

**Problem 1.** Logistic Regression, FFNN

The file Default.csv contains data regarding loan defaults:

default: individual's default status labeled as "Yes" or "No"

balance: loan balance              income: individual's income

You decided to use the data to fit a logistic regression model.

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 \times \text{loan.balance} + \beta_2 \times \text{income} \quad (1)$$

A. Use your preferred package to fit the logistic regression model and report the following:

a1. The MLE estimates of  $\beta_0, \beta_1, \beta_2$

a2. The maximum likelihood achieved at the MLE estimates

B. Use a simple neural network with no hidden layers to estimate the default probability  $\pi$  to classify default, involving the sigmoid activation function

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{1 + e^x}$$

and the binary cross-entropy loss function

$$R = -\frac{1}{n} \sum_{i=1}^n (1 - y_i) \ln(1 - \hat{y}_i) + y_i \ln \hat{y}_i$$

Note that by adopting the sigmoid activation function, your neural network output  $\hat{y}_i$  is just the estimate for the default probability  $\hat{\pi}_i$

Please write your own code to implement a full Gradient Descent algorithm to fit the neural network.

To facilitate convergence, you should standardize features (loan balance and income). For example, with standardized features and a learning rate of 0.025, you can get to within  $1 \times 10^{-8}$  the likelihood reported in a2.

Please start your fit with a initial weight of  $(0.0, 0.0, 0.0)^T$

Let your code report

b1. the smallest binary cross-entropy your algorithm has achieved over 200 updates (epochs)

b2. the associated weights ( $\beta_j$ s fitted with standardized features)

b3. the corresponding  $\beta_j$ s for the model in equation (1) fitted with non-standardized features.