STAT/BIOSTAT 534 Statistical Computing
Final Project (UPDATED WITH NEW TERMINOLOGY)

Adrian Dobra
adobra@uw.edu

1 Rules

This is your final examination for this course. The work you will submit needs to reflect your own understanding of the material covered in class. It is fine to talk with each other about the project. However, when you actually do the work, you should do it alone. You are free to use any code you wrote for other assignments as well as any code I shared with you. Other than that, the code you will share with me should be entirely your own.

The background material you need to use is presented in Section 2. The full description of the project is given in Section 3. Successfully implementing the algorithm described in Section 3 will give you 300 points towards your final grade. Note: you do not need any knowledge of Bayesian statistics to complete this assignment.

2 Bayesian Inference in Univariate Logistic Regression

We assume to have observed the n independent samples

$$\mathcal{D} = \{(y_1, x_1), \ldots, (y_n, x_n)\}.$$

The response variable Y is binary, i.e. $y_i \in \{0, 1\}$ for $i = 1, 2, \ldots, n$. The explanatory variable X can be continuous or discrete. We consider the univariate logistic regression model

$$\log \frac{P(y = 1|x)}{P(y = 0|x)} = \beta_0 + \beta_1 x. \tag{1}$$

Our model assumptions say that each y_i follows a Bernoulli distribution with probability of success $\pi_i = P(y_i = 1|x_i)$:

$$y_i \sim Ber(\pi_i).$$

Since the samples are assumed to be independent, the likelihood is:

$$L(\beta_0, \beta_1 | \mathcal{D}) = \prod_{i=1}^{n} [P(y_i = 1|x_i)]^{y_i} [1 - P(y_i = 1|x_i)]^{1-y_i},$$

where

$$\pi_i = P(y_i = 1|x_i) = logit^{-1}(\beta_0 + \beta_1 x_i) \in (0,1).$$

We define the *logit* function:

$$logit : (0,1) \longrightarrow (-\infty, +\infty), \quad logit(p) = \log \frac{p}{1-p}.$$

Its inverse is:

$$logit^{-1} : (-\infty, +\infty) \longrightarrow (0,1), \quad logit^{-1}(x) = \frac{\exp(x)}{1 + \exp(x)}.$$

The log-likelihood is

$$\begin{aligned} l(\beta_0, \beta_1 | \mathcal{D}) &= \log L(\beta_0, \beta_1 | \mathcal{D}), \\ &= \sum_{i=1}^{n} (y_i \log \pi_i + (1 - y_i) \log[1 - \pi_i]). \end{aligned}$$

Simple calculations show that

$$\begin{aligned} \frac{\partial l(\beta_0, \beta_1 | \mathcal{D})}{\partial \beta_0} &= \sum_{i=1}^{n} [y_i - \pi_i], \\ \frac{\partial l(\beta_0, \beta_1 | \mathcal{D})}{\partial \beta_1} &= \sum_{i=1}^{n} [y_i x_i - \pi_i x_i]. \end{aligned}$$

We assume that the logistic regression coefficients follow independent $N(0,1)$ priors. The joint posterior distribution of β_0 and β_1 is therefore given by

$$P(\beta_0, \beta_1 | \mathcal{D}) = \frac{1}{P(\mathcal{D})} \exp\left(l^*(\beta_0, \beta_1)\right), \tag{2}$$

where

$$l^*(\beta_0, \beta_1) = -\log(2\pi) - \frac{1}{2}\left(\beta_0^2 + \beta_1^2\right) + l\left(\beta_0, \beta_1 | \mathcal{D}\right),$$

and

$$P(\mathcal{D}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(l^*(\beta_0, \beta_1)\right) d\beta_0 d\beta_1. \tag{3}$$

We call $P(\mathcal{D})$ the marginal likelihood associated with the univariate logistic regression (1).

The gradient of $l^*(\beta_0, \beta_1)$ is

$$\nabla l^*(\beta_0, \beta_1) = \begin{pmatrix} \frac{\partial l^*(\beta_0, \beta_1)}{\partial \beta_0} \\ \frac{\partial l^*(\beta_0, \beta_1)}{\partial \beta_1} \end{pmatrix}.$$

The Hessian matrix associated with $l^*(\beta_0,\beta_1)$ is

$$D^2 l^*(\beta_0,\beta_1) = \begin{bmatrix} \frac{\partial^2 l^*(\beta_0,\beta_1)}{\partial\beta_0^2} & \frac{\partial^2 l^*(\beta_0,\beta_1)}{\partial\beta_0\partial\beta_1} \\ \frac{\partial^2 l^*(\beta_0,\beta_1)}{\partial\beta_1\partial\beta_0} & \frac{\partial^2 l^*(\beta_0,\beta_1)}{\partial\beta_1^2} \end{bmatrix}.$$

2.1 The Newton-Raphson Algorithm

We determine the mode of the posterior distribution (2), i.e.

$$\left(\widehat{\beta}_0,\widehat{\beta}_1\right) = \mathrm{argmax}_{(\beta_0,\beta_1)\in\Re^2} l^*(\beta_0,\beta_1),$$

by employing the Newton-Raphson algorithm. The procedure starts with the initial values $\left(\beta_0^{(0)},\beta_1^{(0)}\right) = (0,0)$. At iteration k, we update our current estimate $\left(\beta_0^{(k-1)},\beta_1^{(k-1)}\right)$ of the mode $\left(\widehat{\beta}_0,\widehat{\beta}_1\right)$ to a new estimate $\left(\beta_0^{(k)},\beta_1^{(k)}\right)$ as follows:

$$\begin{pmatrix} \beta_0^{(k)} \\ \beta_1^{(k)} \end{pmatrix} = \begin{pmatrix} \beta_0^{(k-1)} \\ \beta_1^{(k-1)} \end{pmatrix} - \left[D^2 l^*(\beta_0^{(k-1)},\beta_1^{(k-1)})\right]^{-1} \nabla l^*(\beta_0^{(k-1)},\beta_1^{(k-1)}).$$

The procedure stops when the estimates of the mode do not change after performing a new update, i.e. $|\beta_0^{(k)} - \beta_0^{(k-1)}| < \epsilon$ and $|\beta_1^{(k)} - \beta_1^{(k-1)}| < \epsilon$. Here ϵ is some small positive number, e.g. 0.0001.

2.2 The Laplace Approximation

Since the integral (3) cannot be explicitly calculated, we need to approximate it numerically. We calculate the marginal likelihood $P(\mathcal{D})$ using the Laplace approximation, i.e.

$$\widehat{P(\mathcal{D})} = 2\pi \exp\left(l^*(\widehat{\beta}_0,\widehat{\beta}_1)\right)\left\{\det\left[-D^2 l^*(\widehat{\beta}_0,\widehat{\beta}_1)\right]\right\}^{-1/2}, \tag{4}$$

where $\left(\widehat{\beta}_0,\widehat{\beta}_1\right)$ is the mode of the posterior distribution (2). We note that you should not actually calculate $\widehat{P(\mathcal{D})}$. Instead, you should calculate the logarithm of the marginal likelihood $\log\widehat{P(\mathcal{D})}$.

2.3 The Metropolis-Hastings Algorithm

Sampling from the posterior distribution (2) can be done using the Metropolis-Hastings algorithm. The procedure starts with the initial values $\left(\beta_0^{(0)},\beta_1^{(0)}\right) = \left(\widehat{\beta}_0,\widehat{\beta}_1\right)$, i.e. we start right at the mode of the distribution (2). We update the current state $\left(\beta_0^{(k-1)},\beta_1^{(k-1)}\right)$ of the Markov chain to its next state $\left(\beta_0^{(k)},\beta_1^{(k)}\right)$ as follows.

We generate a candidate state $\left(\widetilde{\beta}_0,\widetilde{\beta}_1\right)$ by sampling from the bivariate normal distribution

$$N_2\left(\begin{pmatrix} \beta_0^{(k-1)} \\ \beta_1^{(k-1)} \end{pmatrix}, -\left[D^2 l^*\left(\widehat{\beta}_0,\widehat{\beta}_1\right)\right]^{-1}\right). \tag{5}$$

Note that the covariance matrix of the proposal (5) is the negative of the inverse of the Hessian matrix evaluated at the mode of (2).

We accept the move to the proposed state, i.e. we set $\left(\beta_0^{(k)}, \beta_1^{(k)}\right) = \left(\widetilde{\beta}_0, \widetilde{\beta}_1\right)$ with probability

$$\min\left\{1, \exp\left[l^*\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right) - l^*\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)\right]\right\}. \tag{6}$$

Otherwise the Markov chain stays at its current state, i.e. we set $\left(\beta_0^{(k)}, \beta_1^{(k)}\right) = \left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)$. We see that the proposal distribution (5) is symmetric, i.e. the probability of proposing $\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right)$ if the chain is currently in $\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)$ is equal with the probability of proposing $\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)$ if the chain is currently in $\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right)$. As such, the proposal distribution (5) cancels when we calculate the acceptance probability (6).

The implementation of an iteration of the Metropolis-Hastings algorithm proceeds as follows. If the proposed state leads to an increase of l^*, i.e.

$$l^*\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right) \geq l^*\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right),$$

we accept the move to the proposed state and set $\left(\beta_0^{(k)}, \beta_1^{(k)}\right) = \left(\widetilde{\beta}_0, \widetilde{\beta}_1\right)$. Otherwise, if the proposed state leads to a decrease in l^*, i.e.

$$l^*\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right) < l^*\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right),$$

we sample u from a Uniform$(0,1)$ distribution. If

$$\log(u) \leq l^*\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right) - l^*\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right),$$

we accept the move to the proposed state and set $\left(\beta_0^{(k)}, \beta_1^{(k)}\right) = \left(\widetilde{\beta}_0, \widetilde{\beta}_1\right)$. If

$$\log(u) > l^*\left(\widetilde{\beta}_0, \widetilde{\beta}_1\right) - l^*\left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)$$

we reject the move and the chain stays at the current state, i.e. we set $\left(\beta_0^{(k)}, \beta_1^{(k)}\right) = \left(\beta_0^{(k-1)}, \beta_1^{(k-1)}\right)$.

2.4 Monte Carlo Integration

You need to evaluate the integral (3) by sampling from the two independent normal priors for the regression coefficients β_0 and β_1, as follows. Simulate $\left(\beta_0^{(1)}, \beta_1^{(1)}\right), \ldots, \left(\beta_0^{(10000)}, \beta_1^{(10000)}\right)$ from a bivariate normal distribution

$$N_2\left(\left(\begin{array}{c} 0 \\ 0 \end{array}\right), \left(\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array}\right)\right).$$

The Monte Carlo estimate of (3) is given by the average

$$\frac{1}{10000}\sum_{j=1}^{10000} \exp\left(l\left(\beta_0^{(j)}, \beta_1^{(j)}\right)\right).$$

3 Project Description (300 points)

The dataset you need to use is contained in the file "534finalprojectdata.txt". This file has 148 rows (samples) and 61 columns (variables). The **first** 60 columns are associated with 60 explanatory variables X, while column 61 (**the last column**) corresponds with the response binary variable Y.

You need to write a C/C++ parallel program using the Primary/replica scheme. The primary process should proceed as follows:

1. The primary process should initialize an empty single-list $\mathcal{L}$.

2. The primary process sends work requests to the replica processes and should record the results of these requests in the list $\mathcal{L}$.

3. A work request to a replica process should contain a single integer number between 1 and 60. That is, the primary process asks a replica process to analyze the logistic regression of Y (the last column in the data) on X (the i-th column in the data).

4. The result of a work request should contain the following numbers:

   (a) the index of the explanatory variable associated with the logistic regression processed by the replica process;

   (b) the estimates of the log-marginal likelihood of this logistic regression. You need to produce two estimates: the first estimate is the Laplace approximation (4), while the second estimate is based on Monte Carlo integration as described in Section 2.4.

   (c) estimates of the coefficients β_0 and β_1 of this logistic regression. These estimates are obtained as follows:

   - Use the Metropolis-Hastings algorithm to simulate 10000 samples

     $$\left\{\left(\beta_0^{(k)}, \beta_1^{(k)}\right) : k = 1, \ldots, 10000\right\},$$

     from the posterior distribution (2).
   - The estimates for β_0 and β_1 are given by the sample means:

     $$\bar{\beta}_0 = \frac{1}{10000} \sum_{k=1}^{10000} \beta_0^{(k)}, \quad \bar{\beta}_1 = \frac{1}{10000} \sum_{k=1}^{10000} \beta_1^{(k)}.$$

5. The primary process records the information from the replica process in the list $\mathcal{L}$ such that the univariate logistic regression appear in decreasing order of their marginal log-likelihood as estimated by Monte Carlo integration. The length of $\mathcal{L}$ should not be bigger than 5 at any point during the execution of the program.

6. The primary process outputs the contents of the list $\mathcal{L}$.

7. The primary process tells the replica processes to shutdown.

Very important: please include the output from the primary process in your submission. This way we will know whether your program returns the correct output. You will determine the top five logistic regressions together with the estimates of their coefficients.

A replica process should proceed as follows:

1. The replica process should read in the data (just once, at initialization) and should start listening for orders from the primary process.
2. Once an order is received, the replica process should perform all the required calculations associated with a univariate logistic regression.
3. The replica process should return the results to the primary process.
4. The replica process should shutdown once the primary process indicates there is no further work to be executed.