Project 1 - Logistic Regression

Hubert Bujakowski
Mikołaj Gałkowski
Julia Przybytniowska

April 2024

1 Introduction

Logistic regression is a fundamental machine learning algorithm widely used for binary classification tasks. It models the probability of a binary outcome based on one or more predictor variables. In this project, we aim to implement different optimization algorithms for logistic regression and compare their performance.

Project Overview

The project is divided into three main tasks:

1. **Task 1: Data Collection and Preparation**
   The objective of this task is to collect and prepare datasets for conducting experiments. We obtained nine different datasets from various sources, each containing a binary class variable and numerical predictor variables. These datasets were carefully selected to ensure diversity and suitability for our experiments. Preprocessing steps included handling missing values and removing collinear variables.

2. **Task 2: Implementation of Optimization Algorithms**
   In this task, we implemented three optimization algorithms for parameter estimation in logistic regression: Iterative Reweighted Least Squares (IRLS), Stochastic Gradient Descent (SGD), and Adaptive Moment Estimation (ADAM). Each algorithm was designed to support interactions between variables, which can enhance the model's flexibility and predictive power.

3. **Task 3: Experimentation and Analysis**
   The third task involves conducting experiments to compare the performance of the optimization algorithms. We proposed a common stopping rule for all algorithms and evaluated their performance using Balanced Accuracy as the performance measure. Additionally, convergence analysis was performed to examine the convergence behavior of each algorithm. Finally, we compared the classification performance of logistic regression with other popular classification methods, such as Linear Discriminant Analysis, Quadratic Discriminant Analysis, Decision Trees, and Random Forests.

2 Methodology

In our project, we utilized datasets available from repositories such as OpenML and UCI. From the OpenML website, we specifically chose 3 small datasets, each containing a limited number of descriptive variables (less than 10). Additionally, we picked 5 large datasets from OpenML and included 1 large dataset sourced from UCI repository. For datasets classified as large, we considered those with a number of descriptive variables exceeding 10, post the removal of highly correlated variables from the analysis.

We prioritized selecting datasets not previously used in other courses, which may result in less known datasets being included:

- Small datasets:
  - Iris (Id=969)
  - Hayes-Roth (Id=974)
  - Banknote Authentication (Id=1462)
- Large datasets:
  - fri_c0_1000_25 (Id=849)
  - fri_c2_500_25 (Id=879)
  - autoUniv-au1-1000 (Id=1547)
  - Bank dataset (bank32nh) (Id=833)
  - Breast Cancer Wisconsin (Diagnostic) (Id=1510)
  - Connectionist Bench (Sonar, Mines vs. Rocks) (UCI Id=151)

Then, the selected data sets were properly prepared for the correct operation of the logistic regression algorithm. Although our data was devoid of missing values, we added this functionality to our implementation to standardize the code, replacing the missing values with the average of the corresponding column. In addition, to streamline the modeling process, we conducted a detailed exploration of variable correlations and their selection. Any columns correlated more than 80% were identified and then the redundant ones were removed.

To evaluate the performance of our models, we conducted comprehensive experiments involving multiple training-test splits. Each model was trained on 5 different training-test splits, with the random_state parameter changing in each iteration to ensure different data splits (the seeds were set to values of 0,1,2,3,4 to make the results reproducible). To make the comparisons valid, we kept a single test size value of 20%, ensuring consistency across all experiments.

The stopping condition in the logistic regression optimization loop is based on monitoring the change in model weights between iterations. If the Euclidean distance between the current weights and the weights from the previous iteration falls below a predefined tolerance threshold (10^{-5}), the optimization process halts, indicating convergence.

3 Convergence Analysis

In this section, we analyze the convergence behavior of the optimization algorithms by examining the value of the log-likelihood function over iterations.

Before describing the log-likelihood function, let's define the likelihood function $\mathcal{L}$. The likelihood function represents the probability of observing the given data under the logistic regression model. For binary classification tasks, the likelihood function is defined as:

$$\mathcal{L}(\mathbf{w}) = \prod_{i=1}^{N} p_i^{y_i}(1-p_i)^{1-y_i} \tag{1}$$

where $\mathbf{w}$ represents the model parameters (weights), N is the number of observations, y_i is the true label of the i-th observation, and p_i is the predicted probability of the positive class for the i-th observation.

The log-likelihood function, denoted as l, is the logarithm of the likelihood function. It is given by:

$$l(\mathbf{w}) = \sum_{i=1}^{N} [y_i \log(p_i) + (1-y_i)\log(1-p_i)] \tag{2}$$

To assess the convergence of the optimization algorithms, we track the value of the log-likelihood function over iterations. Convergence is typically achieved when the change in the log-likelihood function becomes negligible, indicating that the optimization process has reached a stable point.

The convergence plots shown on Figure 1, designed for small datasets, and Figure 2, tailored for large datasets, illustrate the value of the log-likelihood function against the number of iterations for each optimization algorithm.

The convergence patterns of optimization algorithms, as depicted in Figures 1 and 2, provide intriguing insights. In some instances, all optimizers exhibit similar convergence behaviors, resulting in overlapping plots, particularly notable in datasets like 974 and 1510. However, there are occasional occurrences where one optimizer consistently outperforms the others across all datasets.

In our analysis, the IRLS algorithm generally performs worse on large datasets, with one notable exception observed in dataset 151, where its performance is better. This observation highlights the significant influence of dataset characteristics on algorithm performance. The nature and distribution of data play a crucial role in shaping how optimization techniques converge and perform.

Furthermore, it's worth noting that the curve of the SGD algorithm may lack smoothness compared to algorithms like Adam and IRLS due to its practice of updating weights after each observation. This variability in behavior underscores the importance of understanding the characteristics of different optimization algorithms.

Moreover, it emphasizes that there is no universally superior optimizer. Instead, the choice of optimizer is problem-specific, akin to the "no free lunch" theorem in optimization, suggesting that no single algorithm outperforms all others across all possible problems.

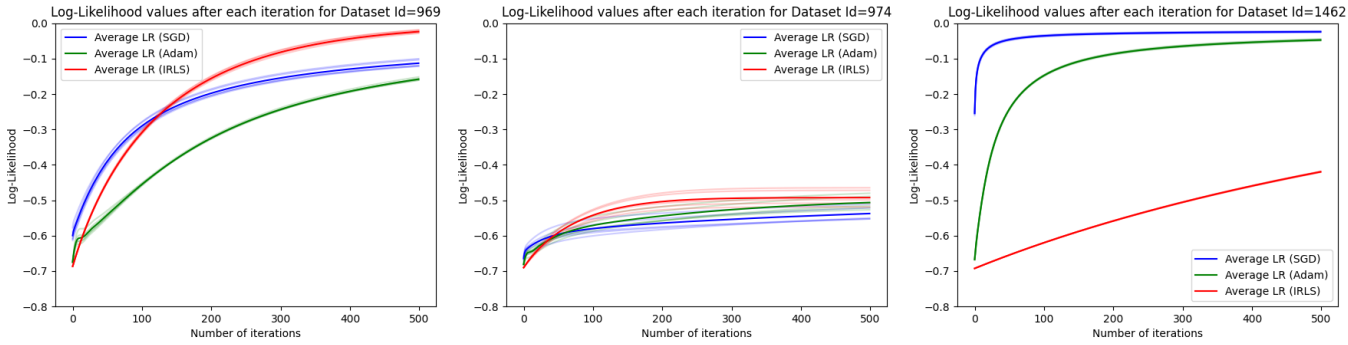

Figure 1: Small datasets

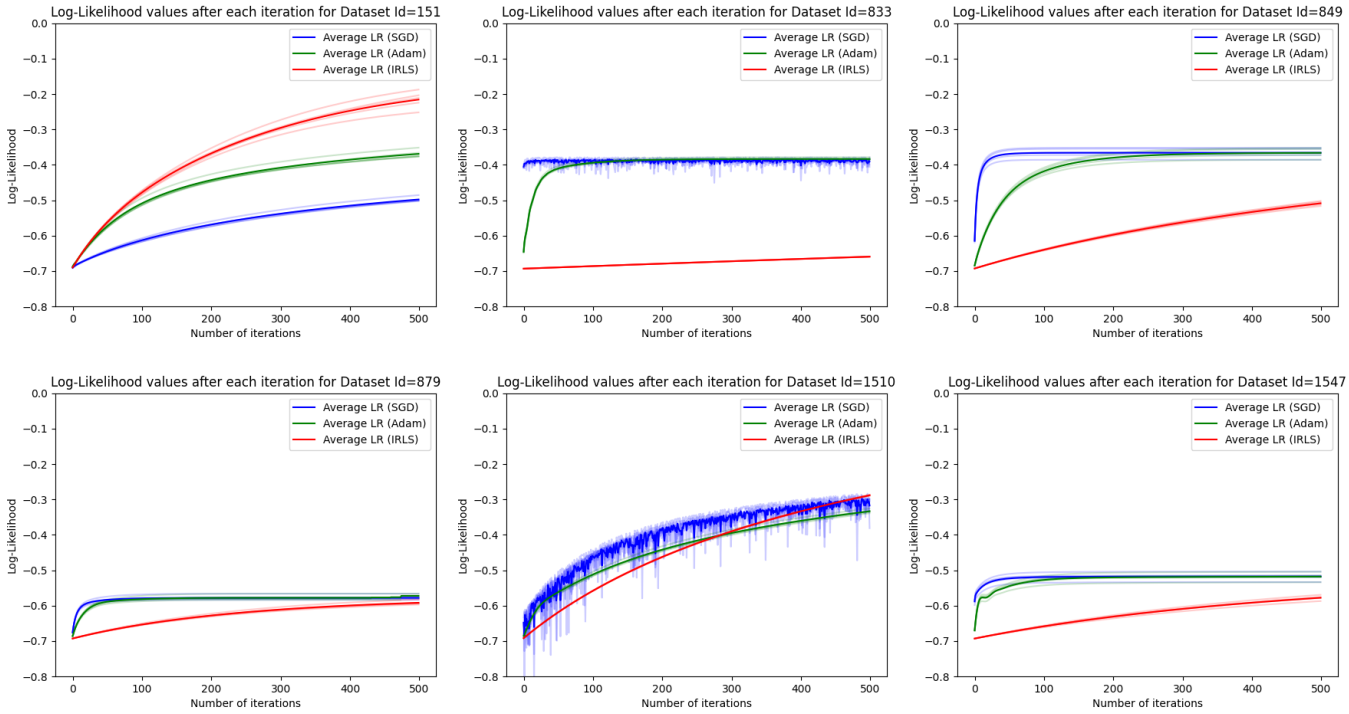

Figure 2: Large datasets

4 Comparison of classification performance

In this section, we compare the classification performance of various algorithms using the Balanced Accuracy metric. Balanced Accuracy is a performance measure commonly used for imbalanced classification tasks. It considers both sensitivity and specificity, making it suitable for evaluating models on datasets with unequal class distributions. Balanced Accuracy is calculated as follows:

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \tag{3}$$

where TP is the number of true positives, FN is the number of false negatives, TN is the number of true negatives, and FP is the number of false positives.

We compare the classification performance of logistic regression (LR) algorithms using three different optimization techniques: Stochastic Gradient Descent (SGD), Iterative Reweighted Least Squares (IRLS), and Adaptive Moment Estimation (Adam). Additionally, we compare LR with other popular classification algorithms, including Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Decision Trees, and Random Forests.

Among our three small datasets shown in the Table 1, it's notable that each algorithm achieves similar results without any significant deviations. However, dataset 974 stands out as decision tree models perform better on certain datasets, suggesting potential violations of logistic regression assumptions. Interestingly, for dataset 974, logistic regression with the ADAM optimizer outperforms Logisitic Regression optimizer with IRLS and SGD.

Regarding the large datasets shown in the Table 2, decision tree models perform the best, achieving superior results on 4 out of 6 datasets. Furthermore it can be also seen in Figure 3 that decision tree models do not significantly underperform on datasets where they don't achieve best results, unlike other algorithms compared to when decision tree models perform the best.

In general, no clear winner emerges among the optimization algorithms for logistic regression on large datasets, as they perform similarly overall. However, there is an exception with dataset 1510, where the IRLS algorithm outperforms the other algorithms.

Table 1: Comparison of classification methods' performance across small datasets with regard to balanced accuracy.

Model	Dataset 969	Dataset 974	Dataset 1462
LR (SGD)	**0.987** $\pm$ 0.018	0.648 $\pm$ 0.141	**0.994** $\pm$ 0.003
LR (Adam)	0.980 $\pm$ 0.029	0.697 $\pm$ 0.069	0.985 $\pm$ 0.008
LR (IRLS)	**0.987** $\pm$ 0.018	0.663 $\pm$ 0.066	0.973 $\pm$ 0.009
LDA	**0.987** $\pm$ 0.018	0.656 $\pm$ 0.082	0.972 $\pm$ 0.011
QDA	**0.987** $\pm$ 0.018	0.639 $\pm$ 0.095	0.984 $\pm$ 0.008
Decision Tree	0.975 $\pm$ 0.026	**0.742** $\pm$ 0.150	0.985 $\pm$ 0.010
Random Forest	0.981 $\pm$ 0.017	0.733 $\pm$ 0.130	0.993 $\pm$ 0.006

Table 2: Comparison of classification methods' performance across large datasets with regard to balanced accuracy.

Model	Dataset 151	Dataset 833	Dataset 849	Dataset 879	Dataset 1510	Dataset 1547
LR (SGD)	0.748 $\pm$ 0.035	**0.768** $\pm$ 0.022	0.810 $\pm$ 0.028	0.589 $\pm$ 0.028	0.856 $\pm$ 0.027	0.534 $\pm$ 0.018
LR (Adam)	0.769 $\pm$ 0.058	**0.768** $\pm$ 0.011	0.811 $\pm$ 0.027	0.589 $\pm$ 0.028	0.863 $\pm$ 0.025	0.526 $\pm$ 0.025
LR (IRLS)	0.750 $\pm$ 0.068	0.741 $\pm$ 0.014	0.810 $\pm$ 0.031	0.591 $\pm$ 0.034	**0.955** $\pm$ 0.022	0.526 $\pm$ 0.017
LDA	0.778 $\pm$ 0.051	0.745 $\pm$ 0.013	0.808 $\pm$ 0.031	0.588 $\pm$ 0.043	0.950 $\pm$ 0.025	0.532 $\pm$ 0.025
QDA	0.809 $\pm$ 0.070	0.752 $\pm$ 0.012	0.802 $\pm$ 0.020	0.587 $\pm$ 0.030	0.938 $\pm$ 0.013	0.600 $\pm$ 0.029
Decision Tree	0.692 $\pm$ 0.031	0.703 $\pm$ 0.012	0.778 $\pm$ 0.014	0.767 $\pm$ 0.026	0.887 $\pm$ 0.027	**0.609** $\pm$ 0.028
Random Forest	**0.833** $\pm$ 0.044	0.749 $\pm$ 0.014	**0.859** $\pm$ 0.010	**0.775** $\pm$ 0.036	0.930 $\pm$ 0.011	0.604 $\pm$ 0.019

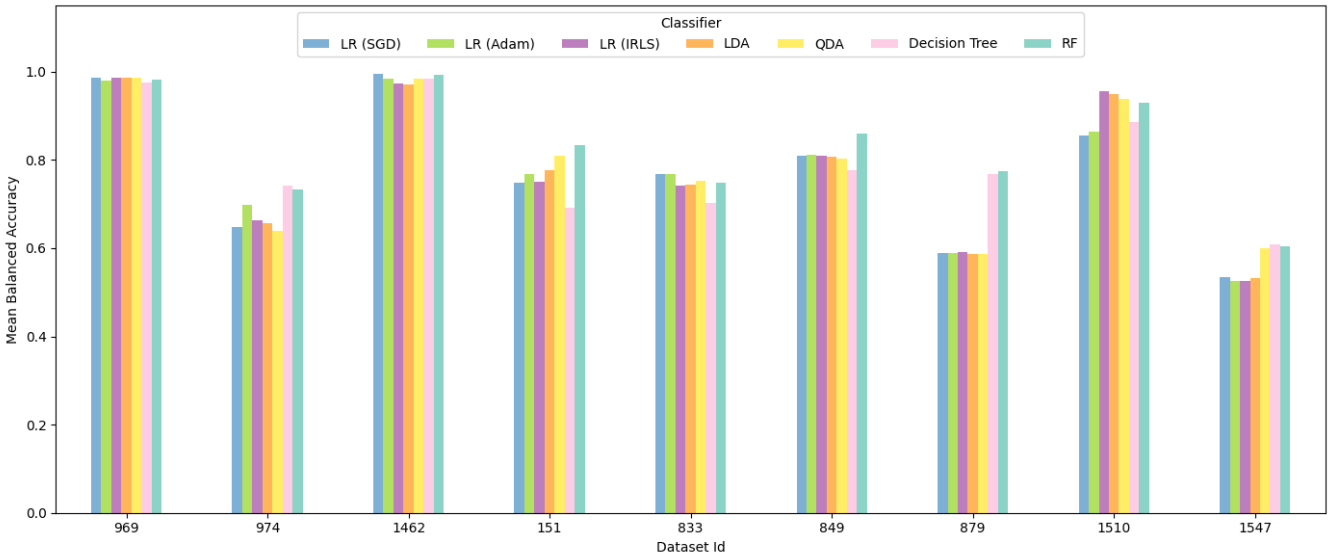

Figure 3: Model performance comparison for small and large datasets. The first three groups of bars represent results for small datasets.

5 Comparison of Classification Performance of Models with and without Interactions

In this section, we compare the classification performance of logistic regression models with and without interactions between variables. Interactions between variables are created by including products of two variables in addition to the original input variables. For example, a logistic regression model without interactions is based solely on the original input variables, while a model with interactions incorporates additional interaction terms.

We evaluate the performance of both models using the Balanced Accuracy metric and analyze the differences in their classification performance. This comparison provides insights into the impact of interactions between variables on the predictive power of logistic regression models for binary classification tasks using different optimizers (SGD, IRLS, Adam).

Upon analyzing the importance of the impact of interactions on optimization processes, we observed a nuanced effect across small datasets, where the incorporation of interactions produced varied outcomes. While interactions initially disrupted learning in two out of three datasets, they significantly enhanced performance in the third one. Conversely, with larger datasets, we noted improvements in the first and last datasets, while the remainder showed no significant enhancement.

This underscores the necessity of tailoring optimization strategies to individual datasets and classification tasks. The heterogeneous response to interactions across datasets emphasizes the need for thorough experimentation and consideration.

Furthermore, it's worth noting that the degradation of the performance of the model including interactions may be because they add noise to the data, making it harder for the model to extract real signal and useful information. Ultimately, our findings advocate for a data-driven approach, where the addition of interactions should be carefully assessed and adjusted based on the specific characteristics of each dataset and problem domain.

Table 3: Influence of interactions on classification performance across small datasets with regard to balanced accuracy.

Model	Interactions	Dataset 969	Dataset 974	Dataset 1462
LR (SGD)	-	**0.987** $\pm$ 0.018	0.648 $\pm$ 0.141	0.994 $\pm$ 0.003
LR (SGD)	+	0.970 $\pm$ 0.029	0.657 $\pm$ 0.069	**1.000** $\pm$ 0.000
LR (Adam)	-	0.980 $\pm$ 0.029	**0.697** $\pm$ 0.069	0.985 $\pm$ 0.008
LR (Adam)	+	0.973 $\pm$ 0.037	0.646 $\pm$ 0.072	0.999 $\pm$ 0.001
LR (IRLS)	-	**0.987** $\pm$ 0.018	0.663 $\pm$ 0.066	0.973 $\pm$ 0.009
LR (IRLS)	+	**0.987** $\pm$ 0.018	0.658 $\pm$ 0.071	0.981 $\pm$ 0.006

Table 4: Influence of interactions on classification performance across large datasets with regard to balanced accuracy.

Model	Interactions	Dataset 151	Dataset 833	Dataset 849	Dataset 879	Dataset 1510	Dataset 1547
LR (SGD)	-	0.748 $\pm$ 0.035	**0.768** $\pm$ 0.022	0.810 $\pm$ 0.028	0.589 $\pm$ 0.028	0.856 $\pm$ 0.027	0.534 $\pm$ 0.018
LR (SGD)	+	0.800 $\pm$ 0.025	0.755 $\pm$ 0.024	0.707 $\pm$ 0.020	0.556 $\pm$ 0.028	0.923 $\pm$ 0.034	0.613 $\pm$ 0.019
LR (Adam)	-	0.769 $\pm$ 0.058	**0.768** $\pm$ 0.011	**0.811** $\pm$ 0.027	0.589 $\pm$ 0.028	0.863 $\pm$ 0.025	0.526 $\pm$ 0.025
LR (Adam)	+	**0.803** $\pm$ 0.067	0.759 $\pm$ 0.009	0.658 $\pm$ 0.029	0.534 $\pm$ 0.026	0.929 $\pm$ 0.010	0.601 $\pm$ 0.012
LR (IRLS)	-	0.750 $\pm$ 0.068	0.741 $\pm$ 0.014	0.810 $\pm$ 0.031	**0.591** $\pm$ 0.034	**0.955** $\pm$ 0.022	0.526 $\pm$ 0.017
LR (IRLS)	+	0.798 $\pm$ 0.065	0.748 $\pm$ 0.012	0.705 $\pm$ 0.037	0.587 $\pm$ 0.039	0.945 $\pm$ 0.018	**0.615** $\pm$ 0.014

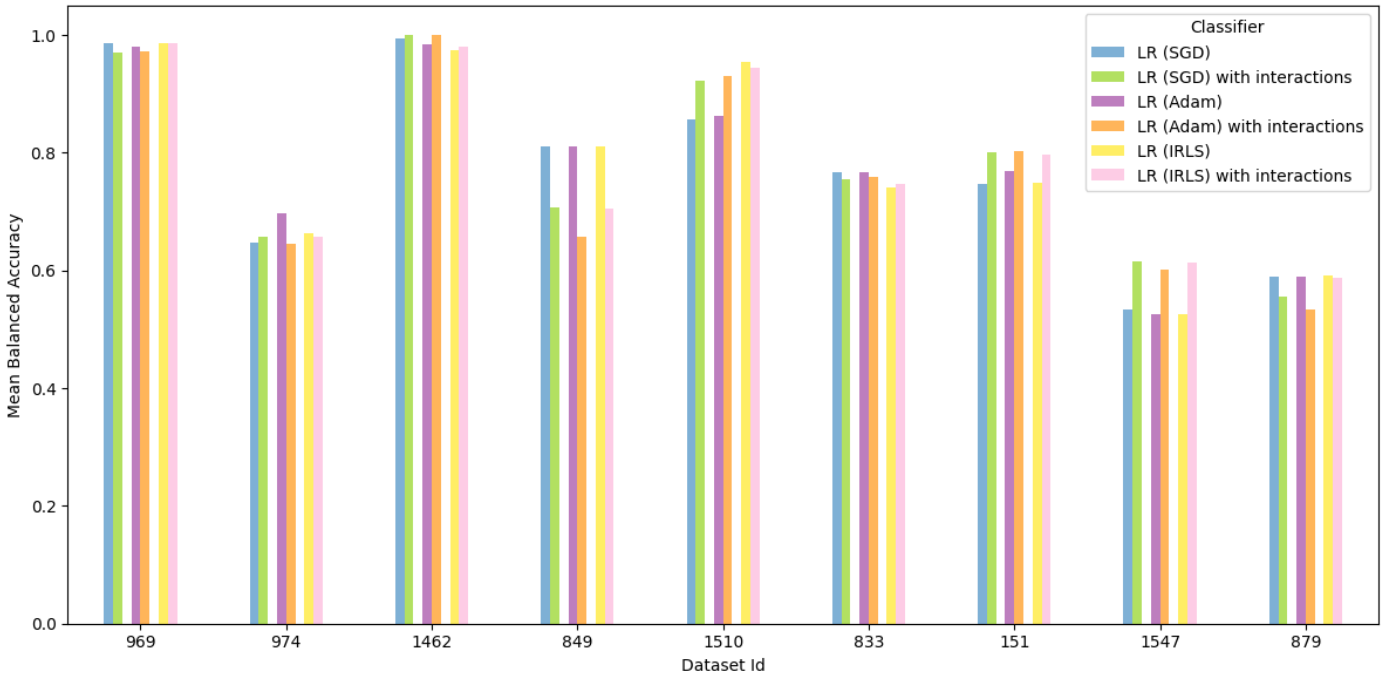

Figure 4: Logistic regression algorithm comparison with and without interactions for small and large datasets. The first three groups of bars represent results for small datasets.

6 Conclusions

In this project, we explored optimization algorithms for logistic regression and their application to binary classification tasks. Through rigorous experimentation and analysis, we gained insights into the convergence behavior and classification performance of different algorithms.

Our findings highlight the importance of selecting appropriate optimization algorithms and model configurations for binary classification tasks. We observed that incorporating interactions between variables can enhance the predictive power of logistic regression models in certain scenarios.

Overall, this project contributes to our understanding of optimization algorithms in logistic regression and provides valuable insights for future research and practical applications in machine learning and data analysis.