# STA304XS – Generalized Linear Models

Assignment 3 2025

October 1, 2025

## General Notes and Instructions

- Where questions require algebra, you may provide hand-written answers provided that they are properly labelled though it is suggested that you typeset answers properly — Report writing is a skill you should develop as soon as possible. You may use any typesetting software you wish, but I would encourage you to use R-Markdown or LaTeX if you have sufficient experience. See the primers on R-Markdown on Amathuba and make use of the cheat-sheets for reference.

- Unless stated otherwise, where you are required to write R/Python-Code, DO NOT give blocks of code interspersed between your answers (you may still refer to R-functions in-text). Instead, provide the complete code for each of the relevant questions under separate headings as an appendix to your write-up. For example, for Question 2, provide all code pertaining to this question as a single script in the appendix under the subsection heading 'R/Python-Code: Question 2'. When you typeset R/Python Code use courier or an equivalent 'typewriter'-like font.

- Plots are very useful but use them sparingly! Make sure that a given plot is relevant to the question and pertains to text in your answer. Figures are meant to enrich your analysis, not leave it to the reader to analyse. Make sure that your plots maintain a 1-1 aspect ratio (square) and have appropriate legends. This can be achieved in RMarkdown/Jupyter Notebook by setting the appropriate code-chunk options.

- You may write as much R/Python Code as you like, but the body of your text (i.e., excluding the appendix with your R/Python Code) may not exceed 10 pages.

- All assignments must be accompanied by a signed plagiarism declaration. A template is provided on Amathuba. No plagiarism declaration no marks.

- Hand-in dates will be announced on Amathuba.

- Electronic hand-ins MUST BE IN PDF FORMAT.

# Overview

In this assignment, you will explore the connection between **Ridge Logistic Regression** and **Bayesian MAP estimation** with Gaussian priors. You will derive and implement a modified **IWLS algorithm** for ridge-penalised logistic regression, and compare your solution to standard implementations (e.g. `glmnet`, `sklearn`). This assignment blends theory, algorithmic derivation, and computational practice.

# Dataset

Suppose there is a disease spreading rapidly across the world and that a laboratory responsible for recording nasal swab results captures the outcome of a test for the disease as well as some physiological measurements taken during the test. Download the dataset Assignment_Disease.txt from Amathuba and read the dataset into R or Python. The dataset contains 202 observations on the following variables:

| Variable | Description |
|---|---|
| DiseaseStatus | Disease indicator: 0 if negative, 1 if positive. |
| Measurement1 | First measurement recorded. |
| Measurement2 | Second measurement recorded. |
| Measurement3 | Third measurement recorded. |

For purposes of determining whether a particular person has the disease or not you are tasked with predicting whether an entry is either positive or negative (DiseaseStatus) using: Measurement1, Measurement2 and Measurement3.

**Note**: This is just an expository dataset.

**Question 1** (7 marks)

**BAYESIAN INTERPRETATION**

(a) (MAP Estimation and Penalisation) (5)

- Starting with the logistic model:

$$\Pr(Y_i = 1 \mid \mathbf{x}_i) = \text{logit}^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta})$$

- Assume a prior:

$$\beta_j \sim \mathcal{N}(0, \tau^2), \text{ independently}$$

- Derive the MAP estimator:

$$\log p(\boldsymbol{\beta} \mid \mathbf{y}) \propto \ell(\boldsymbol{\beta}) - \frac{1}{2\tau^2} \sum_{j=1}^{p} \beta_j^2$$

- Show this corresponds to:

$$\text{minimise } -\ell(\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_2^2, \quad \text{where } \lambda = \frac{1}{2\tau^2}$$

(b) (Interpretation) (2)

- Briefly interpret the role of $\lambda$ or $\tau^2$.
- What does this form of regularisation achieve?
- How is this Bayesian view useful compared to MLE?

**Question 2** (7 marks)

**DERIVING RIDGE-IWLS**

(a) (Standard IWLS Recap) (2)

Derive the standard IWLS update for logistic regression:

$$\boldsymbol{\beta}^{(t+1)} = \left(\mathbf{X}^\top \mathbf{W}^{(t)} \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{W}^{(t)} \mathbf{z}^{(t)}$$

where:

$$\mathbf{W}^{(t)} = \operatorname{diag}(p_i^{(t)}(1 - p_i^{(t)})), \quad \mathbf{z}^{(t)} = \mathbf{X}\boldsymbol{\beta}^{(t)} + (\mathbf{W}^{(t)})^{-1}\left(\mathbf{y} - \mathbf{p}^{(t)}\right)$$

(b) (Ridge-IWLS Derivation) (5)

- Starting from:

$$\text{minimise } -\ell(\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_2^2$$

- Derive the modified IWLS update formula:

$$\boldsymbol{\beta}^{(t+1)} = \left(\mathbf{X}^\top \mathbf{W}^{(t)} \mathbf{X} + 2\lambda \mathbf{I}\right)^{-1} \mathbf{X}^\top \mathbf{W}^{(t)} \mathbf{z}^{(t)}$$

- Carefully show and explain each step (include matrix calculus as appropriate).

**Question 3** (8 marks)

**IMPLEMENTATION AND EVALUATION**

(a) (Code Ridge-IWLS) (5)

- Implement your derived Ridge-IWLS algorithm in R or Python.
- Allow for varying $\lambda$ values.
- Use convergence criteria (e.g. change in $\boldsymbol{\beta}$ or log-likelihood).

```
glm_fit = function(y, X, ...) {
    # Prepare elements for the updating procedure here:
    # Evaluate the updating equation:
    for(i in 2:k) {
    }
    # Process some of the results here:
    # Return relevant content:
}
```

Marks will be allocated to reflect the quality of your function.

(b) (Compare with standard software) (2)

- Fit the same model using:
  - `glmnet` (R) with `alpha = 0`

    - or `LogisticRegression(penalty='l2')` in `scikit-learn`
  - Compare coefficient estimates, predictive performance, and runtime.

(c) (Reflect) (1)

  - Discuss differences (if any) between your implementation and the package output.
  - Comment on algorithmic stability or limitations of IWLS.

**Question 4** (3 marks)
**BONUS: VISUALISATION AND ANALYSIS**

- Plot coefficient shrinkage as $\lambda$ increases.
- Plot training vs. validation accuracy as a function of $\lambda$.
- Simulate overfitting and show how ridge improves generalisation.

# Submission Checklist

- [ ] PDF report with derivations, discussion, and plots.
- [ ] Code in a Jupyter notebook or R script.
- [ ] Results of software comparison.
- [ ] Clear indication of $\lambda$ values used and convergence diagnostics.