

Economics 696, Lecture Note 6: Acceptance Sampling, Importance Sampling, and Data Augmentation

1 Probit Model

As an alternative to the logit model, consider the probit model:

$$\Pr(Y_i = 1|x) = \Phi(x'_i\beta),$$

where Φ is the cumulative distribution function of the standard normal distribution:

$$\Phi(b) = \int_{-\infty}^b \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) dt,$$

One appealing feature of the probit setup is that it generalizes well to multinomial and multivariate models. The (partial) likelihood function for the probit model can be written as:

$$f(y|x, \beta) = \prod_{i=1}^n \Phi(x'_i\beta)^{y_i} (1 - \Phi(x'_i\beta))^{1-y_i},$$

where $y = (y_1, \dots, y_n)'$ and $x = (x_1, \dots, x_n)'$. This is not too hard to calculate on a computer (for example, in Matlab there is a function called `erf` which evaluates Φ), although for generalizations of the probit model, simply calculating the likelihood can be very difficult.

The posterior distribution for this model (with prior $p(\beta)$) can be written using Bayes Theorem as:

$$p(\beta|y, x) = \frac{f(y|x, \beta)p(\beta)}{\int f(y|x, \beta)p(\beta)d\beta}.$$

(Note we will implicitly make the assumptions in Section 5 of LN4, which justify working with the partial likelihood.)

In this case, it is easy to calculate the numerator term $f(y|x, \beta)p(\beta)$, but directly calculating the integral in the denominator of the posterior is difficult, for similar reasons to the logit model.

2 Acceptance Sampling

In LN4, we used Monte Carlo simulation to generate random draws from the posterior densities. This was possible because the posterior distributions had forms that are well known (normal, multivariate normal, chi-square, etc.) and for which there are relatively simple ways to generate random draws for these distributions. Using the computer, we could take a large number of draws, and form sample averages to get good approximations to various posterior expectations.

Suppose we want to draw random samples from some density $p(\theta)$, but we cannot recognize it as falling in a standard class of distributions, so we cannot draw from it directly. Acceptance sampling is a method to do this, based on draws from some *other* distribution.

Let $p_S(\theta)$ be a “source” density from which we can easily generate random draws. Suppose that $p_S(\theta)$ can be factored into a constant times some kernel:

$$p_S(\theta) = C_S \cdot k_S(\theta).$$

Likewise, suppose that the “target” density $p(\theta)$ can be factored into a constant times some kernel:

$$p(\theta) = C \cdot k(\theta).$$

In this approach, we only need to be able to evaluate $k_S(\theta)$ and $k(\theta)$, and generate draws from $p_S(\theta)$. Of course, we could always take the constant to be 1, but this is useful for working with posterior distributions, because we could set the kernel of the target density to be the prior times the likelihood.

Let

$$r = \sup_{\theta \in \Theta} \frac{k(\theta)}{k_S(\theta)}$$

and suppose $r < \infty$. The acceptance algorithm generates, for each step $j = 1, \dots, J$, a draw $\theta^{(j)}$ as follows:

1. Draw $u \sim Unif[0, 1]$.
2. Draw $\theta^* \sim p_S(\theta)$, independently of u .
3. If $u > k(\theta^*)/[r \cdot k_S(\theta^*)]$, then return to step 1.
4. Set $\theta^{(j)} = \theta^*$.

Then the $\theta^{(j)}$ will be independent draws from $p(\theta)$. (For a proof of why this works, see Geweke, 4.1.)

So, for some function $g(\theta)$, for which we want to approximate

$$E[g(\theta)] = \int g(\theta)p(\theta)d\theta,$$

we can form

$$\frac{1}{J} \sum_{j=1}^J g(\theta^{(j)}).$$

Note that in practice, we will have to generate more than J draws from $p_S(\theta)$, because not all will be “accepted.” This will depend on how much p_S differs from p .

3 Importance Sampling

Importance sampling is similar to acceptance sampling, but we use all the draws from $p_S(\theta)$. Instead of accepting or rejecting them, we weight the draws when forming sample averages.

As before, let the source density be

$$p_S(\theta) = C_S k_S(\theta)$$

and let the target density be

$$p(\theta) = C k(\theta).$$

For $j = 1, \dots, J$, let $\theta^{(j)}$ be independent draws from $p_S(\theta)$. Define weights

$$w^{(j)} = \frac{k(\theta^{(j)})}{k_S(\theta^{(j)})}.$$

Then, we can approximate

$$E[g(\theta)] = \int g(\theta) p(\theta) d\theta$$

by

$$\frac{\frac{1}{J} \sum_j w^{(j)} g(\theta^{(j)})}{\frac{1}{J} \sum_j w^{(j)}}.$$

To see why this works, note that

$$\begin{aligned} \frac{\int g(\theta) \frac{k(\theta)}{k_S(\theta)} p_S(\theta) d\theta}{\int \frac{k(\theta)}{k_S(\theta)} p_S(\theta) d\theta} &= \frac{\int g(\theta) \frac{C_S}{C} p(\theta) d\theta}{\int \frac{C_S}{C} p(\theta) d\theta} \\ &= \int g(\theta) p(\theta) d\theta. \end{aligned}$$

Formally, we need the support of $p_S(\theta)$ to include the support of $p(\theta)$, so that the weights are always well defined, and we need the weighting to be bounded and for the various moments to exist.

As a practical matter, if the source density is very different from the target density, then some weights will be close to zero, so that the sample average is dominated by relatively few terms, rendering it a poor approximation.

4 Data Augmentation for the Probit Model

For concreteness, let us return to the probit model introduced at the beginning of this note. The probit model can be rewritten in a *Latent Variable Form*:

$$\epsilon_i | X \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1).$$

$$y_i^* = x_i' \beta + \epsilon_i,$$

$$y_i = \begin{cases} 1 & \text{if } y_i^* > 0 \\ 0 & \text{if } y_i^* \leq 0 \end{cases}$$

This is an equivalent model, in the sense that for any given β and any X it yields the same distribution for y . To see this, note that

$$\begin{aligned} Pr(y_i = 1|x, \beta) &= Pr(y_i^* > 0|x, \beta) \\ &= Pr(x_i' \beta + \epsilon_i > 0|x, \beta) \\ &= Pr(\epsilon_i > -x_i' \beta|x, \beta) \\ &= Pr(\epsilon_i < x_i' \beta|x, \beta) \\ &= \Phi(x_i' \beta). \end{aligned}$$

It will be useful to work with the joint distribution of β and the latent variables $y^* \equiv (y_1^*, \dots, y_n^*)$. We can write

$$\begin{aligned} p(\beta, y^*|y, x) &\propto p(\beta, y^*|x)p(y|y^*, x, \beta) \\ &\propto p(\beta|x)p(y^*|\beta, x)p(y|y^*, x, \beta) \end{aligned}$$

By the partial likelihood assumptions $p(\beta|x) = p(\beta)$. The other two terms on the right hand side of the preceding display have a product form, so we can write

$$p(\beta, y^*|y, x) \propto p(\beta) \prod_{i=1}^n p(y_i^*|x_i, \beta) \prod_{i=1}^n p(y_i|y_i^*, x_i, \beta)$$

Note that

$$p(y_i^*|x_i, \beta) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y_i^* - x_i' \beta)^2\right)$$

and

$$p(y_i|y_i^*, x_i, \beta) = 1(y_i = 1)1(y_i^* > 0) + 1(y_i = 0)1(y_i^* \leq 0).$$

Putting all this together, we get:

$$p(\beta, y^*|y, x) \propto p(\beta) \prod_{i=1}^n \{1(y_i = 1)1(y_i^* > 0) + 1(y_i = 0)1(y_i^* \leq 0)\} \phi(y_i^*|x_i' \beta, 1),$$

where $\phi(\cdot|\mu, \sigma^2)$ denotes the normal density function with mean μ and variance σ^2 .

Suppose that we could actually observe the latent y_i^* . Let $y^* = (y_1^*, \dots, y_n^*)'$. Then the posterior distribution for β would be easy to calculate. If we use a constant prior for β , we could use the results in Lecture Note 4 to obtain

$$\beta|y^*, X \sim \mathcal{N}(b^*, (X'X)^{-1}),$$

where

$$b^* = (X'X)^{-1}X'y^*.$$

Formally, we can write

$$\begin{aligned} p(\beta|y^*, y, x) &= \frac{p(\beta, y^*|y, x)}{p(y^*|y, x)} \\ &\propto p(\beta) \prod_{i=1}^n \phi(y_i^*|x_i'\beta, 1) \end{aligned}$$

which gives the desired result.

Conversely, suppose we knew β . Then it would be easy to draw for the latent variables y^* from their distribution conditional on β and x , which is normal with mean $x_i'\beta$ and variance 1. However, if we also condition on y_i , we need to take into account this additional source of information. Heuristically, observing $y_i = 1$ tells us that $y_i^* > 0$ and observing $y_i = 0$ tells us that $y_i^* \leq 0$. Formally, write

$$p(y_i^*|\beta, x_i, y_i) \propto \begin{cases} \phi(y_i^*|x_i'\beta, 1)1(y_i^* > 0) & \text{if } y_i = 1 \\ \phi(y_i^*|x_i'\beta, 1)1(y_i^* \leq 0) & \text{if } y_i = 0 \end{cases}$$

Thus, if $y_i = 1$, the distribution of y_i^* is a normal distribution with mean $x_i'\beta$ and variance 1, truncated so that y_i^* is strictly positive. Likewise, if $y_i = 0$ then the distribution is truncated so that y_i^* is negative.

The idea behind data augmentation is to augment the observed data with the latent data, and iterate between these two distributions, generating a sequence of draws for β and y^* . Start by setting $\beta^{(1)} = (0, \dots, 0)'$ or some other vector of numbers. (We will use superscript to denote the iteration number.) Then draw:

$$\begin{aligned} y^{*(1)} &\sim p(y^*|\beta^{(1)}, x, y) \\ \beta^{(2)} &\sim p(\beta|y^{*(1)}, x, y) \\ y^{*(2)} &\sim p(y^*|\beta^{(2)}, x, y) \\ &\vdots \end{aligned}$$

At each draw for β , we substitute the most recent draw for the latent y^* , and likewise in the draws for y^* . It can then be shown that

$$(\beta^{(J)}, y^{*(J)}) \xrightarrow{\text{d.}} p(\beta, y^*|y, x), \quad \text{as } J \rightarrow \infty. \quad (1)$$

In addition, for any integrable function $g(\beta, y^*)$,

$$\frac{1}{J} \sum_{j=1}^J g(\beta^{(j)}, y^{*(j)}) \xrightarrow{\text{a.s.}} \int g(\beta, y^*) p(\beta, y^*|y, x) d\beta dy^*, \quad \text{as } J \rightarrow \infty. \quad (2)$$

The average on the left of expression (2) is a “time” or *ergodic* average. Notice that the result is similar to a law of large numbers, but that we are not generating the draws for β, y^* independently. In fact, we are not even generating them from the “correct” distribution, although according to expression (1) the distribution approaches the correct distribution eventually.

Since according to (1) the posterior distribution of β, y^* converges jointly, it is also true that the marginal distributions converge, so

$$\beta^{(j)} \xrightarrow{d.} p(\beta|y, x), \quad \text{as } J \rightarrow \infty.$$

Also, it follows from (2) that for an integrable function $h(\beta)$,

$$\frac{1}{J} \sum_{j=1}^J h(\beta^{(j)}) \xrightarrow{\text{a.s.}} \int h(\beta)p(\beta|y, x)d\beta, \quad \text{as } J \rightarrow \infty.$$

5 General Data Augmentation Algorithm

This idea generalizes to models with missing data or latent variables. Let us denote the *observed* data by z_o and the *latent* variables by z_l . The *complete-data likelihood* is a conditional density $p(z_l, z_o|\theta)$, where θ is a parameter vector. The *observed-data likelihood* integrates over the latent variables:

$$p(z_o|\theta) = \int p(z_l, z_o|\theta)dz_l.$$

We assign a *prior* $p(\theta)$ to the parameter vector, and we are interested in calculating the joint posterior $p(\theta, z_l|z_o)$. We might also be interested in calculating the marginal posterior $p(\theta|z_o)$ which can be obtained from the joint posterior. Suppose that the “full conditional” distributions

$$p(z_l|\theta, z_o)$$

and

$$p(\theta|z_o, z_l)$$

are easy to sample from. (This is often the case if the model for θ has a simple form conditional on the latent variables z_l .) Then we can proceed as before; initialize $\theta^{(1)}$ to some value, and then draw

$$\begin{aligned} z_l^{(1)} &\sim p(z_l|\theta^{(1)}, z_o) \\ \theta^{(2)} &\sim p(\theta|z_o, z_l^{(1)}) \\ z_l^{(2)} &\sim p(z_l|\theta^{(2)}, z_o) \\ &\vdots \end{aligned}$$

As before the draws for (θ, z_l) will converge to their posterior distribution conditional on z_o , and ergodic averages will converge to expectations with respect to $p(\theta, z_l|z_o)$.

You might wonder about the initialization of the process. The convergence holds for any initial values (and we could change the order and initialize z_l , then draw for θ). However, the choice of starting value can affect the quality of the approximations. We will return the choice of starting values and other issues related to the implementation of this procedure in later lectures.

The data augmentation (DA) principle is useful whenever: (a) the model has a latent variable structure; and (b) the *complete-data* version of the model is easier to work with than the *observed-data* model. In particular, it can handle some very complicated models, for which even just evaluating the likelihood function is difficult. It was first suggested by Tanner and Wong (1987). Albert and Chib (1993) develop the probit routine discussed here. DA is a special case of the Gibbs sampler, to be discussed next.