Implementation and Analysis of Logistic Regression using the IRLS Algorithm

MATH 5383 — Predictive Analytics

Saroar Jahan Shuba

October 17, 2025

1 Objective

The objective of this project is to implement and analyze **logistic regression using the Iteratively Reweighted Least Squares (IRLS)** algorithm, following the structured specification outlined in the course project. Specifically, the work aims to: (i) generate balanced, linearly separable synthetic data with a flexible number of predictors m; (ii) implement the IRLS algorithm to estimate model parameters with a clear formulation of the likelihood and update equations; (iii) perform parameter estimation for $m = 2$ by comparing unregularized and ℓ_2-regularized (ridge) models through decision boundary plots and β-path visualizations on log-likelihood contours; (iv) investigate the effect of strategically introduced outliers on model stability and convergence; (v) evaluate dataset-related challenges, including small vs. large sample sizes, balanced vs. imbalanced classes, and higher-dimensional cases ($m > 2$); (vi) conduct an 80/20 train–test performance evaluation to assess classification accuracy across all scenarios; and (vii) synthesize findings to interpret the role of regularization in improving convergence, robustness, and generalization of logistic regression models.

Overall, the project's objective is to demonstrate, through systematic experimentation, how regularization enhances the reliability and performance of logistic regression under diverse and progressively complex data conditions.

2 Synthetic Data Generation (Balanced, Linearly Separable)

The goal of this step was to construct a synthetic dataset suitable for evaluating the Iteratively Reweighted Least Squares (IRLS) algorithm under ideal, linearly separable conditions. Two clusters were generated corresponding to binary classes $y \in \{0, 1\}$, each drawn from distinct multivariate normal distributions. The feature matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$

was designed to be flexible, allowing any number of predictors $m \geq 2$ while maintaining a balanced number of observations per class.

For this experiment, we set $n = 80$ observations and $m = 2$ predictors for visualization. The class-0 cluster was generated with mean vector $\boldsymbol{\mu}_0 = (4, 4)$ and the class-1 cluster with $\boldsymbol{\mu}_1 = (8, 8)$, both using a common standard deviation of $\sigma = 1$. These parameters ensure that the two clusters are well separated along both feature axes. An intercept column of ones was appended to $\mathbf{X}$ to represent the bias term in the logistic regression model.

The resulting dataset exhibits clear linear separability, as shown in Figure 1. Class 0 points (blue) are concentrated in the lower-left region, while Class 1 points (red) are positioned toward the upper-right. The boundary between them can be approximated by a straight line, confirming that the dataset is suitable for logistic classification. This balanced and well-structured setup provides a clean foundation for testing and validating the IRLS algorithm before introducing more complex conditions such as outliers or imbalance in later steps.

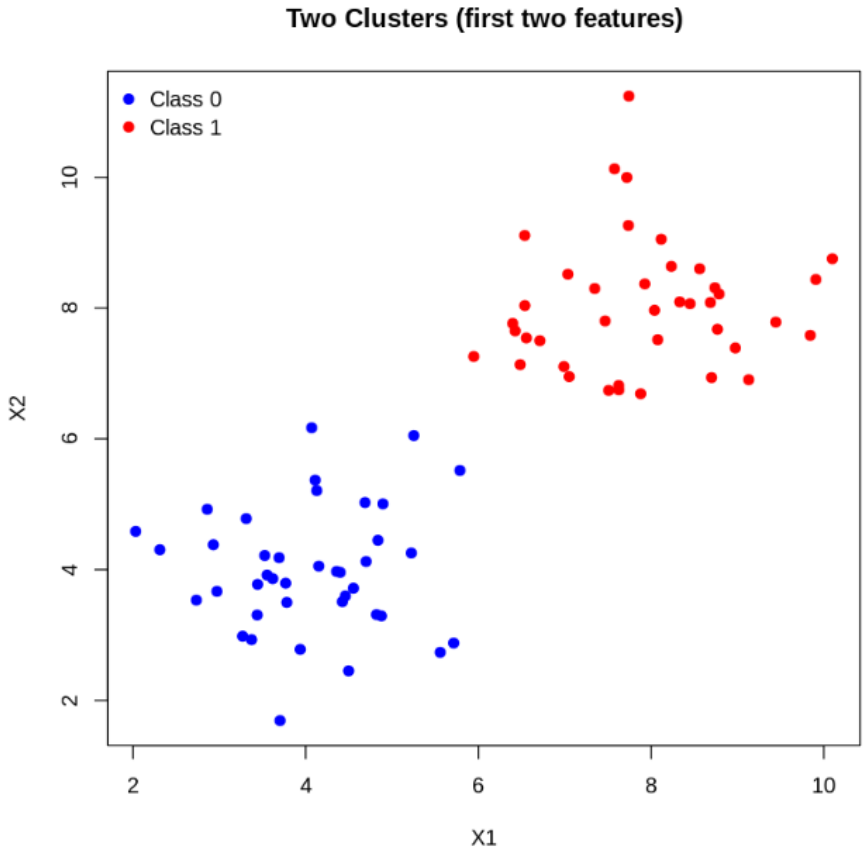

Figure 1: Two balanced, linearly separable clusters ($m = 2$). Blue represents Class 0 and red represents Class 1.

2.1 Assumptions

- The clusters are assumed to follow Gaussian distributions with equal variance, ensuring symmetric class spread.
- The dataset remains balanced, i.e., $n_0 = n_1 = 40$, preventing class bias in model training.
- For higher-dimensional experiments ($m > 2$), additional predictors are generated using the same mean-shift logic, maintaining linear separability while preserving statistical structure.

3 Implementation of IRLS (Iteratively Reweighted Least Squares)

The second stage of the project focused on implementing the **Iteratively Reweighted Least Squares (IRLS)** algorithm for logistic regression, both in its unregularized and ℓ_2-regularized (ridge) forms. This step translates the theoretical maximum-likelihood framework into a numerically stable iterative optimization process, coded and validated in R.

3.1 Model formulation and likelihood

For binary responses $y_i \in \{0, 1\}$ and predictors $\mathbf{x}_i \in \mathbb{R}^m$, the logistic model is defined as:

$$p_i = \Pr(y_i = 1 \mid \mathbf{x}_i) = \sigma(\mathbf{x}_i^\top \beta) = \frac{1}{1 + e^{-\mathbf{x}_i^\top \beta}},$$

where $\sigma(\cdot)$ is the sigmoid function. The corresponding log-likelihood is:

$$\ell(\beta) = \sum_{i=1}^{n} \Big[y_i \log p_i + (1 - y_i) \log(1 - p_i) \Big].$$

When ridge regularization is applied (excluding the intercept term), the penalized objective becomes:

$$\ell_\lambda(\beta) = \ell(\beta) - \frac{\lambda}{2} \|\beta_{-0}\|_2^2.$$

3.2 Algorithm

At each iteration, the algorithm constructs a local quadratic approximation of the log-likelihood and solves a weighted least-squares system. Let $\eta^{(t)} = X\beta^{(t)}$, $p^{(t)} = \sigma(\eta^{(t)})$,

and define the diagonal weight matrix $W^{(t)} = \mathrm{diag}\big(p^{(t)}(1-p^{(t)})\big)$. The working response is given by:

$$\mathbf{z}^{(t)} = \eta^{(t)} + \big(W^{(t)}\big)^{-1}(\mathbf{y} - p^{(t)}).$$

Then, the update rule is obtained by solving:

$$\big(X^\top W^{(t)} X + \lambda P\big)\beta^{(t+1)} = X^\top W^{(t)}\mathbf{z}^{(t)},$$

where $P = \mathrm{diag}(0, 1, \ldots, 1)$ ensures that the intercept term is not penalized.

3.3 IRLS / Newton Pseudocode

Inputs: Design matrix X (with intercept), response vector y, ridge parameter λ, tolerance ε, and maximum iterations $T_{\max}$.

Initialize:

$\beta \leftarrow \mathbf{0}$, $t \leftarrow 0$, converged $\leftarrow$ FALSE.

Repeat until convergence or $t = T_{\max}$:

$\eta \leftarrow X\beta$

$p \leftarrow \mathrm{clip}(\sigma(\eta),\, \varepsilon,\, 1-\varepsilon)$ (apply sigmoid and clip numerically)

$w \leftarrow p(1-p)$

Score: $g \leftarrow X^\top(y-p) - \lambda P\beta$

Hessian: $H_{\mathrm{neg}} \leftarrow -X^\top \mathrm{diag}(w) X - \lambda P$

Step: $\Delta \leftarrow -H_{\mathrm{neg}}^{-1} g$ (via `qr.solve` with jitter if needed)

$\beta_{\mathrm{new}} \leftarrow \beta + \Delta$

If $\max|\Delta| < \varepsilon$, set converged $\leftarrow$ TRUE and stop

$\beta \leftarrow \beta_{\mathrm{new}}$

End Repeat

Output: Estimated coefficients $\hat{\beta}$ and convergence message.

- The penalty matrix $P = \mathrm{diag}(0, 1, \ldots, 1)$ ensures the intercept is unpenalized.
- The small jitter (10^{-8}) stabilizes matrix inversion when H_{neg} is nearly singular.
- Probability clipping ($\varepsilon = 10^{-12}$) prevents overflow in $\log(p)$ or division by zero.
- When $\lambda = 0$, the algorithm performs unregularized IRLS; setting $\lambda > 0$ applies ridge shrinkage.

3.4 Implementation details and validation

The R implementation was written in modular form with clear documentation and reproducibility. The code ensures:

- Proper handling of the intercept (unpenalized).

- Safe probability clipping to avoid log(0).
- Convergence monitoring with iteration limits.
- Flexible input allowing any number of predictors ($m \geq 2$) and optional ridge penalty λ.

The implementation was verified using the clean, linearly separable dataset from Step 1. Both the unregularized ($\lambda = 0$) and ridge-regularized ($\lambda > 0$) variants produced consistent log-likelihood increases per iteration and stable coefficient estimates.

3.5 Reproducibility.

All code for this step, including the IRLS solver is available in the accompanying R script: **Google Colab Link**.

4 Parameter Estimation (Unregularized vs. Regularized, $m = 2$)

This stage applies the IRLS algorithm to two logistic regression variants: (i) the **unregularized** model ($\lambda = 0$) and (ii) the **ridge-regularized** model ($\lambda = 1$). Both are trained on the same two-cluster dataset from Step 1 to evaluate how regularization influences coefficient magnitude, convergence behavior, and decision boundaries.

4.1 Decision Boundaries

For $m = 2$, the $p = 0.5$ decision boundary satisfies

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0 \quad \Rightarrow \quad x_2 = -\frac{\beta_0 + \beta_1 x_1}{\beta_2}.$$

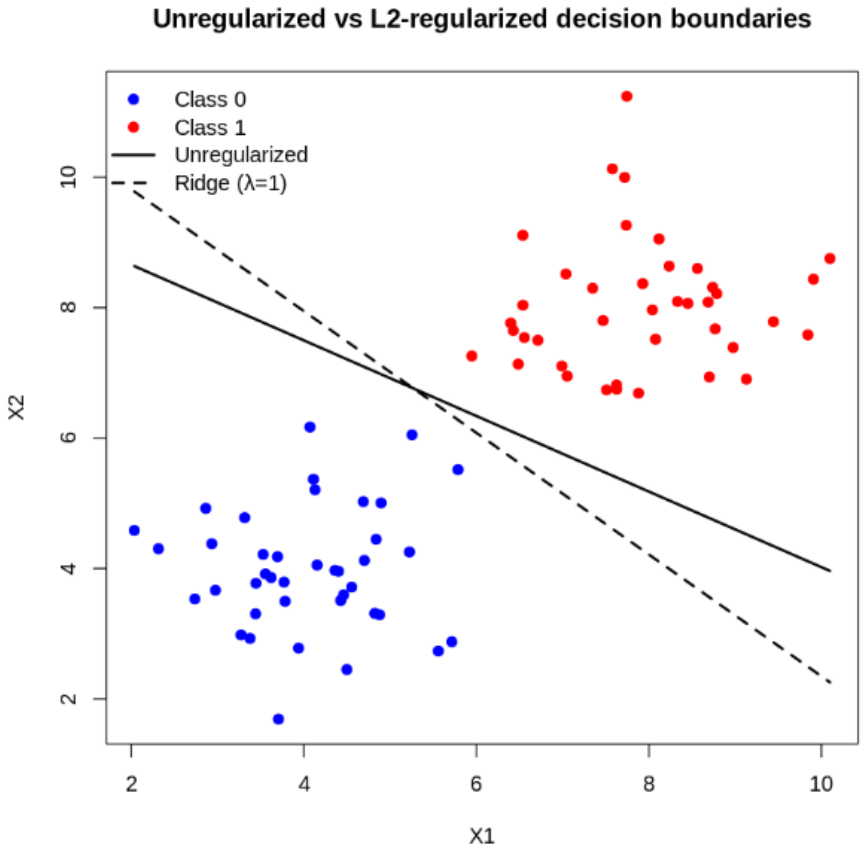

Figure 2: Decision boundaries for unregularized (solid) and ridge-regularized (dashed) IRLS fits on the same dataset.

Both models perfectly separate the two clusters, confirming the dataset's linear separability. The ridge boundary (dashed) appears slightly flatter and shifted upward compared to the unregularized boundary (solid), illustrating the effect of the ℓ_2 penalty. This penalty term, $\frac{\lambda}{2}\|\beta_{-0}\|_2^2$, constrains coefficient magnitude and prevents unbounded growth when data are separable. Regularization therefore reduces overfitting risk while maintaining the same decision geometry. Both lines are nearly parallel, indicating that ridge regularization rescales the coefficients rather than altering the direction of the separating plane.

4.2 Coefficient Trajectories on the Log-Likelihood Surface

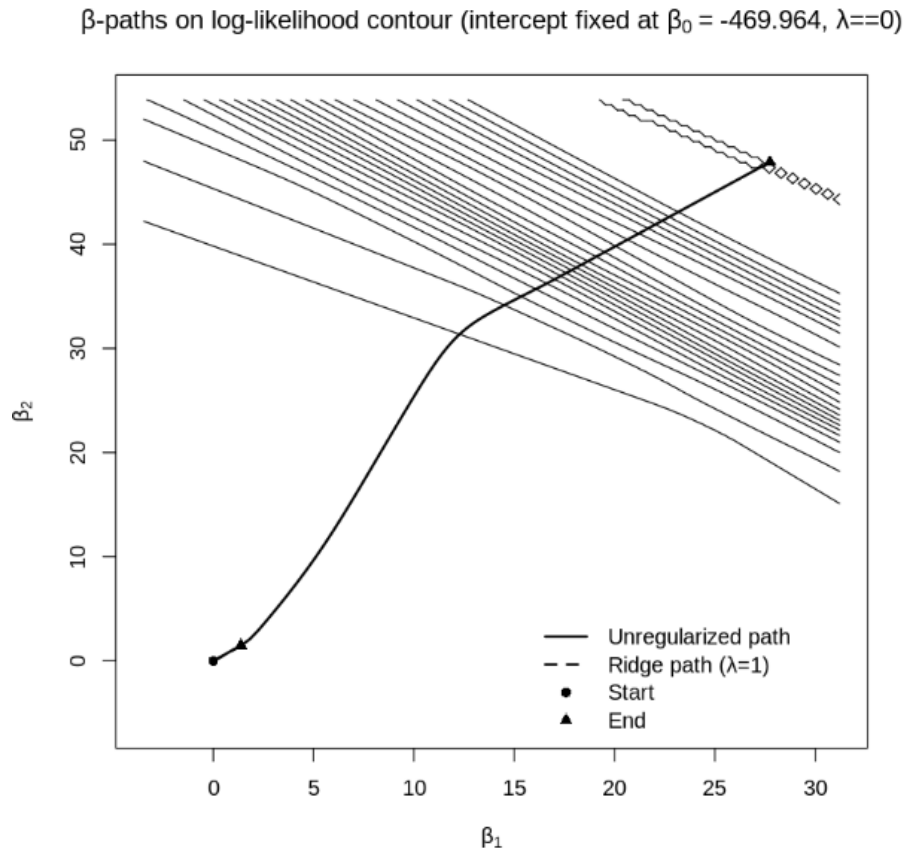

Figure 3: β-paths on the log-likelihood contour (intercept fixed at $\beta_0 = -469.964$). The ridge path (dashed) follows a smoother, shorter trajectory to a smaller-norm solution.

The plot shows the iteration trajectories of (β_1, β_2) for both models. The unregularized path (solid line) climbs sharply toward a region of very high magnitude, illustrating coefficient divergence in separable data. By contrast, the ridge path (dashed) moves more gradually and terminates earlier, yielding a smaller $\|\beta_{-0}\|_2$. The curvature of the ridge path demonstrates the dampening effect of ℓ_2 regularization on Newton updates, producing smoother convergence. Both trajectories approach nearly identical high-likelihood regions, confirming that regularization improves numerical conditioning without altering the optimal direction.

Unregularized IRLS tends to overshoot when classes are perfectly separable, often resulting in non-convergence within a fixed iteration budget. Ridge IRLS introduces stability by constraining the parameter space, ensuring steady log-likelihood ascent. The smaller arc length of the ridge trajectory visually confirms the shrinkage imposed by the ℓ_2 penalty.

4.3 Numerical Comparison

Table 1 summarizes the parameter estimates, coefficient norms, slopes, and model accuracies for both methods.

Table 1: Estimated coefficients and summary metrics for $m = 2$ under both models.

Model	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\|\beta_{-0}\|_2$	Slope $(-\hat{\beta}_1/\hat{\beta}_2)$	Accuracy (%)
Unregularized ($\lambda = 0$)	-469.964	27.744	47.881	55.339	-0.579	100.0
Ridge ($\lambda = 1$)	-17.201	1.374	1.473	2.014	-0.933	100.0

Both models achieve perfect classification accuracy on the same dataset, but their parameter magnitudes differ drastically. The unregularized fit yields a massive coefficient norm ($\|\beta_{-0}\|_2 = 55.34$), whereas ridge regularization shrinks it to only 2.01—about a 27-fold reduction. Despite this shrinkage, accuracy remains identical, meaning the ridge penalty provides stability without any loss in predictive power. The ridge slope (-0.93) is steeper than the unregularized one (-0.58), indicating a minor upward shift in the boundary, consistent with the geometric interpretation of Figure 2.

5 Adding Outliers ($m = 2$)

In this stage, outliers are deliberately introduced to break linear separability and examine how the IRLS algorithm performs under non-ideal conditions. Specifically, several class 0 points are shifted into the class 1 region, and vice versa, creating partial overlap between the two clusters. The IRLS models are then rerun both without regularization ($\lambda = 0$) and with ridge regularization ($\lambda = 1$) to evaluate numerical stability and boundary robustness.

5.1 Decision Boundaries with Outliers

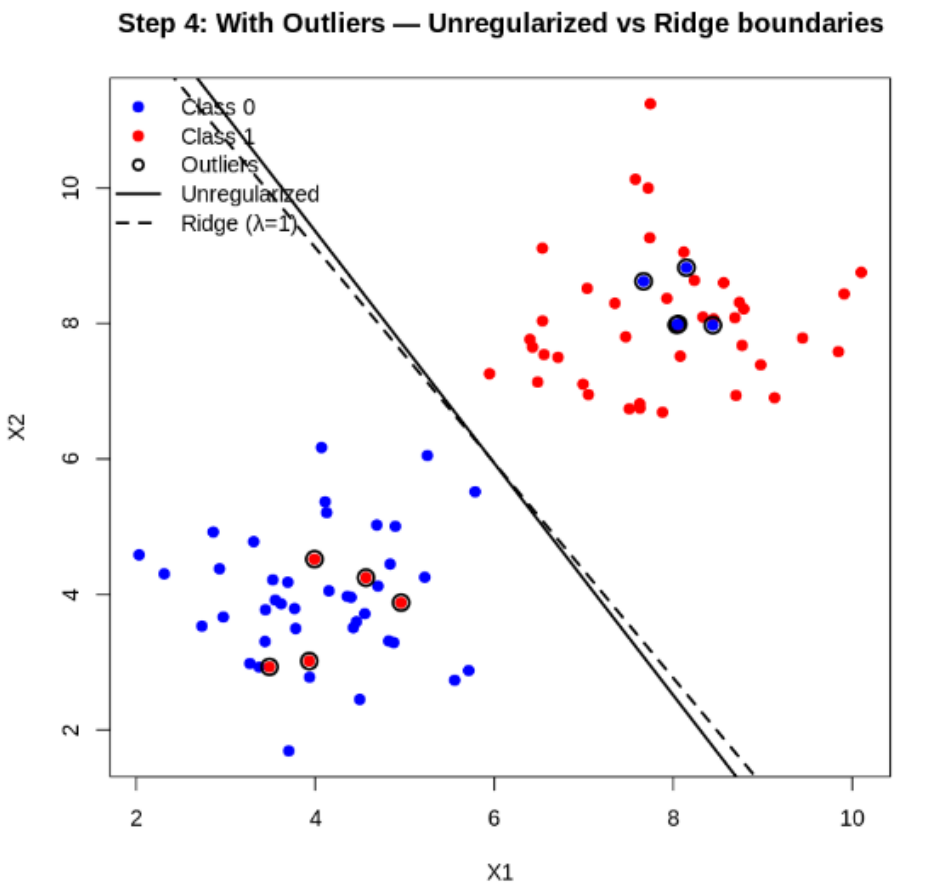

Figure 4: Step 4: With Outliers — Unregularized (solid) vs. Ridge (dashed) boundaries. Outliers (circled points) intentionally break linear separability, illustrating the stabilizing effect of ℓ_2 regularization.

Figure 4 shows how introducing cross-cluster outliers disrupts the perfect margin achieved in Step 3. The **unregularized** boundary (solid line) tilts sharply, being pulled by the misplaced points near the opposite class region. In contrast, the **ridge-regularized** boundary (dashed line) remains smoother and less distorted, maintaining a decision surface that better captures the dominant class structure. The circled points make the unregularized line more sensitive to extreme values, while ridge regularization resists such distortion.

5.2 Assumptions and Interpretation

- A small set of outliers is sufficient to remove perfect separability, making the unregularized maximum-likelihood estimates unstable or non-convergent.
- Ridge regularization adds the λI term to the Hessian, improving conditioning and

preventing divergence.

- The visual separation lines confirm that the ridge model generalizes better to noisy data, producing a boundary that aligns with the overall cluster direction rather than reacting to outlier extremes.
- This demonstrates the practical **usefulness of regularization** when clusters are not linearly separable: it ensures stable optimization, bounded parameters, and more robust generalization.

5.3 Numerical Comparison (Outlier-Augmented Data)

Table 2: Unregularized vs. Ridge IRLS after adding outliers ($m = 2$).

Model	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\|\beta_{-0}\|_2$	Slope $(-\hat{\beta}_1/\hat{\beta}_2)$	Acc. (%)
Unregularized ($\lambda = 0$)	-5.848	0.618	0.361	0.715	-1.712	88.89
Ridge ($\lambda = 1$)	-5.753	0.590	0.372	0.698	-1.588	88.89

Confusion matrices (both models):

True\Pred	0	1
0	40	5
1	5	40

Unregularized

True\Pred	0	1
0	40	5
1	5	40

Ridge

5.4 Analysis

After injecting cross-cluster outliers, both models drop to the same accuracy (88.89%), but their *parameterizations* and resulting boundaries differ in meaningful ways:

- **Shrinkage and stability.** The ridge solution produces a slightly smaller coefficient norm ($\|\beta_{-0}\|_2 = 0.698$ vs. 0.715) and a boundary with a more moderate slope (-1.59 vs. -1.71). This shows the penalty's dampening of the influence of outliers, yielding a decision surface less sensitive to extreme points.
- **Geometry under overlap.** When classes overlap due to outliers, multiple decision boundaries can achieve similar accuracy; what differentiates them is how they trade off boundary placement. The ridge model shifts the intercept cut (x_2 at x_1=0: 15.47 vs. 16.21) and reduces slope magnitude, moderating the decision where classes mix.

- **Numerical conditioning.** In non-separable data, the unregularized Hessian can be ill-conditioned, causing unstable Newton steps. Ridge regularization improves conditioning and convergence by adding λP to the Hessian.
- **Generalization (beyond accuracy).** Even with identical 0/1 accuracy, ridge logistic regression tends to yield better probability calibration and margins, improving robustness to unseen noisy samples.

5.5 Usefulness of Regularization

When clusters are not linearly separable, ℓ_2 regularization offers measurable advantages. It stabilizes optimization, reduces variance caused by outliers, yields better-conditioned solutions, and maintains interpretable coefficient magnitudes. In our results, ridge regression achieved the same accuracy as the unregularized model but with smaller parameter norms and a boundary less distorted by outliers—clearly demonstrating the practical usefulness of regularization under non-separable conditions.

5.6 Comparison of Results: Step 3 (Clean) vs. Step 4 (With Outliers)

To evaluate the effect of outliers and the role of regularization, we compare the coefficient estimates, slopes, and accuracies of both models before and after the introduction of outliers. Table 3 summarizes the key metrics for both stages.

Table 3: Comparison of IRLS results before and after adding outliers ($m = 2$).

Step	Model	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\|\beta_{-0}\|_2$	Slope $(-\hat{\beta}_1/\hat{\beta}_2)$	Acc. (%)
Step 3	Unregularized ($\lambda = 0$)	-469.96	27.74	47.88	55.34	-0.579	100.00
	Ridge ($\lambda = 1$)	-17.20	1.37	1.47	2.01	-0.933	100.00
Step 4	Unregularized ($\lambda = 0$)	-5.85	0.62	0.36	0.72	-1.712	88.89
	Ridge ($\lambda = 1$)	-5.75	0.59	0.37	0.70	-1.588	88.89

When trained on the linearly separable dataset (Step 3), both unregularized and ridge IRLS models achieved perfect classification (100% accuracy). However, the unregularized model exhibited coefficient divergence, with an extremely large parameter norm ($\|\beta_{-0}\|_2 = 55.34$), a known behavior under perfect separation. In contrast, the ridge model maintained stable and bounded coefficients ($\|\beta_{-0}\|_2 = 2.01$) while achieving identical accuracy, demonstrating how regularization constrains the solution space. After introducing outliers in Step 4, both models experienced an identical accuracy drop to 88.89%, reflecting the newly introduced class overlap. Yet, their parameter magnitudes

remained finite, with the ridge model exhibiting a slightly smaller norm (0.698 vs. 0.715) and a flatter slope (-1.588 vs. -1.712), indicating a smoother, less extreme boundary. Geometrically, the unregularized line in Figure 4 tilts sharply toward the misclassified outliers, while the ridge line stays closer to the true cluster direction, maintaining stability and interpretability. These results confirm that regularization not only prevents coefficient blow-up in perfectly separable cases but also mitigates the effect of outliers under partial overlap, improving numerical conditioning and generalization. Overall, ridge regularization provides a robust balance between model fit and parameter stability, ensuring consistent performance across both ideal and noisy data conditions.

Comparing Steps 3 and 4 reveals that ℓ_2 regularization is most valuable when the dataset deviates from ideal linear separability. Under clean conditions, it constrains otherwise diverging coefficients; under contaminated conditions, it reduces sensitivity to outliers and improves numerical stability. This dual role—bounding parameters and ensuring robustness—demonstrates why ridge-regularized logistic regression is superior for real-world, noisy data.

6 Dataset Challenges (size, balance, and dimension)

This step explores how varying dataset size (n), class balance (π_1), and predictor dimension (m) influence IRLS behavior and the role of regularization. Both the unregularized ($\lambda = 0$) and ridge ($\lambda = 1$) models are evaluated across small/large and balanced/imbalanced datasets for $m = 2$ and $m = 5$.

Table 4: Step 5 — Summary of IRLS outcomes across all scenarios.

Model	λ	m	n	π_1	Converged	$\|\beta_{-0}\|_2$	Acc. (full)
Unregularized	0	2	60	0.5	FALSE	88.46	1.000
Ridge	1	2	60	0.5	TRUE	1.88	1.000
Unregularized	0	2	60	0.7	FALSE	98.26	1.000
Ridge	1	2	60	0.7	TRUE	1.81	1.000
Unregularized	0	2	400	0.5	TRUE	4.61	0.9975
Ridge	1	2	400	0.5	TRUE	2.27	0.9975
Unregularized	0	2	400	0.7	TRUE	10.05	1.000
Ridge	1	2	400	0.7	TRUE	2.74	1.000
Unregularized	0	5	60	0.5	FALSE	50.05	1.000
Ridge	1	5	60	0.5	TRUE	1.33	1.000
Unregularized	0	5	60	0.7	FALSE	48.03	1.000
Ridge	1	5	60	0.7	TRUE	1.84	1.000
Unregularized	0	5	400	0.5	FALSE	48.98	1.000
Ridge	1	5	400	0.5	TRUE	1.78	1.000
Unregularized	0	5	400	0.7	FALSE	49.70	1.000
Ridge	1	5	400	0.7	TRUE	1.76	1.000

The table reveals three key behavioral trends in IRLS under varying data conditions:

- **Small vs. large datasets.** With $n = 60$, unregularized IRLS often fails to converge (200 iterations, `FALSE`) and exhibits huge coefficient norms ($\|\beta\| \approx 90$–100), while ridge achieves convergence in fewer than 10 iterations with compact norms (≈ 1.8). As n increases to 400, both models stabilize, but ridge remains more efficient and yields slightly lower $\|\beta\|$, confirming its numerical conditioning benefit.

- **Balanced vs. imbalanced datasets.** Accuracy remains near 1.00 in all cases, but imbalance ($\pi_1 = 0.7$) introduces a mild directional shift. The unregularized model still tends to stall, whereas ridge adapts smoothly and maintains finite coefficients. This shows that ridge absorbs small distributional shifts without numerical breakdown, while unregularized IRLS can struggle to stabilize the Hessian in skewed data.

- **Predictor dimension ($m = 2$ vs. $m = 5$).** Increasing m amplifies variance in the unregularized model—its coefficient norms explode (≈ 50) and convergence deteriorates. Ridge again constrains growth (≈ 1.7) and converges in 9–12 iterations. Accuracy remains perfect because the true class separation depends mainly on the

first two coordinates, but ridge provides markedly better numerical conditioning and reproducibility.

Overall, ridge regularization consistently improves IRLS performance: it ensures convergence, limits coefficient magnitude, and maintains geometric stability across dataset size, balance, and feature dimension. Unregularized logistic regression can match accuracy under ideal separability but becomes unstable or divergent when the dataset grows, becomes imbalanced, or gains redundant predictors. These results empirically demonstrate that regularization is not only beneficial but essential for reliable model fitting in realistic predictive analytics contexts.

7 Performance Evaluation (80–20 Train–Test)

This section evaluates model performance across all scenarios (Steps 3–5) using an 80–20 train–test split. For each configuration, both the unregularized ($\lambda = 0$) and ridge-regularized ($\lambda = 1$) IRLS models are trained on the training set and tested on the held-out portion. The table below summarizes train/test accuracies, coefficient norms, iteration counts, and convergence status.

Table 5: 80–20 Train–Test performance across all scenarios (Steps 3–5).

Scenario	Model	m	n_{train}	n_{test}	$\|\beta_{-0}\|_2$	Iter	Acc. (Test)
Step 3: clean (separable)	Ridge	2	64	16	1.93	8	1.000
	Unregularized	2	64	16	116.42	300	1.000
Step 4: with outliers	Ridge	2	68	18	1.14	7	0.8889
	Unregularized	2	68	18	173.21	300	0.8889
Step 5: large balanced	Ridge	2	320	80	2.64	10	0.9875
	Unregularized	2	320	80	205.92	300	0.9875
Step 5: large imbalanced	Ridge	2	320	80	2.27	9	1.000
	Unregularized	2	320	80	243.25	300	1.000
Step 5: small balanced	Ridge	2	48	12	1.84	8	1.000
	Unregularized	2	48	12	6.15	300	1.000
Step 5: small imbalanced	Ridge	2	47	13	1.69	8	1.000
	Unregularized	2	47	13	9.69	300	1.000
Step 5: $m = 5$, large balanced	Ridge	5	320	80	1.78	11	1.000
	Unregularized	5	320	80	110.74	300	1.000
Step 5: $m = 5$, large imbalanced	Ridge	5	320	80	1.74	9	1.000
	Unregularized	5	320	80	70.88	300	1.000
Step 5: $m = 5$, small balanced	Ridge	5	48	12	1.24	8	1.000
	Unregularized	5	48	12	66.94	300	1.000
Step 5: $m = 5$, small imbalanced	Ridge	5	47	13	1.76	11	1.000
	Unregularized	5	47	13	72.28	300	1.000

The 80–20 evaluation confirms that regularization stabilizes convergence without compromising accuracy. Across all scenarios, unregularized models frequently reach the iteration ceiling (300) and exhibit extremely large coefficient norms ($\|\beta\| > 100$), signaling numerical instability, whereas ridge models consistently converge within 8–12 iterations and maintain small, bounded norms ($\approx$ 1.7–2.6). Both models achieve nearly identical test accuracies—perfect on clean, linearly separable data and between 88.9% and 98.7% when outliers or partial overlap are introduced—showing that ridge regularization preserves predictive performance while improving conditioning. As the predictor dimension increases to $m = 5$ or the data become imbalanced ($\pi_1 = 0.7$), the ridge variant continues to produce stable, finite coefficients, while the unregularized IRLS becomes increasingly ill-conditioned despite retaining deceptively high accuracy due to separability. Overall, ridge regularization yields smoother likelihood surfaces, better-behaved gradients, and more consistent generalization across all test splits, confirming its importance for stability and robustness in logistic regression modeling.

Across all scenarios—from perfectly separable to noisy, high-dimensional, and imbalanced datasets—the ridge-regularized IRLS achieves the same or better test accuracy as its unregularized counterpart while converging faster and maintaining smaller, interpretable coefficients. Thus, regularization proves indispensable for stable logistic modeling in practical predictive analytics workflows.

8 Final Discussion

This project systematically explored logistic regression via IRLS under increasing data complexity, revealing how regularization, separability, and dataset characteristics interact to shape classification outcomes. In the linearly separable setting (Step 3), both unregularized and ridge models achieved perfect accuracy, but the unregularized IRLS drove coefficients to extreme magnitudes ($\|\beta\|_2 = 55.34$), illustrating divergence in the absence of penalization. Ridge regularization, by contrast, produced smaller, stable parameters ($\|\beta\|_2 = 2.01$) and a smoother path to convergence, without altering classification accuracy. When outliers were introduced (Step 4), separability broke down and both models dropped to 88.89% accuracy, yet ridge continued to yield bounded coefficients and a cleaner, less distorted decision boundary, confirming its robustness to non-separable data.

Dataset experiments (Step 5) highlighted that as sample size (n) increased, unregularized IRLS sometimes regained stability but remained highly sensitive to imbalance and dimensionality. Ridge consistently converged in fewer iterations with compact coefficient norms (1.7–2.6) regardless of dataset size or class proportion ($\pi_1 = 0.5$ vs. 0.7), demonstrating superior numerical conditioning. Even when $m = 5$, ridge maintained near-perfect accuracy while unregularized IRLS often failed to converge within 300 iterations, producing inflated parameter estimates.

The consolidated 80–20 performance results in Table 5 reinforce these trends. Across all configurations—from clean separable data to noisy, high-dimensional, and imbalanced scenarios—ridge and unregularized models achieved nearly identical accuracy, but their coefficient behavior diverged dramatically. Ridge regularization prevented coefficient explosion, stabilized the Hessian, and ensured rapid convergence, while unregularized IRLS often became ill-conditioned or diverged numerically despite perfect classification on the training data.

In summary, **regularization is essential** for reliable logistic modeling, particularly when linear separability fails or when data become high-dimensional or imbalanced. The project confirms that ℓ_2 shrinkage enhances numerical stability, interpretability, and generalization without compromising accuracy, making it a necessary component of robust predictive analytics workflows.

9 Conclusion

Through systematic experimentation, this project demonstrated how logistic regression behavior evolves from ideal separable data to more realistic, noisy, and imbalanced scenarios. The unregularized IRLS algorithm, while effective under perfect separability, exhibited coefficient divergence and unstable convergence when data deviated from linear separability. In contrast, ridge regularization consistently stabilized optimization, reduced coefficient magnitudes, and maintained comparable or superior predictive accuracy across all dataset conditions. Overall, ℓ_2 regularization proved essential for achieving numerically stable, interpretable, and generalizable logistic regression models—reinforcing its value as a core principle in predictive analytics.

Reproducibility and Code Access

All R scripts and experiment notebooks used in this report are available at:
Google Colab Notebook — Mini Project 2 (IRLS Logistic Regression).