STA 250 Homework 1

Yichuan Wang

Problem 2

(i)

$$Y_i|\beta \sim \text{Binomial}(m_i, \text{logit}^{-1}(x_i^T\beta)), \beta \in \mathbb{R}^p$$

So we have

$$p(y_i|\beta) = \binom{m_i}{y_i}(\frac{e^{x_i^T\beta}}{1 + e^{x_i^T\beta}})^{y_i}(1 - \frac{e^{x_i^T\beta}}{1 + e^{x_i^T\beta}})^{m_i-y_i}$$

$$= \binom{m_i}{y_i}\frac{e^{(x_i^T\beta)y_i}}{(1 + e^{x_i^T\beta})^{m_i}}$$

Then

$$p(\mathbf{y}|\beta) \propto \frac{\exp(\sum_{i=1}^n (x_i^T\beta)y_i)}{\prod_{i=1}^n (1 + e^{x_i^T\beta})^{m_i}}$$

and

$$p(\beta) \propto \exp[-\frac{1}{2}(\beta - \mu_0)^T\Sigma_0^{-1}(\beta - \mu_0)]$$

So the posterior is, up to proportionality

$$p(\beta|\mathbf{y}) \propto \frac{\exp[\sum_{i=1}^n (x_i^T\beta)y_i - \frac{1}{2}(\beta - \mu_0)^T\Sigma_0^{-1}(\beta - \mu_0)]}{\prod_{i=1}^n (1 + e^{x_i^T\beta})^{m_i}}$$

We may take logarithm of the proportional posterior for calculation, which is

$$\sum_{i=1}^n (x_i^T\beta)y_i - \frac{1}{2}(\beta - \mu_0)^T\Sigma_0^{-1}(\beta - \mu_0) - \sum_{i=1}^n m_i \log(1 + e^{x_i^T\beta})$$

(ii) In this problem, I used the Metropolis-Hastings-within-Gibbs to generate

the Markov chain for posterior distribution. The proposal distribution was

$$\begin{pmatrix} \beta_0^{(t)} \\ \beta_1^{(t)} \end{pmatrix} \Big| \begin{pmatrix} \beta_0^{(t-1)} \\ \beta_1^{(t-1)} \end{pmatrix} \sim N(\begin{pmatrix} \beta_0^{(t-1)} \\ \beta_1^{(t-1)} \end{pmatrix}, \begin{pmatrix} v_0 & 0 \\ 0 & v_1 \end{pmatrix})$$

The algorithm was performed as the following:

(1) Set initial values $\beta_0^{(0)} = 0, \beta_1^{(0)} = 0, t = 0, v_0 = v_1 = 1$.

(2) Let $\beta_0^{(t+1)} = \beta_0^{(t)}, \beta_1^{(t+1)} = \beta_1^{(t)}$, sample $\beta_0^{(t+1)}$ from $N(\beta_0^{(t)}, v_0)$.

(3) Compute the logarithm posterior of $\begin{pmatrix} \beta_0^{(t+1)} \\ \beta_1^{(t+1)} \end{pmatrix} \Big| \begin{pmatrix} \beta_0^{(t)} \\ \beta_1^{(t)} \end{pmatrix}$, $\log \pi(\beta^{(t+1)}|\beta^{(t)})$,

as well as the logarithm posterior of $\begin{pmatrix} \beta_0^{(t)} \\ \beta_1^{(t)} \end{pmatrix} \Big| \begin{pmatrix} \beta_0^{(t+1)} \\ \beta_1^{(t+1)} \end{pmatrix}$, $\log \pi(\beta^{(t)}|\beta^{(t+1)})$.

(4) Since we are using a symmetric proposal distribution, we have $\log \alpha = \log \pi(\beta^{(t)}|\beta^{(t-1)}) - \log \pi(\beta^{(t-1)}|\beta^{(t)})$. (5) According to MH algorithm, when $\alpha > 1$, we always accept sample; when $\alpha < 1$, we accept with probability α.

(6) After updating (accepting or rejecting sample) $\beta_0^{(t+1)}$, we sample $\beta_1^{(t+1)}$ from $N(\beta_1^{(t)}, v_1)$ and repeat steps (3) to (5) to update it.

(7) Set $t = t + 1$, repeat steps (2) through (6) until desired iterations or convergence.

For each dataset, 10000 iterations were executed and the first 1000 were considered as "burnin" period, during which retuning was performed at every 100th iteration: if the acceptance rate for either parameter is too low (less than 20%), the variance v_j would be divided by 1.1; if the rate is too high (greater than 60%), the variance would be multiplied by 1.1.

(iii) The following table and graph display the coverage property based on results from Gauss:

	beta_0	beta_1
p_01	0.01	0.01
p_05	0.07	0.03
p_10	0.10	0.06
p_25	0.27	0.20
p_50	0.52	0.48
p_75	0.76	0.77
p_90	0.92	0.88
p_95	0.94	0.95
p_99	0.99	0.99

Actual vs. Nominal Coverage (200 datasets)