Maternal Health Risk Level Classification with Softmax Logistic Regression (From Scratch)

Data Science Final Project Report

Kritim Bastola
University of New Mexico

December 14, 2025

Abstract

This project predicts maternal health risk level (high, mid, or low) from six clinical indicators (Age, SystolicBP, DiastolicBP, Blood Sugar, Body Temperature, and Heart Rate) using multinomial logistic regression (softmax regression) implemented from scratch with batch gradient descent. The model is trained on 1014 patient records from the UCI Machine Learning Repository. After standardizing features and tuning learning rate and L2 regularization, the best configuration achieved **66.3%** test accuracy and a macro F1 score of **0.641**. Performance is strongest for "low risk" and "high risk", while "mid risk" is frequently confused with neighboring classes. This report presents the model derivation, optimization procedure, and a critical discussion of limitations and next steps.

1 Introduction

Maternal health monitoring is a public health priority because delayed identification of high-risk pregnancies can lead to preventable complications. In many settings, risk assessment is performed from a small set of vital signs and lab measurements. The goal of this project is to build a simple, transparent baseline classifier that maps routinely collected measurements to a categorical risk label: *high risk*, *mid risk*, or *low risk*.

To stay aligned with course material, the model is multinomial logistic regression (softmax regression) trained with batch gradient descent. This approach provides a strong baseline: it is fast, interpretable , and produces calibrated class probabilities.

2 Dataset

The dataset contains 1014 records and 6 numeric features with a 3-class categorical label (`RiskLevel`). The class distribution is moderately imbalanced (low risk is the most common class). Data was sourced from the UCI Machine Learning Repository (Maternal Health Risk dataset, DOI: 10.24432/C5DP5D). A few measurements include extreme or potentially erroneous values (e.g., unusually low heart rate); no outlier removal was applied to keep the baseline pipeline simple and reproducible.

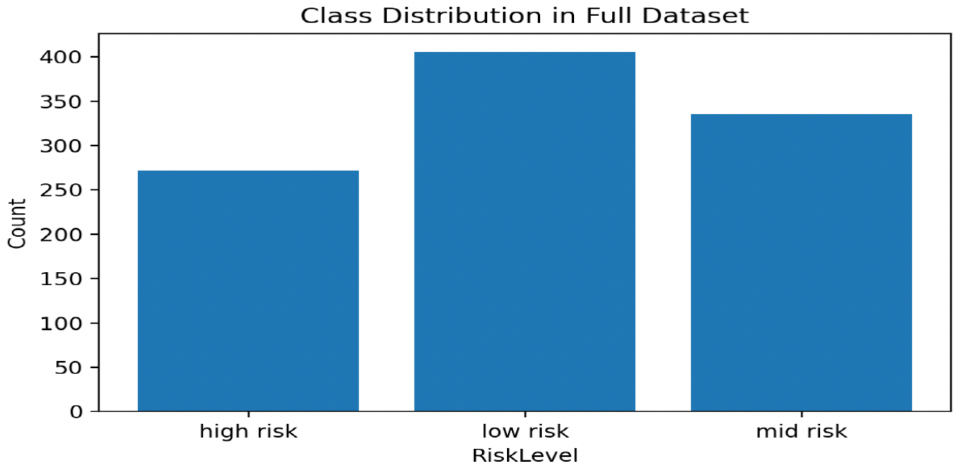

Figure 1: Class distribution in the full dataset.

2.1 Feature summary

Table 1: Summary statistics of numeric features.

Feature	Description	Mean	Std	Min	Max
Age	Age in years	29.87	13.47	10.00	70.00
SystolicBP	Upper blood pressure (mmHg)	113.20	18.40	70.00	160.00
DiastolicBP	Lower blood pressure (mmHg)	76.46	13.89	49.00	100.00
BS	Blood sugar (mmol/L)	8.73	3.29	6.00	19.00
BodyTemp	Body temperature (F)	98.67	1.37	98.00	103.00
HeartRate	Heart rate (bpm)	74.30	8.09	7.00	90.00

3 Methodology

This section summarizes the model, objective function, and optimization procedure used to train multinomial logistic regression. The implementation follows the course notes: compute linear class scores, convert them to probabilities with softmax, and minimize negative log-likelihood (cross-entropy) using gradient descent.

3.1 Model

Let $x \in \mathbb{R}^d$ be a feature vector ($d = 6$) and let there be $K = 3$ classes. Softmax regression models class scores as a linear function:

$$z = W^\top x + b, \quad W \in \mathbb{R}^{d\times K},\ b \in \mathbb{R}^K.$$

These scores are converted to probabilities using softmax:

$$p(y = k \mid x) = \frac{\exp(z_k)}{\sum_{j=1}^{K} \exp(z_j)}.$$

For numerical stability, the implementation subtracts $\max(z)$ from each row of logits before exponentiation.

3.2 Loss function

Given n training examples, with one-hot labels $y^{(i)} \in \{0,1\}^K$ and predicted class probabilities $p^{(i)} \in [0,1]^K$, the average negative log-likelihood (cross-entropy) is:

$$L = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} y_k^{(i)} \log p_k^{(i)}.$$

To reduce overfitting, L2 regularization is added:

$$L_{\text{reg}} = L + \frac{\lambda}{2}\|W\|_F^2,$$

where $\lambda \geq 0$ controls the strength of shrinkage.

3.3 Optimization

With a design matrix $X \in \mathbb{R}^{n\times d}$, probability matrix $P \in \mathbb{R}^{n\times K}$, and one-hot label matrix $Y \in \mathbb{R}^{n\times K}$, the vectorized gradients are:

$$\frac{\partial L}{\partial W} = \frac{1}{n}X^\top(P-Y) + \lambda W, \qquad \frac{\partial L}{\partial b} = \frac{1}{n}\mathbf{1}^\top(P-Y),$$

where $\mathbf{1} \in \mathbb{R}^n$ is an all-ones vector. Parameters are updated by batch gradient descent:

$$W \leftarrow W - \eta\frac{\partial L}{\partial W}, \qquad b \leftarrow b - \eta\frac{\partial L}{\partial b},$$

where η is the learning rate.

3.4 Experimental setup

Data was split using stratified sampling into 80% training and 20% test. This resulted in 812 training examples and 202 test examples. The training portion contained 325 low-risk, 269 mid-risk, and 218 high-risk instances; the test portion contained 81, 67, and 54 instances respectively.

All features were standardized using the training mean and standard deviation. Model selection explored learning rates $\eta \in \{0.5, 0.2, 0.1, 0.05, 0.02, 0.01\}$ and L2 regularization $\lambda \in \{0, 0.001, 0.01, 0.05, 0.1, 0.2\}$. The best performance in this sweep occurred with $\eta = 0.1$ and $\lambda = 0.2$, trained for 4000 iterations.

4 Results

This section reports predictive performance on the held-out test set ($n = 202$). Overall accuracy is **66.3%**. The macro-averaged F1 score is **0.641**, which accounts for class imbalance by averaging per-class F1.

Table 2: Precision/Recall/F1 on the test set.

Class	Precision	Recall	F1	Support
high risk	0.750	0.778	0.764	54
low risk	0.642	0.864	0.737	81
mid risk	0.595	0.328	0.423	67
Macro avg	0.662	0.657	0.641	202

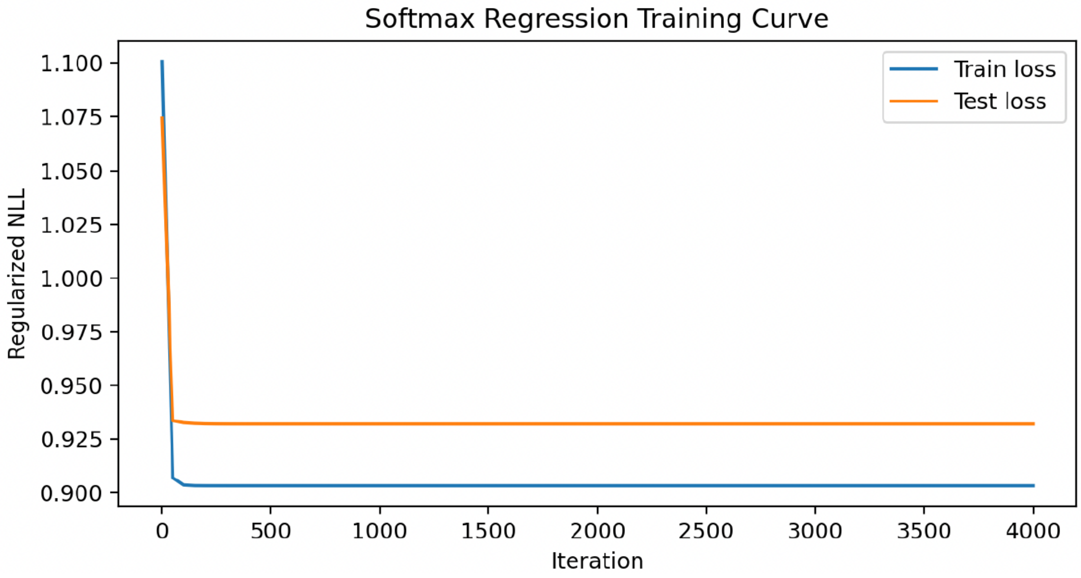

Figure 2: Training and test loss over iterations (regularized NLL).

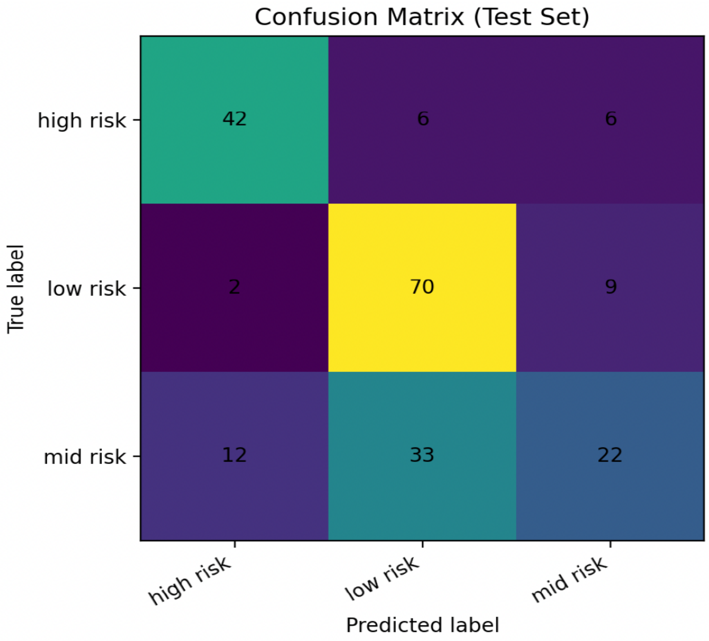

Figure 3: Confusion matrix on the test set.

For reference, the confusion matrix counts (true rows by predicted columns) are:

$$\begin{bmatrix} 42 & 6 & 6 \\ 2 & 70 & 9 \\ 12 & 33 & 22 \end{bmatrix}.$$

5 Critical Discussion

Softmax regression provides a simple baseline, but results show clear limitations. The model performs reasonably for *low risk* (recall 0.864) and *high risk* (recall 0.778), yet struggles with *mid risk* (recall 0.328). The confusion matrix suggests that many mid-risk cases are predicted as low-risk or high-risk, implying overlap in the feature space between neighboring categories.

Regularization helped. Without L2 ($\lambda = 0$), test accuracy was approximately 60%, while $\lambda = 0.2$ increased test accuracy to 66.3%. This indicates that a small dataset with correlated clinical features benefits from shrinking coefficients to reduce variance.

However, even with regularization, the decision boundaries remain linear. If the true relationship between features and risk is nonlinear (e.g., thresholds or interactions between blood pressure and blood sugar), a linear model will misclassify borderline cases. Potential improvements include: (1) adding polynomial or interaction features, (2) class-weighted loss (or focal loss) to emphasize mid-risk, (3) more robust preprocessing (outlier handling), and (4) nonlinear models such as tree ensembles or neural networks, evaluated with careful cross-validation.

Because this dataset originates from real-world measurement settings, data quality matters. The presence of extreme values (e.g., very low heart rate) may introduce noise that disproportionately affects a linear classifier. In a production setting, domain-informed validation rules could remove implausible readings or mark them as missing.

6 Conclusion

A multinomial logistic regression classifier was implemented from scratch using softmax, cross-entropy loss, and gradient descent. On the Maternal Health Risk dataset (1014 records), the tuned model achieved 66.3% test accuracy and macro F1 of 0.641. The model is most reliable at identifying low-risk and high-risk patients, while mid-risk remains challenging due to overlap and the constraints of a linear decision boundary. This makes softmax regression a useful, interpretable baseline and a reference point for more expressive models.

References

1. Ahmed, M. (2020). *Maternal Health Risk* [Dataset]. UCI Machine Learning Repository. DOI: 10.24432/C5DP5D.

2. Course notes (Week 13): Logistic Regression and Softmax Regression, provided by instructor (used for model derivation and gradient updates).