020. Multi-label classifier performance evaluation with confusion matrix. Computer Science & Information Technology, 1, pp.1-14.
23. Kumar, B.S., Daniya, T. and Ajayan, J., 2020. Breast cancer prediction using machine learning algorithms. International Journal of Advanced Science and Technology, 29(3), pp.7819-7828.
24. Kumar, S. and Gota, V., 2023. Logistic regression in cancer research: A narrative review of the concept, analysis, and interpretation. Cancer Research, Statistics, and Treatment, 6(4), pp.573-578.
25. Ladbury, C., Zarinshenas, R., Semwal, H., Tam, A., Vaidehi, N., Rodin, A.S., Liu, A., Glaser, S., Salgia, R. and Amini, A., 2022. Utilization of model-agnostic explainable artificial intelligence frameworks in oncology: a narrative review. Translational Cancer Research, 11(10), p.3853.
26. Li, Y., Sun, H., Wang, C. & Xu, Y., 2019. Breast cancer classification based on logistic regression using clinical features. Journal of X-Ray Science and Technology, 27(6), 949-958.
27. liliyi, N.A., Rossli, S.A., Ahmad, N. and Noor, N.M., 2021, March. Comparison on Some Machine Learning Techniques in Breast Cancer Classification. In 2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES) (pp. 499-504). IEEE.
28. Momenyan, S. et al., 2018 'Survival prediction of patients with breast cancer: Comparisons of decision tree and logistic regression analysis', International Journal of Cancer Management, 11(7). doi:10.5812/ijcm.9176.
29. Petinrin, O.O., Saeed, F., Toseef, M., Liu, Z., Basurra, S., Muyide, I.O., Li, X., Lin, Q. and Wong, K.C., 2023. Machine learning in metastatic cancer research: Potentials, possibilities, and prospects. Computational and Structural Biotechnology Journal.
30. Pudjihartono, N., Fadason, T., Kempa-Liehr, A.W. and O'Sullivan, J.M., 2022. A review of feature selection methods for machine learning-based disease risk prediction. Frontiers in Bioinformatics, 2, p.927312.
31. Rane, N., Sunny, J., Kanade, R. and Devi, S., 2020. Breast cancer classification and prediction using machine learning. International Journal of Engineering Research and Technology, 9(2), pp.576-580.

32. Rodríguez-Pérez, R. and Bajorath, J., 2020 ‘Interpretation of machine learning models using Shapley Values: Application to compound potency and multi-target activity predictions’, Journal of Computer-Aided Molecular Design, 34(10), pp. 1013–1026. doi:10.1007/s10822-020-00314-0.
33. Roslidar, R., Rahman, A., Muharar, R., Syahputra, M.R., Arnia, F., Syukri, M., Pradhan, B. and Munadi, K., 2020. A review on recent progress in thermal imaging and deep learning approaches for breast cancer detection. IEEE access, 8, pp.116176-116194.
34. Sarker, I.H., 2021. Machine learning: Algorithms, real-world applications and research directions. SN computer science, 2(3), p.160.
35. Sharma, A. and Rani, R., 2021. A systematic review of applications of machine learning in cancer prediction and diagnosis. Archives of Computational Methods in Engineering, 28(7), pp.4875-4896.
36. Sheth, V., Tripathi, U. and Sharma, A., 2022. A Comparative Analysis of Machine Learning Algorithms for Classification Purpose. Procedia Computer Science, 215, pp.422-431.
37. Shrivastava, Dr.V. and BURI, R.B., 2023 ‘Early risk prediction of breast cancer among patients using machine learning techniques’, SSRN Electronic Journal [Preprint]. doi:10.2139/ssrn.4361052.
38. Tarawneh, O., Otair, M., Husni, M., Abuaddous, H.Y., Tarawneh, M. and Almomani, M.A., 2022. Breast cancer classification using decision tree algorithms. International Journal of Advanced Computer Science and Applications, 13(4).
39. Telsang, V.A. and Hegde, K., 2020, December. Breast cancer prediction analysis using machine learning algorithms. In 2020 International Conference on Communication, Computing and Industry 4.0 (C2I4) (pp. 1-5). IEEE.
40. Viswanatha, V., 2023. Breast cancer classification using logistic regression.
41. Wibowo, V.V.P., Rustam, Z., Laeli, A.R. and Sa'id, A.A., 2021, December. Logistic regression and logistic regression-genetic algorithm for classification of liver cancer data. In 2021 International Conference on Decision Aid Sciences and Application (DASA) (pp. 244-248). IEEE.
42. Wilkinson, L. and Gathani, T., 2022. Understanding breast cancer as a global health concern. The British journal of radiology, 95(1130), p.20211033.

43. World Health Organization (WHO) (2020). Cancer. Available at: https://www.who.int/news-room/fact-sheets/detail/cancer (Accessed: 04 February 2024).
44. World Health Organization, 2022. Breast cancer: Key facts. Available at: https://www.who.int/news-room/fact-sheets/detail/breast-. (Accessed: 25 March 2024).
45. Yaqoob, A., Musheer Aziz, R. and Verma, N.K. (2023) 'Applications and techniques of machine learning in cancer classification: A systematic review', Human-Centric Intelligent Systems, 3(4), pp. 588–615. doi:10.1007/s44230-023-00041-3.
46. Yaqoob, A., Musheer Aziz, R. and Verma, N.K., 2023. Applications and techniques of machine learning in cancer classification: a systematic review. Human-Centric Intelligent Systems, pp.1-28.
47. Yassin, N.I.R. et al., 2018 'Machine learning techniques for breast cancer computer aided diagnosis using different image modalities: A systematic review', Computer Methods and Programs in Biomedicine, 156, pp. 25–45. doi:10.1016/j.cmpb.2017.12.012.
48. Yue, W., Wang, Z., Chen, H., Payne, A. and Liu, X., 2018. Machine learning with applications in breast cancer diagnosis and prognosis. Designs, 2(2), p.13.
49. Zolfaghari, B., Mirsadeghi, L., Bibak, K. and Kavousi, K., 2023. Cancer prognosis and diagnosis methods based on ensemble learning. ACM Computing Surveys, 55(12), pp.1-34.