STAT 604:
Intro. to Statistical Computing

David B. Dahl
Fall 2007

MIDTERM

20 November 2007

1. Suppose that $X_1, \ldots, X_n$ are independent Bernoulli random variables, where the success probability for the i^{th} random variable is $\theta + (\frac{i-0.5}{n} - 0.5)\beta$. That is:

$$p(x_i) = \left[\theta + \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right]^{x_i} \left[1 - \theta - \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right]^{1-x_i} \quad \text{for } x_i \in \{0, 1\},\ i = 1, \ldots, n$$

a. [5 pts.] In the context of this problem, give the function of θ and β that a maximum likelihood estimation procedure seeks to maximize. If you get stuck on a problem, you may ask for help and obtain partial credit.

$$L(\theta, \beta; x_1, \ldots, x_n) = \prod_{i=1}^{n} \left[\theta + \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right]^{x_i} \left[1 - \theta - \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right]^{1-x_i}, \quad \text{or}$$

$$l(\theta, \beta; x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i \log\left[\theta + \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right] + \sum_{i=1}^{n} (1 - x_i) \log\left[1 - \theta - \Big(\frac{i-0.5}{n} - 0.5\Big)\beta\right]$$

For simplicity, write these as $L(\theta, \beta)$ and $l(\theta, \beta)$.

b. [15 pts.] Give the steps of the Newton Raphson procedure to maximize the likelihood of θ and β. To save time, you do not need to find the Hessian, but should provide the gradient.

Define $\nabla l(\theta, \beta)$ and $D^2 l(\theta, \beta)$ as the gradient and Hessian of the log-likelihood l evaluated at (θ, β). Note that:

$$\nabla l(\theta, \beta) = \Bigg[\sum_{i=1}^{n} \frac{x_i}{\theta + \Big(\frac{i-0.5}{n} - 0.5\Big)\beta} - \sum_{i=1}^{n} \frac{1 - x_i}{1 - \theta - \Big(\frac{i-0.5}{n} - 0.5\Big)\beta}, \\ \sum_{i=1}^{n} \frac{x_i \Big(\frac{i-0.5}{n} - 0.5\Big)}{\theta + \Big(\frac{i-0.5}{n} - 0.5\Big)\beta} - \sum_{i=1}^{n} \frac{(1 - x_i)\Big(\frac{i-0.5}{n} - 0.5\Big)}{1 - \theta - \Big(\frac{i-0.5}{n} - 0.5\Big)\beta} \Bigg]$$

Let $(\hat{\theta}_0, \hat{\beta}_0) = (0.5, 0.0)$, an initial guess for the maximizer of the log-likelihood. At iteration i, let:

$$(\hat{\theta}_i, \hat{\beta}_i) = (\hat{\theta}_{i-1}, \hat{\beta}_{i-1}) - [D^2 l(\hat{\theta}_{i-1}, \hat{\beta}_{i-1})]^{-1} [\nabla l(\hat{\theta}_{i-1}, \hat{\beta}_{i-1})].$$

Continue iterating until the norm of the gradient evaluated at $(\hat{\theta}_i, \hat{\beta}_i)$ is less than a pre-specified, small number ϵ. Ensure that we have found a maximizer by verifying that the Hessian evaluated at the $(\hat{\theta}_i, \hat{\beta}_i)$ is negative definite. If so, define $(\hat{\theta}_{\text{MLE}}, \hat{\beta}_{\text{MLE}}) = (\hat{\theta}_i, \hat{\beta}_i)$.

c. [15 pts.] Describe a procedure to test $H_o : \beta = 0$ versus $H_a : \beta > 0$.

Use a permutation test: Since $\beta = 0$ under H_0, $X_1, \ldots, X_n$ are identically distributed under the null hypothesis. Create many permutation datasets by randomly shuffling the order of the data, each time computing the MLE based on the permuted data. The generates generates draws from the sampling distribution of β under the null hypothesis. Get a Monte Carlo estimate of the p-value by computing the proportion of β's from the permutation distribution that exceed $\hat{\beta}_{\text{MLE}}$ using the Newton-Raphson procedure above. If the p-value is small (say less than $\alpha = 0.05$), reject the null hypothesis and conclude $\beta > 0$.

d. [15 pts.] Now take a Bayesian approach for inference about θ and β and consider the following prior:

$$p(\theta, \beta) \propto \mathrm{I}\left\{\theta + \left(\frac{i - 0.5}{n} - 0.5\right)\beta \in (0, 1), \text{for } i = 1, \ldots, n\right\}$$

Provide the details of a Markov chain Monte Carlo algorithm to sample from the posterior distribution of θ and β. Use a bivariate random walk on the unit-square as the proposal mechanism for (θ, β).

The posterior distribution $f^\star(\theta, \beta) = p(\theta, \beta | x_1, \ldots, x_n) = c \cdot f(\theta, \beta)$, where c is an unknown normalizing constant and $f(\theta, \beta)$ is the likelihood (given in Part a) times the prior (given in the description of this part of the problem). Set $(\theta_0, \beta_0) = (0.5, 0.0)$. For a large integer B, let $i = 1, \ldots, B$:

1. Propose a value $(\theta^\star, \beta^\star) = (\theta_{i-1}, \beta_{i-1}) + (u_1, u_2)$, where u_1 and u_2 are independent draws from the uniform (0,1) distribution.
2. Compute the Metropolis ratio:
$$r_M = \frac{f(\theta^\star, \beta^\star)}{f(\theta_{i-1}, \beta_{i-1})}$$
3. If a uniform (0,1) random number is less than r_M, accept the proposal, i.e., $(\theta_i, \beta_i) = (\theta^\star, \beta^\star)$. Else, set $(\theta_i, \beta_i) = (\theta_{i-1}, \beta_{i-1})$.

Discard initial samples that indicate the initial starting value. Compute summaries of the posterior distribution using Monte Carlo integration on the remaining values.

2. [5 pts.] Add the decimal integer 56 to the binary number 110101, representing the result as a binary number. 56 (111000) + 53 (110101) = 109 (1101101)

3. [15 pts.] Compare and contrast the relative merits of rejection sampling, importance sampling, and Markov chain Monte Carlo for Monte Carlo integration. Think along the lines of ease of implementation, range of applicability in practice, and ease of assessing Monte Carlo error.

In terms of implementation, both rejection sampling and importance sampling are rather trial. Importance sampling is perhaps a bit more complicated since the weights need to be included in the Monte Carlo integration. Markov chain Monte Carlo can be easy to implement if the the proposal distribution is simple, but can also be very complicated for nontrivial proposal distributions.

In terms of applicability, rejection sampling is the most limited in its application since the target density must be known completely. Even when the density is completely known, the rejection sampler can produce little or no output if the envelop does not match the target density well. Both importance sampling and MCMC only need to know the target density up to a normalizing constant, making them more widely applicable. Importance sampling works well when the importance distribution matches the target well. Unfortunately, it can be difficult to find a good importance distribution such that the weights or standardization weights are not nearly zero except for a few draws. This is not too difficult for low dimensional problems but, in large dimensional problems, importance sampling can have a few draws that dominate the Monte Carlo integration. MCMC is much more applicable to a wide range of problems, even working in high dimensional problems (with good a proposal distribution) that would be too difficult for importance sampling.

In terms of assessing Monte Carlo error, rejection sampling is by far the easiest since it generates independent and identically distributed draws from the target distribution. Draws from importance sampling are also independent so, as long as one takes into account of the weights, it is rather straight-forward to assess variability. Assessing Monte Carlo error for MCMC is much more difficult since draws from MCMC are subject to burn-in issues and are not independent of each other.