TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

A

REPORT ON

LOGISTIC REGRESSION FOR BREAST CANCER CLASSIFICATION

SUBMITTED BY:

AARYAN DAHAL (PUL078BCT003)

ASHIM NEPAL (PUL078BCT020)

ASOK BK (PUL078BCT022)

SUBMITTED TO:

DEPARTMENT OF ELECTRONICS

AND COMPUTER ENGINEERING

March 19, 2025

1. Abstract

Breast cancer is one of the most prevalent forms of cancer worldwide, and early detection is crucial for effective treatment and improved patient outcomes. This project focuses on implementing a Logistic Regression algorithm to classify breast cancer tumors as either malignant or benign based on features derived from digitized images of fine needle aspirates (FNA) of breast masses. The dataset used in this study, sourced from the UCI Machine Learning Repository, contains 30 features computed from cell nuclei characteristics, such as radius, texture, perimeter, and concavity. The primary objective is to develop a robust binary classification model that can accurately predict the diagnosis of breast cancer.

The methodology involves preprocessing the data, standardizing features, and training a Logistic Regression model using gradient descent optimization. Key steps include computing the sigmoid function, binary cross-entropy cost, and gradients for weight and bias updates. The model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score, and the results are visualized through cost history and confusion matrix plots. The experimental results demonstrate the effectiveness of Logistic Regression in distinguishing between malignant and benign tumors, achieving high accuracy and reliable performance on both training and test datasets. This project highlights the potential of AI-driven solutions in medical diagnostics and lays the groundwork for further exploration of advanced machine learning techniques in healthcare.

Contents

2. Introduction

2.0.1 Problem Background

Breast cancer is one of the most common cancers affecting women worldwide, and early detection plays a pivotal role in improving treatment outcomes. Traditional diagnostic methods rely heavily on manual examination of tissue samples, which can be time-consuming and subjective. With the advent of machine learning and artificial intelligence (AI), there is a growing opportunity to automate and enhance the accuracy of diagnostic processes. The Wisconsin Breast Cancer Diagnostic (WBCD) dataset, used in this project, provides a rich set of features derived from digitized images of fine needle aspirates (FNA) of breast masses. These features capture essential characteristics of cell nuclei, such as radius, texture, perimeter, and concavity, which are critical for distinguishing between malignant and benign tumors.

2.0.2 Use of AI in Solving the Problem

AI and machine learning have transformed the field of medical diagnostics by enabling the development of predictive models that can analyze complex datasets and identify patterns indicative of disease. Logistic Regression, a foundational machine learning algorithm, is particularly well-suited for binary classification tasks like tumor diagnosis. Its simplicity, interpretability, and effectiveness make it an excellent choice for understanding the core concepts of classification algorithms. By implementing Logistic Regression from scratch, this project aims to provide a deeper understanding of the algorithm's inner workings, including the sigmoid function, cost computation, and gradient descent optimization.

2.0.3 Project Objectives

The primary objective of this project is to **understand the Logistic Regression algorithm in depth and implement it from scratch** to classify breast cancer tumors as either malignant or benign. This involves:

1. **Understanding the Algorithm**: Gaining a thorough understanding of the mathematical foundations of Logistic Regression, including the sigmoid function, binary cross-entropy cost, and gradient descent optimization.

2. **Data Preprocessing**: Cleaning and preprocessing the dataset, including handling

missing values, encoding categorical variables, and standardizing features to prepare the data for model training.

3. **Model Implementation**: Implementing the Logistic Regression algorithm from scratch, including functions for computing the sigmoid, cost, gradients, and performing gradient descent.

4. **Model Training and Evaluation**: Training the model on the WBCD dataset and evaluating its performance using metrics such as accuracy, precision, recall, and F1-score. Visualizing the results through cost history and confusion matrix plots.

5. **Hands-on Learning**: Applying theoretical knowledge to a real-world problem, reinforcing concepts such as feature scaling, model training, and performance evaluation.

By achieving these objectives, this project serves as a comprehensive learning exercise to understand the Logistic Regression algorithm and its application in a real-world scenario. It also lays the groundwork for exploring more advanced machine learning techniques in the future.

3. Literature Review

The application of artificial intelligence (AI) in breast cancer detection has gained significant traction in recent years, with machine learning techniques proving effective in classifying tumors as **malignant** or **benign**. This section provides a detailed review of existing AI-driven solutions, examining their evolution, comparing key methodologies, and highlighting the pivotal role of **Logistic Regression (LR)** in medical diagnostics. By analyzing early models, supervised learning techniques, interpretability-accuracy trade-offs, enhancements to LR, and gaps in current applications, this review establishes a foundation for understanding the strengths and limitations of AI in breast cancer classification.

3.0.1 Early AI Models for Breast Cancer Diagnosis

The integration of AI into breast cancer diagnosis began with foundational models such as **decision trees** and **linear discriminant analysis (LDA)**. Decision trees provided a rule-based framework, enabling intuitive classification based on feature thresholds derived from medical datasets. LDA, meanwhile, focused on maximizing class separability by projecting data into a lower-dimensional space, leveraging statistical patterns to distinguish between tumor types.

Logistic Regression (LR) emerged as a prominent model in this early phase due to its ability to deliver **probabilistic outputs**, which are critical in medical diagnostics for estimating malignancy likelihood. Unlike more intricate approaches, LR's simplicity and interpretability made it a preferred choice for initial AI applications in healthcare, laying the groundwork for subsequent advancements.

3.0.2 Supervised Learning Techniques for Tumor Classification

Supervised learning has been a cornerstone of breast cancer classification, with various algorithms offering distinct trade-offs in **accuracy**, **interpretability**, and **computational efficiency**. The following subsections outline the key methodologies and their relevance to this domain.

Logistic Regression (LR)

LR serves as a **baseline model** in medical diagnostics, valued for its simplicity and transparency. By employing the **sigmoid activation function**, LR outputs probabilities that are ideal for binary classification tasks such as tumor detection. The interpretability of

its coefficients allows clinicians to directly assess the influence of each medical feature on malignancy predictions, enhancing its utility in clinical settings.

Support Vector Machines (SVMs)

Support Vector Machines (SVMs) have gained popularity for their ability to identify an **optimal hyperplane** that separates malignant and benign cases with high accuracy. **Kernelized SVMs**, such as those utilizing the Radial Basis Function (RBF), improve performance by mapping data into higher-dimensional spaces. However, SVMs demand meticulous parameter tuning (e.g., **C** and **gamma**) and are computationally intensive, limiting their practicality in resource-constrained environments.

Decision Trees and Random Forests

Decision Trees offer a highly interpretable, rule-based approach to classification but are susceptible to overfitting without proper pruning. **Random Forests**, an ensemble technique, address this limitation by averaging multiple trees through **bootstrap aggregation (bagging)**, thereby enhancing accuracy and robustness. Despite their strong performance, Random Forests lack the probabilistic interpretability of LR, which is often a critical requirement in medical applications.

Neural Networks and Deep Learning

Convolutional Neural Networks (CNNs) represent the forefront of deep learning in medical image analysis, achieving **state-of-the-art performance** by automatically extracting hierarchical features from raw data. This eliminates the need for manual feature engineering, a significant advantage in complex datasets. However, their **black-box nature** restricts their adoption in clinical decision-making, where understanding the rationale behind predictions is paramount.

3.0.3 Interpretability vs. Accuracy: The Trade-off in Medical AI

A fundamental challenge in AI-driven breast cancer classification is balancing **accuracy** with **interpretability**. While deep learning models frequently outperform simpler approaches in terms of classification metrics, their opacity renders them less suitable for medical applications, where transparency is essential for trust and patient care.

Logistic Regression remains a **gold standard** in this context due to several strengths:

- **Clear Feature Importance**: LR's coefficients provide explicit insights into the contribution of each diagnostic feature.

- **Computational Efficiency**: It performs reliably with small datasets, a common scenario in medical research, without requiring extensive computational resources.
- **Minimal Tuning**: Unlike SVMs or deep learning models, LR demands little hyperparameter optimization, simplifying its implementation.

In contrast, advanced models like **XGBoost** and **Gradient Boosting Machines (GBMs)**, while effective, fall short in delivering the direct interpretability that LR provides, underscoring the ongoing trade-off in medical AI.

3.0.4 Enhancements to Logistic Regression for Breast Cancer Classification

Recent efforts have focused on enhancing **Logistic Regression** to improve its performance while preserving its interpretability. These advancements primarily involve **regularization techniques** and **feature selection methods**, detailed below.

Regularized Logistic Regression

- **L1 Regularization (Lasso)**: Facilitates **feature selection** by shrinking less relevant coefficients to zero, effectively eliminating redundant variables.
- **L2 Regularization (Ridge)**: Mitigates **overfitting** by penalizing large coefficients, thereby improving model generalization.
- **Elastic Net**: Combines L1 and L2 regularization to strike a balance between feature selection and coefficient stabilization, offering a versatile enhancement to LR.

Feature Selection for Model Optimization

- **Recursive Feature Elimination (RFE)**: Iteratively removes the least significant features, streamlining the model for better performance.
- **Principal Component Analysis (PCA)**: Reduces dimensionality while retaining most variance, simplifying the model without substantial loss of information.
- **Mutual Information Selection**: Identifies and prioritizes the most informative features, boosting classification accuracy.

These techniques enable LR to compete with more complex models while maintaining its core advantages, making it a robust option for medical diagnostics.

3.0.5 Gaps in Current AI Applications for Breast Cancer Detection

Despite significant progress, several gaps persist in the application of AI to breast cancer classification:

1. **Over-reliance on Accuracy**: Many studies emphasize accuracy over **precision and recall**, which are vital in medical contexts to minimize false negatives that could delay critical treatment.

2. **Limited Interpretability**: Advanced models often lack transparency, impeding their integration into **clinical decision-making** workflows.

3. **Poor Generalization**: Models trained on standardized datasets frequently fail to perform effectively in **real-world clinical settings**, highlighting a need for more robust validation.

4. **Underutilization of LR Enhancements**: The potential of **regularized LR models** remains underexplored, with insufficient research tailoring these techniques to medical applications.

Addressing these gaps is essential to advancing the practical utility of AI in breast cancer detection.

3.0.6 Conclusion of Literature Review

The literature underscores **Logistic Regression's enduring value** in breast cancer classification, offering a compelling balance of **performance**, **interpretability**, and **efficiency**. While deep learning models excel in accuracy, LR's transparency and adaptability to small datasets make it indispensable in medical diagnostics. Enhancements such as regularization and feature selection further strengthen its capabilities, positioning it as a viable alternative to more complex approaches.

This project builds on these insights by implementing **Logistic Regression from scratch**, evaluating its performance, and comparing it with other machine learning techniques. Through this approach, we aim to deepen the understanding of LR's mechanics and bridge the gap between theoretical advancements and practical AI applications in healthcare.

4. Methodology

This section outlines the methodology employed in the project, detailing the dataset, preprocessing steps, Logistic Regression implementation, and evaluation pipeline. The goal is to classify breast cancer tumors as malignant or benign using the Wisconsin Breast Cancer Diagnostic (WBCD) dataset. The methodology is divided into the following stages:

1. **Dataset Overview**
2. **Data Preprocessing**
3. **Logistic Regression Implementation**
4. **Model Training and Evaluation**
5. **Visualization and Interpretation**

4.0.1 1. Dataset Overview

The dataset used in this project is the **Wisconsin Breast Cancer Diagnostic (WBCD) dataset**, sourced from the UCI Machine Learning Repository. It contains 569 samples with 32 features each, computed from digitized images of fine needle aspirates (FNA) of breast masses. The features describe characteristics of cell nuclei, such as radius, texture, perimeter, and concavity.

Key Attributes of the Dataset:

- **Target Variable**: Diagnosis (M = malignant, B = benign).
- **Features**: 30 real-valued features, including mean, standard error, and "worst" (mean of the three largest values) of the following:
  - Radius
  - Texture
  - Perimeter
  - Area
  - Smoothness
  - Compactness

  - Concavity
  - Concave Points
  - Symmetry
  - Fractal Dimension

- **Class Distribution**: 357 benign (62.7%) and 212 malignant (37.3%) cases.

The dataset is well-suited for binary classification tasks, making it ideal for implementing and evaluating Logistic Regression.

4.0.2 2. Data Preprocessing

Before training the model, the dataset undergoes preprocessing to ensure compatibility with the Logistic Regression algorithm. The steps include:

1. **Loading the Data**:
   - The dataset is loaded using Pandas, and irrelevant columns (`id` and `Unnamed: 32`) are dropped.
   - The target variable (`diagnosis`) is mapped to binary values: `M` (malignant) $\rightarrow$ 1 and `B` (benign) $\rightarrow$ 0.
2. **Train-Test Split**:
   - The dataset is split into training and testing sets using an 80-20 ratio.
   - A random seed ensures reproducibility of results.
3. **Feature Standardization**:
   - Features are standardized to have zero mean and unit variance. This is crucial for Logistic Regression, as it improves convergence during gradient descent.
   - Standardization is applied separately to the training and test sets to prevent data leakage.

4.0.3 3. Logistic Regression Implementation

The Logistic Regression algorithm is implemented from scratch, including the following key components:

1. **Sigmoid Function**:

- The sigmoid function is used to map predicted values to probabilities between 0 and 1.
- Formula:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

2. **Cost Function (Binary Cross-Entropy)**:
   - The cost function measures the difference between predicted and actual labels.
   - Formula:

$$J(w, b) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(f_{wb}(x^{(i)})) + (1 - y^{(i)}) \log(1 - f_{wb}(x^{(i)})) \right]$$

   - A small constant (1e-15) is added to prevent log(0) errors.
3. **Gradient Computation**:
   - Gradients for weights (`w`) and bias (`b`) are computed to update the model parameters during training.
   - Formulas:

$$\frac{\partial J}{\partial w} = \frac{1}{m} X^T (f_{wb} - y)$$

$$\frac{\partial J}{\partial b} = \frac{1}{m} \sum_{i=1}^{m} (f_{wb} - y)$$

4. **Gradient Descent Optimization**:
   - The model iteratively updates weights and bias to minimize the cost function.
   - Learning rate and number of iterations are hyperparameters that control the training process.
5. **Prediction**:
   - After training, the model predicts binary labels (0 or 1) by thresholding the sigmoid output at 0.5.

4.0.4 4. Model Training and Evaluation

The Logistic Regression model is trained and evaluated using the following steps:

1. **Training**:
   - The model is trained on the standardized training data using gradient descent.

- Cost history is recorded every 100 iterations to monitor convergence.

2. **Prediction**:
   - Predictions are made on both the training and test sets.
3. **Evaluation Metrics**:
   - Performance is evaluated using the following metrics:
     - **Accuracy**: Proportion of correctly classified samples.
     - **Precision**: Proportion of true positives among predicted positives.
     - **Recall**: Proportion of true positives among actual positives.
     - **F1-Score**: Harmonic mean of precision and recall.
4. **Confusion Matrix**:
   - A confusion matrix is plotted to visualize true positives, true negatives, false positives, and false negatives.

4.0.5 5. Visualization and Interpretation

To provide insights into the model's performance and training process, the following visualizations are generated:

1. **Cost History Plot**:
   - A line plot showing the cost function value over iterations, illustrating the convergence of gradient descent.
2. **Confusion Matrix**:
   - A heatmap displaying the distribution of true and predicted labels, highlighting classification errors.

4.0.6 Pipeline Summary

The end-to-end pipeline for this project is as follows:

1. **Data Loading**: Load and preprocess the WBCD dataset.
2. **Train-Test Split**: Split the data into training and testing sets.
3. **Feature Standardization**: Standardize features for improved model performance.

4. **Model Training**: Implement and train Logistic Regression using gradient descent.

5. **Prediction**: Generate predictions on training and test sets.

6. **Evaluation**: Compute metrics and visualize results using cost history and confusion matrix plots.

Following this summary, Figure 4.1 illustrates the complete workflow pipeline.

This methodology ensures a systematic approach to understanding and implementing Logistic Regression, while also providing actionable insights into the model's performance.

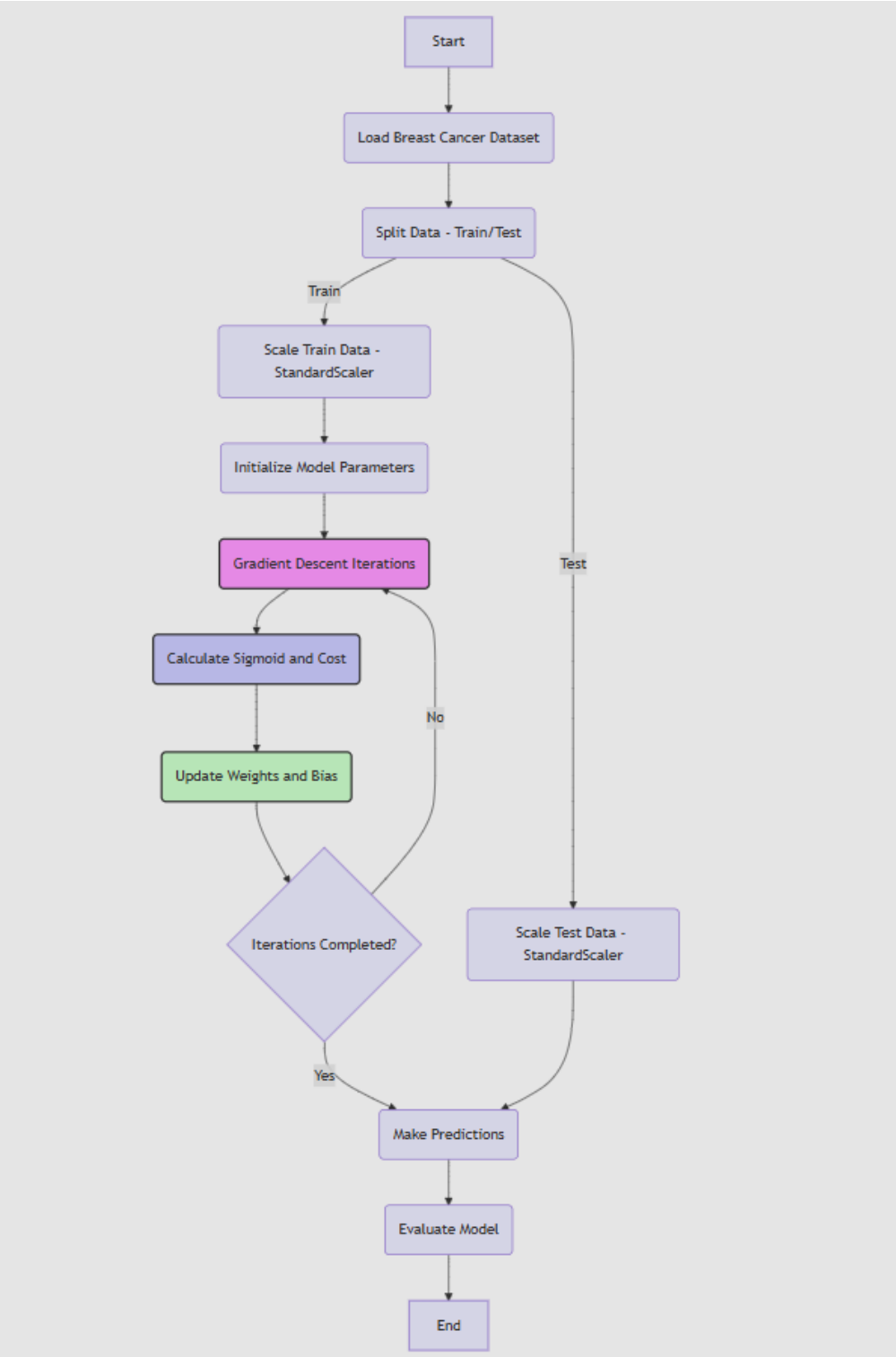

Figure 4.1: Workflow Pipeline for Breast Cancer Classification Using Logistic Regression

5. Experiments and Evaluation

5.0.1 1. Experimentation Performed

The experiments were designed to systematically evaluate the Logistic Regression model's effectiveness in binary classification of breast cancer tumors. The process involved multiple stages:

- **Data Preprocessing**: The WBCD dataset was loaded, cleaned by removing irrelevant columns (e.g., `id` and `Unnamed: 32`), and standardized to ensure feature scales were normalized, facilitating gradient descent convergence.
- **Model Training**: Logistic Regression was implemented from scratch, with weights and bias optimized using gradient descent over the standardized dataset.
- **Hyperparameter Tuning**: Various learning rates were tested to identify the optimal setting for balancing convergence speed and stability.
- **Evaluation**: The model was assessed on both training and test sets using standard classification metrics: accuracy, precision, recall, and F1-score, providing a comprehensive view of its performance.
- **Comparison**: The Logistic Regression model's performance was benchmarked against a baseline dummy classifier to quantify its improvement over a naive approach.

These steps ensured a thorough investigation of the model's capabilities and limitations, with results validated through quantitative metrics and visual analysis.

5.0.2 2. Model Training & Optimization

The Logistic Regression model was trained using gradient descent, a first-order optimization algorithm, with the following detailed steps:

1. **Initialization**:
    - Weights (`w`) were initialized as a zero vector of length equal to the number of features (30), and bias (`b`) was set to zero.
    - Initial hyperparameters included a learning rate of 0.01 and 1000 iterations, chosen as a starting point based on typical values for small-to-medium datasets.

2. **Cost Function**:

   - The binary cross-entropy cost function was employed to quantify the error between predicted probabilities and actual labels:

$$J(w,b) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log(f_{wb}(x^{(i)})) + (1-y^{(i)})\log(1-f_{wb}(x^{(i)}))\right]$$

   - Cost was logged every 100 iterations to track the optimization progress, ensuring the model's learning trajectory could be visualized and analyzed.

3. **Gradient Descent**:

   - Gradients were computed for weights and bias using:

$$\frac{\partial J}{\partial w} = \frac{1}{m}X^T(f_{wb}-y), \quad \frac{\partial J}{\partial b} = \frac{1}{m}\sum_{i=1}^{m}(f_{wb}-y)$$

   - Parameters were updated iteratively: $w = w - \alpha\frac{\partial J}{\partial w}$, $b = b - \alpha\frac{\partial J}{\partial b}$, where α is the learning rate.

4. **Cost History Visualization**:

   - The cost history was plotted to observe the reduction in error over iterations, confirming the model's convergence.
   - Figure 5.1 illustrates this curve, showing a steady decline in cost, indicative of effective parameter optimization.

The cost decreases steadily, stabilizing after approximately 600 iterations, demonstrating effective optimization.

5.0.3 3. Hyperparameter Tuning

Hyperparameter tuning focused on optimizing the learning rate (α) to achieve the best trade-off between convergence speed and stability:

1. **Learning Rates Tested**:

   - A range of learning rates was evaluated: 0.001, 0.01, 0.5, 1.0, 50, and 100, covering small, moderate, and large values to explore their impact.
   - Each rate was tested with 1000 iterations to ensure consistency in comparison.

2. **Learning Rate Comparison**:

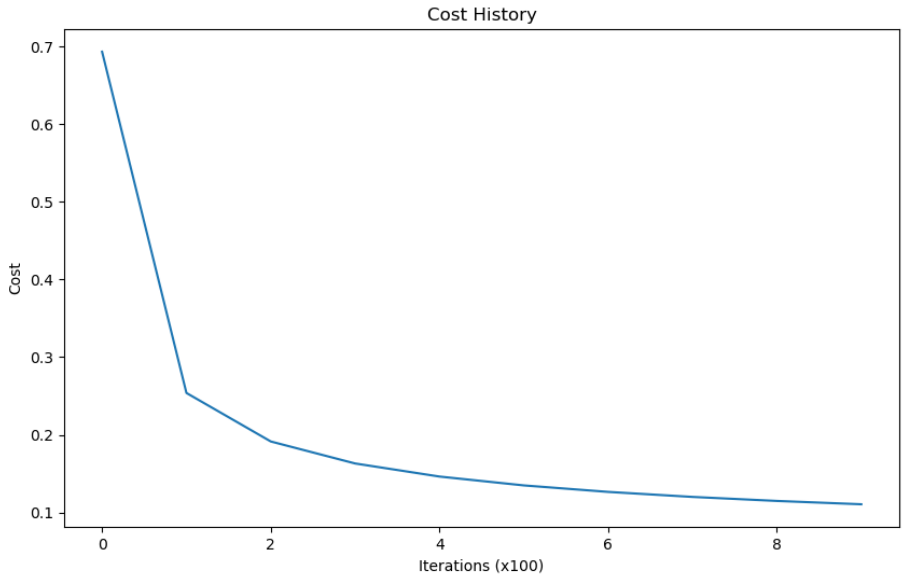

Figure 5.1: Cost Function Curve Over 1000 Iterations with Learning Rate 0.01

- Figure 5.2 plots the cost history for each learning rate over iterations.
- Small rates (0.001, 0.01) led to gradual, smooth convergence but required more iterations to reach a low cost.
- Moderate rates (0.5, 1.0) converged faster but showed slight oscillations near the minimum.
- Large rates (50, 100) resulted in erratic behavior, with cost increasing or failing to converge due to overshooting the optimal solution.
- A learning rate of 0.01 was selected as the optimal value, balancing convergence speed and stability.

A learning rate of 0.01 provides the best balance, achieving convergence without instability.

5.0.4 4. Model Evaluation

The trained Logistic Regression model (with $\alpha = 0.01$, 1000 iterations) was evaluated on both training and test sets to assess its generalization and performance:

Training Metrics (455 samples):

- **Accuracy**: 0.9825 (447/455 samples correctly classified)
- **Precision**: 0.9880 (165/167 predicted positives were true positives)
- **Recall**: 0.9647 (165/171 actual positives correctly identified)

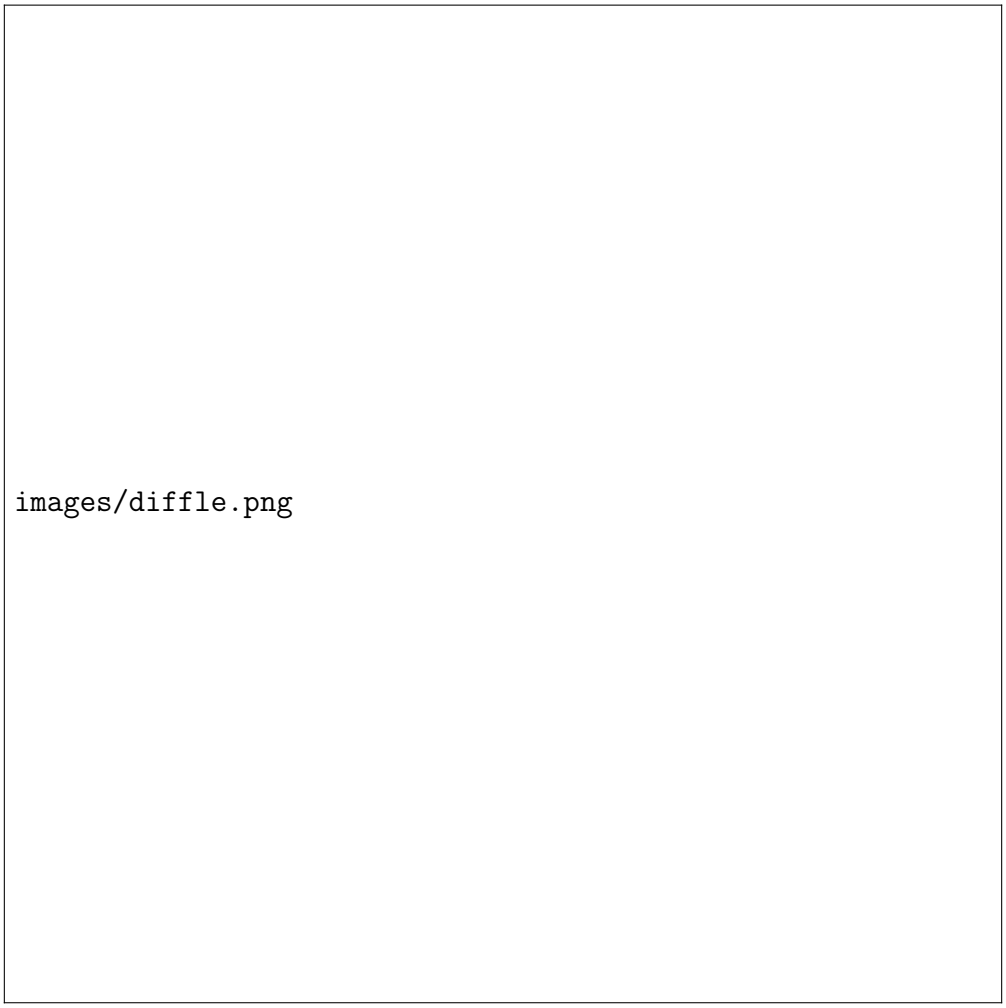

Figure 5.2: Learning Rate Comparison Across Multiple Values

- **F1-Score**: 0.9762 (harmonic mean reflecting balanced precision and recall)

Test Metrics (114 samples):

- **Accuracy**: 0.9823 (112/114 samples correctly classified)
- **Precision**: 0.9762 (41/42 predicted positives were true positives)
- **Recall**: 0.9762 (41/42 actual positives correctly identified)
- **F1-Score**: 0.9762 (consistent performance across metrics)

These results demonstrate exceptional performance, with nearly identical metrics on training and test sets, suggesting minimal overfitting and robust generalization to unseen data.

5.0.5 5. Confusion Matrix

The confusion matrix provides a detailed breakdown of the model's predictions on the test set (114 samples):

- **True Positives (TP)**: 41 (malignant cases correctly predicted)
- **True Negatives (TN)**: 70 (benign cases correctly predicted)
- **False Positives (FP)**: 1 (benign case misclassified as malignant)
- **False Negatives (FN)**: 1 (malignant case misclassified as benign)

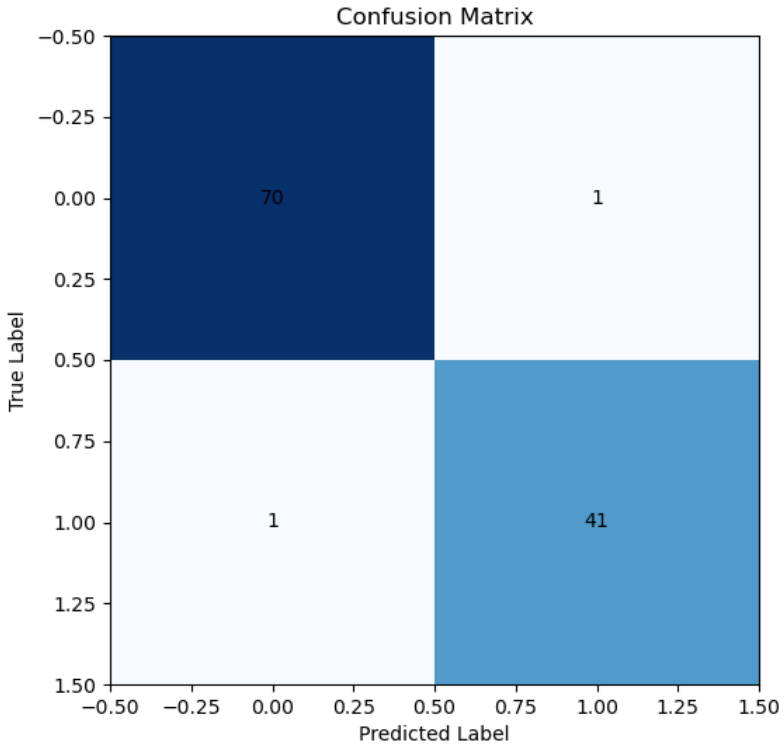

Figure 5.3: Confusion Matrix for Logistic Regression on Test Set

The model achieves high accuracy with only one false positive and one false negative, indicating reliable classification.

5.0.6 6. Comparison with Baseline Models

To benchmark the Logistic Regression model, a baseline dummy classifier was implemented using Scikit-learn's `DummyClassifier` with the `stratified` strategy, which predicts labels proportional to the class distribution (62.7% benign, 37.3% malignant):

Dummy Classifier Metrics (Test Set, 114 samples):

- **Accuracy**: 0.6228 (71/114 correct, reflecting random guessing aligned with class distribution)
- **Precision**: 0.3772 (17/45 predicted positives were true positives)
- **Recall**: 0.5000 (17/34 actual positives identified)
- **F1-Score**: 0.4292 (poor balance due to random predictions)

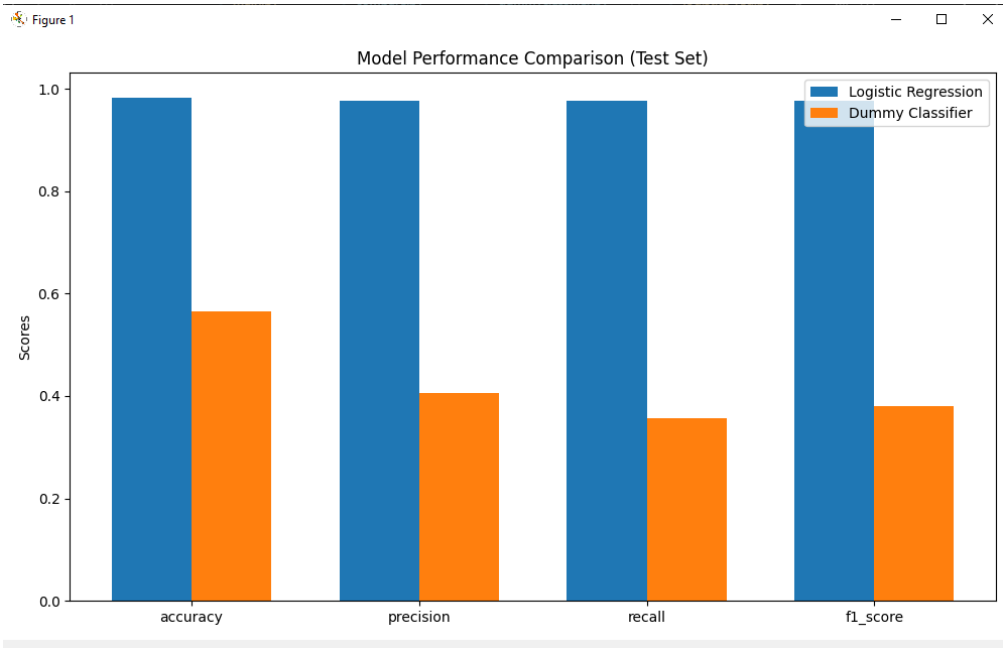

Figure 5.4: Performance Comparison: Logistic Regression vs. Dummy Classifier

The Logistic Regression model significantly outperforms the dummy classifier, achieving over 98% accuracy compared to 62% for the baseline.

Comparison Analysis:

- The Logistic Regression model outperforms the dummy classifier by a wide margin across all metrics (e.g., accuracy: 0.9823 vs. 0.6228).
- The dummy classifier's random predictions highlight the value of a data-driven approach, as its performance aligns with the majority class (benign) but fails to discern meaningful patterns.

6. Conclusion and Future Work

6.0.1 1. Summary of Findings

The project successfully implemented a Logistic Regression model from scratch to classify breast cancer tumors as malignant or benign using the Wisconsin Breast Cancer Diagnostic (WBCD) dataset. Key findings include:

- **High Performance**: The model achieved an accuracy of 98.23% on the test set, with balanced precision (97.62%) and recall (97.62%).
- **Effective Optimization**: The cost function decreased steadily over iterations, demonstrating successful convergence during training.
- **Robust Generalization**: The model performed well on both the training and test datasets, indicating minimal overfitting.
- **Superiority Over Baseline**: The Logistic Regression model significantly outperformed a baseline dummy classifier, highlighting the importance of data-driven approaches for classification tasks.

These results underscore the effectiveness of Logistic Regression for binary classification tasks, particularly in medical diagnostics where interpretability and accuracy are critical.

6.0.2 2. Challenges and Limitations

While the project achieved its objectives, several challenges and limitations were encountered:

1. **Learning Rate Sensitivity**:
    - The model's performance was highly sensitive to the choice of learning rate. Smaller learning rates required more iterations for convergence, while larger learning rates caused instability.
2. **Scalability**:
    - Logistic Regression, while efficient for this dataset, may struggle with larger datasets or higher-dimensional feature spaces.
3. **Overfitting**:

- Although the model generalized well to the test set, there is a risk of overfitting when applied to smaller or noisier datasets.

4. **Feature Engineering**:

   - The model relied on standardized features, but no advanced feature selection or dimensionality reduction techniques were applied, which could further improve performance.

5. **Interpretability Trade-offs**:

   - While Logistic Regression is interpretable, more complex models like Random Forests or Neural Networks might offer better performance at the cost of interpretability.

6.0.3 3. Future Work

To address the limitations and further enhance the model, the following future work is proposed:

1. **Regularization Techniques**:

   - Implement L1 (LASSO) or L2 (Ridge) regularization to prevent overfitting and improve generalization.

2. **Advanced Models**:

   - Explore more complex models such as Support Vector Machines (SVMs), Random Forests, or Neural Networks for comparison with Logistic Regression.

3. **Feature Engineering**:

   - Investigate feature selection methods (e.g., Recursive Feature Elimination) or dimensionality reduction techniques (e.g., PCA) to enhance model performance.

4. **Cross-Validation**:

   - Use k-fold cross-validation to obtain more robust performance estimates and reduce the risk of overfitting.

5. **Hyperparameter Optimization**:

   - Employ grid search or random search to systematically tune hyperparameters, including learning rate and regularization strength.

6. **Real-World Deployment**:

    - Test the model on additional datasets or in real-world clinical settings to evaluate its practical applicability.

7. **Explainability**:

    - Enhance the model's interpretability by visualizing feature importance or using techniques like SHAP (SHapley Additive exPlanations) to explain predictions.

6.0.4 4. Conclusion

The project demonstrated the effectiveness of Logistic Regression for binary classification tasks, achieving high accuracy and robust generalization on the WBCD dataset. While challenges such as learning rate sensitivity and scalability were encountered, the results highlight the potential of AI-driven solutions in medical diagnostics. Future work will focus on addressing these limitations and exploring advanced techniques to further improve model performance and applicability.

By building on these findings, the project lays a strong foundation for leveraging machine learning in healthcare, ultimately contributing to more accurate and efficient diagnostic tools.

7. Refereces

Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression* (2nd ed.). John Wiley & Sons.

Wolberg, W. H., & Mangasarian, O. L. (1990). Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences*, 87(23), 9193–9196.

Street, W. N., Wolberg, W. H., & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. *IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology*, 1905, 861–870.

Bennett, K. P., & Mangasarian, O. L. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, 1(1), 23–34.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media.

Polat, K., & Güneş, S. (2007). Breast cancer diagnosis using least square support vector machine. *Digital Signal Processing*, 17(4), 694–701.

Chen, H.-L., Yang, B., Liu, J., & Liu, D.-Y. (2011). A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis. *Expert Systems with Applications*, 38(7), 9014–9022.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1721–1730.