Data Augmentation / Latent Variable Models

(KPT Ch. 14)

AAEC 6564
Instructor: KLAUS MOELTNER

Matlab scripts: `mod5_probit_data, mod5_probit, mod5_probit_Fair_predict, mod5_probit_Fair, mod5_probit_Fair_nokids, mod5_probit_Fair_chib, mod5_probit_Fair_nokids_chib, mod5_tobit_data, mod5_tobit, mod5_tobit_adoption, mod5_tobit_adoption_predict, mod5_tobit_adoption_naive, mod5_tobit_adoption_naive_predict, mod5_tobit_adoption_chib, mod5_tobit_adoption_notraining, mod5_tobit_adoption_notraining_chib`

Matlab functions: `gs_probit, gs_tobit, gs_probit_chib, gs_tobit_chib`

The concept of *data augmentation* within a Bayesian estimation framework was first proposed by Tanner and Wong (1987). The basic idea is to formally introduce a new set of *elements* into the joint posterior density to *facilitate* posterior simulation.

These "elements" are generally not observed in reality. I choose the neutral word "elements", as these added constructs can take the form of latent (unobserved) "data", or the form of additional parameters. Thus, in principle, "data augmentation" might equivalently be called "parameter augmentation", or, more generally, "posterior augmentation". However, we'll stick with the standard label "data augmentation" for this chapter.

What is meant by "*facilitate*"? The standard interpretation in this context is to enable the implementation of a straightforward Gibbs Sampler with well-understood conditional posteriors. The alternative in absence of data augmentation would be to approximate the entire posterior or conditional parts of it via Metropolis-Hastings or equivalent methods. However, such approximations usually come at the cost of inefficiencies in posterior sampling. In most applications, superior results can be achieved via data augmentation.

Conceptually, reconsider the basic structure of a posterior kernel:

$$p(\boldsymbol{\theta}\,|\,\mathbf{y}) = \frac{p(\boldsymbol{\theta})\,p(\mathbf{y}\,|\,\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\boldsymbol{\theta})\,p(\mathbf{y}\,|\,\boldsymbol{\theta}) \tag{1}$$

where $\boldsymbol{\theta}$ comprises the original parameters of interest, and $\mathbf{y}$ denotes observed data. Assume that the resulting posterior kernel cannot be broken into well-understood conditional components suitable for Gibbs Sampling. Now add the (unobserved) augmentation vector $\mathbf{z}$ to the model.

First, let's treat $\mathbf{z}$ more like a parameter than "data" and introduce it via a joint prior. The augmented posterior can then be written as

$$p(\boldsymbol{\theta},\mathbf{z}\,|\,\mathbf{y}) = \frac{p(\boldsymbol{\theta},\mathbf{z})\,p(\mathbf{y}\,|\,\boldsymbol{\theta},\mathbf{z})}{p(\mathbf{y})} \propto p(\boldsymbol{\theta},\mathbf{z})\,p(\mathbf{y}\,|\,\boldsymbol{\theta},\mathbf{z}) = p(\boldsymbol{\theta})\,p(\mathbf{z}\,|\,\boldsymbol{\theta})\,p(\mathbf{y}\,|\,\boldsymbol{\theta},\mathbf{z}) \tag{2}$$

As indicated in (1) and to make this approach computationally feasible, the joint prior can be decomposed into the usual prior for $\boldsymbol{\theta}$ and a conditional prior $p(\mathbf{z}|\boldsymbol{\theta})$. This is a typical setup in hierarchical regression modeling where **z** might include unobserved random effects, and $\boldsymbol{\theta}$ might include the parameters of the hierarchical distribution for these effects. Note that the prior of **z** may be conditioned on the entire $\boldsymbol{\theta}$ or only on a sub-set of $\boldsymbol{\theta}$.

We could also conceptualize this augmented model by introducing **z** into the likelihood, i.e:

$$p(\boldsymbol{\theta},\mathbf{z}|\mathbf{y}) = \frac{p(\boldsymbol{\theta})\,p(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\boldsymbol{\theta})\,p(\mathbf{y},\mathbf{z}|\boldsymbol{\theta}) = p(\boldsymbol{\theta})\,p(\mathbf{z}|\boldsymbol{\theta})\,p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z}) \tag{3}$$

This approach fits the augmentation intuition behind latent variable models, as will be discussed in more detail below.

Both conceptual approaches are equivalent, as is visible form the last expression in (2) and (3). KPT call $p(\mathbf{y},\mathbf{z}|\boldsymbol{\theta})$ the *complete or augmented data density* ("data density " is synonymous with "sample likelihood"). I tend to call $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z})$ the "conditional likelihood", i.e. the sample likelihood conditioned on the augmented data.

To derive our original posterior of interest, i.e. $p(\boldsymbol{\theta}|\mathbf{y})$, we need to integrate the augmentation vector out of the posterior density. Formally:

$$p(\boldsymbol{\theta}|\mathbf{y}) = \int_{\mathbf{z}} p(\boldsymbol{\theta},\mathbf{z}|\mathbf{y})\,d\mathbf{z} = \int_{\mathbf{z}} p(\boldsymbol{\theta}|\mathbf{z},\mathbf{y})\,p(\mathbf{z}|\mathbf{y})\,d\mathbf{z} \tag{4}$$

In practice, this can be accomplished by drawing **z**, conditional on **y** and $\boldsymbol{\theta}$, along with the actual parameters in an "augmented" posterior simulation routine.

The data augmentation step can simplify posterior sampling in many different ways. Here are some examples:

Example: Latent Variable Models

For latent variable models data augmentation is best interpreted as an introduction of the augmenting elements into the likelihood function. In fact, we quite literally augment the joint posterior with unobserved ("latent") data.

Consider, for example, the generic Probit model:

$$\begin{aligned} y_i^* &= \mathbf{x_i'}\boldsymbol{\beta} + \varepsilon_i \qquad && \varepsilon_i \sim n(0,1) \\ y_i &= 1 \qquad && \text{if} \qquad y_i^* > 0, \\ y_i &= 0 \qquad && \text{otherwise} \end{aligned} \tag{5}$$

where y_i^* is some unobserved (latent) construct, observed only in form of a binary outcome y_i. Given the required variance restriction for the error term, the only original parameter is $\boldsymbol{\beta}$.

A threshold of zero is the standard choice, but one could specify a different threshold in a given application. We first derive the probability for each binary outcome as:

$$pr(y_i = 0) = pr(y_i^* \le 0) = pr(\varepsilon_i \le -\mathbf{x_i'\beta}) = \int_{-\infty}^{-\mathbf{x_i'\beta}} \tfrac{1}{\sqrt{2\pi}} \exp\left(-\tfrac{1}{2}\varepsilon_i^2\right) d\varepsilon_i = \Phi(-\mathbf{x_i'\beta})$$

$$pr(y_i = 1) = pr(y_i^* > 0) = pr(\varepsilon_i > -\mathbf{x_i'\beta}) = pr(\varepsilon_i \le \mathbf{x_i'\beta}) = \tag{6}$$

$$\int_{-\infty}^{\mathbf{x_i'\beta}} \tfrac{1}{\sqrt{2\pi}} \exp\left(-\tfrac{1}{2}\varepsilon_i^2\right) d\varepsilon_i = \Phi(\mathbf{x_i'\beta})$$

where the last equality in the second line follows from the symmetry property of the standard normal density, and symbol $\Phi(.)$ denotes the *cdf* for the standard normal density. This *cdf* does not have a closed-form analytical expression, but can be fast and accurately approximated in most statistical packages. In Matlab, use `normcdf()` or `mvncdf()` for the univariate and multivariate case, respectively.

The likelihood function can be expressed in several different ways, the most popular being:

$$p(\mathbf{y} \mid \boldsymbol{\beta}, \mathbf{X}) = \prod_{i=1}^{n} \left(\Phi(-\mathbf{x_i'\beta})\right)^{1-y_i} \left(\Phi(\mathbf{x_i'\beta})\right)^{y_i} \tag{7}$$

This expression will automatically "pick" the correct $\Phi(.)$ term for each of the two possible outcomes. We'll assign the usual normal prior to $\boldsymbol{\beta}$, the only parameter vector in this model:

$$p(\boldsymbol{\beta}) = (2\pi)^{-k/2} |\mathbf{V_0}|^{-1/2} \exp\left(-\tfrac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\mu_0})' \mathbf{V_0^{-1}} (\boldsymbol{\beta} - \boldsymbol{\mu_0})\right) \tag{8}$$

The generic joint posterior kernel, in this case identical to the "conditional" posterior kernel for $\boldsymbol{\beta}$, can then be written as

$$p(\boldsymbol{\beta} \mid \mathbf{y}, \mathbf{X}) \propto \exp\left(-\tfrac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\mu_0})' \mathbf{V_0^{-1}} (\boldsymbol{\beta} - \boldsymbol{\mu_0})\right) * \prod_{i=1}^{n} \left(\Phi(-\mathbf{x_i'\beta})\right)^{1-y_i} \left(\Phi(\mathbf{x_i'\beta})\right)^{y_i} \tag{9}$$

where Φ denotes the *cdf* of the standard normal distribution. This expression does not lead to a well-understood density from which $\boldsymbol{\beta}$ could be drawn.

No consider a model that introduces the latent data $\mathbf{y}^*$. Conceptually, we then obtain the following augmented posterior:

$$p(\boldsymbol{\beta}, \mathbf{y}^* \mid \mathbf{y}, \mathbf{X}) \propto p(\boldsymbol{\beta})\, p(\mathbf{y}^* \mid \boldsymbol{\beta}, \mathbf{X})\, p(\mathbf{y} \mid \mathbf{y}^*, \boldsymbol{\beta}, \mathbf{X}) \tag{10}$$

Will this simplify our situation? Conditioned only on $\boldsymbol{\beta}$, the latent vector $\mathbf{y}^*$ simply follows the normal density as given in (5). Thus, we have

$$p\left(\mathbf{y}^* \mid \boldsymbol{\beta}, \mathbf{X}\right) = (2\pi)^{-n/2} \exp\left(-\tfrac{1}{2}\left(\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)\right)\right) \tag{11}$$

For the second term, $p\left(\mathbf{y} \mid \mathbf{y}^*, \boldsymbol{\beta}\right)$, we note that given a latent observation y_i^*, the binary numerical value of y_i is determined with certainty, regardless of $\boldsymbol{\beta}$. Thus, as shown in KPT, p. 205, we can write

$$p\left(\mathbf{y} \mid \mathbf{y}^*, \boldsymbol{\beta}, \mathbf{X}\right) = p\left(\mathbf{y} \mid \mathbf{y}^*\right) = \prod_{i=1}^{n}\left(I\left(y_i = 0\right) I\left(y_i^* \le 0\right) + I\left(y_i = 1\right) I\left(y_i^* > 0\right)\right) \tag{12}$$

The first equality is key – it is exactly this de-linking of the likelihood function from $\boldsymbol{\beta}$ (made possible through *data augmentation*) that will simplify our GS. The term on the right hand side equals 1 of course (since $\mathbf{y}^*$ predicts $\mathbf{y}$ with certainty) and seems overly complex. However, this form will be convenient to derive the conditional posterior for $\mathbf{y}^*$ below. It simply states that when $y_i^* \le 0$, $y_i = 0$ with probability 1, and when $y_i^* > 0$, $y_i = 1$ with probability 1 (and vice versa, of course).

The augmented joint posterior kernel can now be written as:

$$\begin{aligned} &p\left(\boldsymbol{\beta}, \mathbf{y}^* \mid \mathbf{y}, \mathbf{X}\right) \propto \exp\left(-\tfrac{1}{2}\left(\boldsymbol{\beta} - \boldsymbol{\mu}_0\right)' \mathbf{V}_0^{-1}\left(\boldsymbol{\beta} - \boldsymbol{\mu}_0\right)\right) * \exp\left(-\tfrac{1}{2}\left(\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)\right)\right) * \\ &\prod_{i=1}^{n}\left(I\left(y_i = 0\right) I\left(y_i^* \le 0\right) + I\left(y_i = 1\right) I\left(y_i^* > 0\right)\right) \end{aligned} \tag{13}$$

The conditional posterior kernel for $\boldsymbol{\beta}$ now takes the following form:

$$p\left(\boldsymbol{\beta} \mid \mathbf{y}^*, \mathbf{y}, \mathbf{X}\right) \propto \exp\left(-\tfrac{1}{2}\left(\boldsymbol{\beta} - \boldsymbol{\mu}_0\right)' \mathbf{V}_0^{-1}\left(\boldsymbol{\beta} - \boldsymbol{\mu}_0\right)\right) * \exp\left(-\tfrac{1}{2}\left(\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)\right)\right) \tag{14}$$

As for the generic normal regression model, this leads to a multivariate normal density and thus solves our original problem. However, we also need to ascertain that we can also easily draw the augmented terms $\mathbf{y}^*$, otherwise we would have simply traded one problem for another.

The conditional posterior kernel for $\mathbf{y}^*$ will encompass all components of the joint posterior kernel in (13) that include $\mathbf{y}^*$, i.e

$$p\left(\mathbf{y}^* \mid \boldsymbol{\beta}, \mathbf{y}, \mathbf{X}\right) \propto \exp\left(-\tfrac{1}{2}\left(\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}\right)\right)\right) * \prod_{i=1}^{n}\left(I\left(y_i = 0\right) I\left(y_i^* \le 0\right) + I\left(y_i = 1\right) I\left(y_i^* > 0\right)\right) \tag{15}$$

More intuition is gained by writing this at the individual observation level, i.e.

$$p\left(y_i^* \mid \boldsymbol{\beta}, y_i, \mathbf{x_i}\right) \propto \exp\left(-\tfrac{1}{2}\left(y_i^* - \mathbf{x}_i'\boldsymbol{\beta}\right)^2\right) * \left(I\left(y_i = 0\right) I\left(y_i^* \le 0\right) + I\left(y_i = 1\right) I\left(y_i^* > 0\right)\right) \tag{16}$$

which immediately implies

$$\begin{aligned} p\left(y_i^* \mid \boldsymbol{\beta}, y_i = 0, \mathbf{x_i}\right) &\propto \exp\left(-\tfrac{1}{2}\left(y_i^* - \mathbf{x}_i'\boldsymbol{\beta}\right)^2\right) * I\left(y_i^* \le 0\right) \\ p\left(y_i^* \mid \boldsymbol{\beta}, y_i = 1, \mathbf{x_i}\right) &\propto \exp\left(-\tfrac{1}{2}\left(y_i^* - \mathbf{x}_i'\boldsymbol{\beta}\right)^2\right) * I\left(y_i^* > 0\right) \end{aligned} \tag{17}$$

These are kernels for the truncated-from-above-at-zero and truncated-from-below-at-zero normal densities, respectively. Thus, we can write

$$\begin{aligned} y_i^* \mid \boldsymbol{\beta}, y_i = 0, \mathbf{x_i} &\sim tn\left(\mathbf{x}_i'\boldsymbol{\beta}, 1, -\infty, 0\right) \\ y_i^* \mid \boldsymbol{\beta}, y_i = 1, \mathbf{x_i} &\sim tn\left(\mathbf{x}_i'\boldsymbol{\beta}, 1, 0, \infty\right) \end{aligned} \tag{18}$$

where $tn\left(a, b, c, d\right)$ denotes the univariate truncated normal density with mean *a*, standard deviation *b*, lower truncation bound *c*, and upper truncation bound *d*. In Matlab, you can use `tnormrnd` to draw form this density, and `tnormpdf` to evaluate it.

The Probit model with simulated data is implemented via Matlab scripts `mod5_probit_data`, and `mod5_probit`, and function `gs_probit`.

Posterior quantities of interest in the Probit model

There are two main *posterior predictive constructs* of interest in a Probit framework:

1. The probability of a "1" (or "positive") outcome given a specific setting for the explanatory variables, which can be written as

$$p\left(\Pr\left(y_p = 1 \mid \mathbf{x_p}\right)\right) = p\left(\Pr\left(y_p^* > 0 \mid \mathbf{x_p}\right)\right) = \int_{\boldsymbol{\beta}} \Pr\left(y_p^* > 0 \mid \mathbf{x_p}, \boldsymbol{\beta}\right) d\boldsymbol{\beta} = \int_{\boldsymbol{\beta}} \Phi\left(\mathbf{x_p}'\boldsymbol{\beta}\right) d\boldsymbol{\beta} \tag{19}$$

2. The marginal effect of explanatory variables on this probability. For continuous regressors, this can be expressed as

$$p\left(\frac{\partial \Pr\left(y_p = 1 \mid \mathbf{x_p}\right)}{\partial x_{pj}}\right) = p\left(\frac{\partial \Pr\left(y^*{}_p > 0 \mid \mathbf{x_p}\right)}{\partial x_{pj}}\right) = \int_{\boldsymbol{\beta}} \beta_j \phi\left(\mathbf{x_p}'\boldsymbol{\beta}\right) d\boldsymbol{\beta} \tag{20}$$

   For binary regressors, we use

$$p\left(\Pr\left(y_p=1|\mathbf{x}_{\mathbf{p},-j},x_j=1\right)-\Pr\left(y_p=1|\mathbf{x}_{\mathbf{p},-j},x_j=0\right)\right)=$$
$$\int_{\boldsymbol{\beta}}\left(\Phi\left(\mathbf{x}_\mathbf{p}'\boldsymbol{\beta}\,|\,x_{pj}=1\right)-\Phi\left(\mathbf{x}_\mathbf{p}'\boldsymbol{\beta}\,|\,x_{pj}=0\right)\right)d\boldsymbol{\beta} \tag{21}$$

In both cases it is customary to set the remaining values in $\mathbf{x_p}$ to the sample mean (see KPT p. 209). The Matlab script `mod5_probit_Fair_predict` shows how this is implemented.

The Tobit Model

The Tobit model is very similar to the Probit. It occurs when we observe the explanatory variables for all observations, but the dependent variable only for a sub-set of the data. This is often referred to as "Censoring". The larger the unobserved portion, the more important is it to recognize the censoring in your data, as the basic normal regression model would generate misleading results. The structural model is given as

$$\begin{aligned} &y_i^*=\mathbf{x_i}'\boldsymbol{\beta}+\varepsilon_i && \varepsilon_i\sim n\left(0,\sigma^2\right)\\ &y_i=y_i^* && \text{if} \quad y_i^*>0,\\ &y_i=0 && \text{otherwise}\end{aligned} \tag{22}$$

Again, the threshold of zero is an arbitrary standard choice and could be changed in a given application. Note that the error variance is now identified, although poorly if the degree of censoring is high.

The probability of a "0"outcome is now derived as

$$pr\left(y_i=0\right)=pr\left(y_i^*\le 0\right)=pr\left(\tfrac{\varepsilon_i}{\sigma}\le\tfrac{-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right)=\int_{-\infty}^{-\mathbf{x_i}'\boldsymbol{\beta}/\sigma}\tfrac{1}{\sqrt{2\pi}}\exp\left(-\tfrac{1}{2}\left(\tfrac{\varepsilon_i}{\sigma}\right)^2\right)d\varepsilon_i=\Phi\left(\tfrac{-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right) \tag{23}$$

The likelihood function can be expressed as:

$$p\left(\mathbf{y}\,|\,\boldsymbol{\beta},\sigma^2,\mathbf{X}\right)=\prod_{i:y_i=0}\Phi\left(\tfrac{-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right)\prod_{i:y_i>0}\tfrac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\tfrac{1}{2\sigma^2}\left(y_i-\mathbf{x_i}'\boldsymbol{\beta}\right)^2\right)=$$
$$\prod_{i:y_i=0}\Phi\left(\tfrac{-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right)\prod_{i:y_i>0}\tfrac{1}{\sigma}\tfrac{1}{\sqrt{2\pi}}\exp\left(-\tfrac{1}{2}\left(\frac{y_i-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right)^2\right)=\prod_{i:y_i=0}\Phi\left(\tfrac{-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right)\prod_{i:y_i>0}\tfrac{1}{\sigma}\phi\left(\frac{y_i-\mathbf{x_i}'\boldsymbol{\beta}}{\sigma}\right) \tag{24}$$

where $\phi(.)$ denotes the standard normal pdf.

We assign the usual normal and inverse-gamma priors to $\boldsymbol{\beta}$ and σ^2, respectively:

$$p(\boldsymbol{\beta}) = (2\pi)^{-k/2} |\mathbf{V_0}|^{-1/2} \exp\left(-\tfrac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu_0})' \mathbf{V_0^{-1}} (\boldsymbol{\beta}-\boldsymbol{\mu_0})\right)$$
$$p(\sigma^2) = \frac{\tau_0^{\nu_0}}{\Gamma(\nu_0)} (\sigma^2)^{-(\nu_0+1)} \exp\left(-\frac{\tau_0}{\sigma^2}\right) \tag{25}$$

Thus, the (joint) posterior kernel emerges as

$$p(\boldsymbol{\beta}, \sigma^2 \mid \mathbf{y}, \mathbf{X}) = \exp\left(-\tfrac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu_0})' \mathbf{V_0^{-1}} (\boldsymbol{\beta}-\boldsymbol{\mu_0})\right) * (\sigma^2)^{-(\nu_0+1)} \exp\left(-\frac{\tau_0}{\sigma^2}\right) *$$
$$\prod_{i:y_i=0} \Phi\left(\tfrac{-\mathbf{x_i'}\boldsymbol{\beta}}{\sigma}\right) \prod_{i:y_i>0} \tfrac{1}{\sigma}\phi\left(\frac{y_i - \mathbf{x_i'}\boldsymbol{\beta}}{\sigma}\right) \tag{26}$$

As for the Probit, this would not lead to well-understood conditional posteriors, so we'll introduce again the latent vector $\mathbf{y}^*$ as augmented data:

$$p(\boldsymbol{\beta}, \sigma^2, \mathbf{y}^* \mid \mathbf{y}, \mathbf{X}) \propto p(\boldsymbol{\beta}, \sigma^2, \mathbf{y}^* \mid \mathbf{X}) p(\mathbf{y} \mid \boldsymbol{\beta}, \sigma^2, \mathbf{y}^*) = p(\boldsymbol{\beta}) p(\sigma^2) p(\mathbf{y}^* \mid \boldsymbol{\beta}, \sigma^2, \mathbf{X}) p(\mathbf{y} \mid \boldsymbol{\beta}, \sigma^2, \mathbf{y}^*) \tag{27}$$

Conditioned only on $\boldsymbol{\beta}$ and σ^2 the latent vector $\mathbf{y}^*$ simply follows the normal density as given in (22) Thus, we have

$$p(\mathbf{y}^* \mid \boldsymbol{\beta}, \sigma^2, \mathbf{X}) = (2\pi)^{-n/2} (\sigma^2)^{-n/2} \exp\left(-\tfrac{1}{2\sigma^2}\left((\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta})\right)\right) \tag{28}$$

For the second term, $p(\mathbf{y} \mid \boldsymbol{\beta}, \sigma^2, \mathbf{y}^*)$, we note as before that given a latent observation y_i^*, the value of y_i is determined with certainty, regardless of $\boldsymbol{\beta}$ and σ^2. Thus, we can write

$$p(\mathbf{y} \mid \mathbf{y}^*, \boldsymbol{\beta}, \sigma^2) = p(\mathbf{y} \mid \mathbf{y}^*) = \prod_{i=1}^{n} \left(I(y_i = 0) I(y_i^* \le 0) + I(y_i = y_i^*) I(y_i^* > 0)\right) \tag{29}$$

The only difference to (12) is that $y_i = y_i^*$ if $y_i^* > 0$ as opposed to $y_i = 1$ in the Probit. Again, note the de-coupling of the likelihood function from the main model parameters $\boldsymbol{\beta}$ and σ^2. This "cleans" up our Gibbs Sampler.

The *augmented* joint posterior kernel takes the following form:

$$p(\boldsymbol{\beta}, \sigma^2, \mathbf{y}^* \mid \mathbf{y}, \mathbf{X}) \propto \exp\left(-\tfrac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu_0})' \mathbf{V_0^{-1}} (\boldsymbol{\beta}-\boldsymbol{\mu_0})\right) * (\sigma^2)^{\frac{-n-2\nu_0-2}{2}} \exp\left(-\frac{\tau_0}{\sigma^2}\right)$$
$$\exp\left(-\tfrac{1}{2\sigma^2}\left((\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta})\right)\right) * \prod_{i=1}^{n} \left(I(y_i = 0) I(y_i^* \le 0) + I(y_i = y_i^*) I(y_i^* > 0)\right) \tag{30}$$

The conditional posterior kernel for $\boldsymbol{\beta}$ now takes the following form:

$$p\left(\boldsymbol{\beta}\,|\,\sigma^2,\mathbf{y}^*,\mathbf{X}\right)\propto\exp\left(-\tfrac{1}{2}\left(\boldsymbol{\beta}-\boldsymbol{\mu_0}\right)'\mathbf{V_0^{-1}}\left(\boldsymbol{\beta}-\boldsymbol{\mu_0}\right)\right)*\exp\left(-\tfrac{1}{2\sigma^2}\left(\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)\right)\right) \tag{31}$$

This is equivalent to the conditional posterior for the basic linear regression model, and we can immediately derive the conditional posterior moments as:

$$\boldsymbol{\beta}\,|\,\sigma^2,\mathbf{y}^*,\mathbf{X}\sim n\left(\boldsymbol{\mu_1},\mathbf{V_1}\right)\ \text{ with }\quad \mathbf{V_1}=\left(\mathbf{V_0^{-1}}+\tfrac{1}{\sigma^2}\mathbf{X}'\mathbf{X}\right)^{-1}\text{ and }\quad \boldsymbol{\mu_1}=\mathbf{V_1}\left(\mathbf{V_0^{-1}}\boldsymbol{\mu_0}+\tfrac{1}{\sigma^2}\mathbf{X}'\mathbf{y}^*\right) \tag{32}$$

The conditional posterior kernel for σ^2 now also follows the basic linear regression model:

$$p\left(\sigma^2\,|\,\boldsymbol{\beta},\mathbf{y}^*,\mathbf{X}\right)\propto\left(\sigma^2\right)^{\frac{-n-2v_0-2}{2}}\exp\left(-\tfrac{1}{2\sigma^2}\left(2\tau_0+\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)\right)\right) \tag{33}$$

and thus

$$\begin{aligned}&\sigma^2\,|\,\boldsymbol{\beta},\mathbf{y}^*,\mathbf{X}\sim ig\left(v_1,\tau_1\right)\ \text{ with}\\ &v_1=\frac{2v_0+n}{2}\qquad\text{and}\qquad\tau_1=\frac{2\tau_0+\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)}{2}\end{aligned} \tag{34}$$

We now need to consider the conditional posterior kernel for $\mathbf{y}^*$. It will include all components of the joint posterior kernel in (13) that include $\mathbf{y}^*$, i.e

$$\begin{aligned}&p\left(\mathbf{y}^*\,|\,\boldsymbol{\beta},\sigma^2,\mathbf{y},\mathbf{X}\right)\propto\exp\left(-\tfrac{1}{2\sigma^2}\left(\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)'\left(\mathbf{y}^*-\mathbf{X}\boldsymbol{\beta}\right)\right)\right)*\\ &\prod_{i=1}^{n}\left(I\left(y_i=0\right)I\left(y_i^*\le 0\right)+I\left(y_i=y_i^*\right)I\left(y_i^*>0\right)\right)\end{aligned} \tag{35}$$

As before, more intuition is gained by writing this at the individual observation level, i.e.

$$p\left(y_i^*\,|\,\boldsymbol{\beta},\sigma^2,y_i,\mathbf{x_i}\right)\propto\exp\left(-\tfrac{1}{2\sigma^2}\left(y_i^*-\mathbf{x_i}'\boldsymbol{\beta}\right)^2\right)*\left(I\left(y_i=0\right)I\left(y_i^*\le 0\right)+I\left(y_i=y_i^*\right)I\left(y_i^*>0\right)\right) \tag{36}$$

which immediately implies

$$\begin{aligned}&p\left(y_i^*\,|\,\boldsymbol{\beta},\sigma^2,y_i=0,\mathbf{x_i}\right)\propto\exp\left(-\tfrac{1}{2\sigma^2}\left(y_i^*-\mathbf{x_i}'\boldsymbol{\beta}\right)^2\right)*I\left(y_i^*\le 0\right)\\ &p\left(y_i^*\,|\,\boldsymbol{\beta},\sigma^2,y_i=y_i^*,\mathbf{x_i}\right)=1\end{aligned} \tag{37}$$

Thus, we leave the $y_i^*>0$ cases untouched (since we are already observing their manifestations in the form of our data y_i), and draw the $y_i^*\le 0$ cases from the truncated-from-above-at-zero normal density:

$$y_i^* \mid \boldsymbol{\beta}, \sigma^2, y_i = 0, \mathbf{x_i} \sim tn\left(\mathbf{x}_i'\boldsymbol{\beta}, \sigma, -\infty, 0\right) \tag{38}$$

Note the presence of the standard deviation σ in this case for the truncated normal density.

The Tobit model with simulated data is implemented via Matlab scripts `mod5_tobit_data`, and `mod5_tobit`, and function `gs_tobit`.

Posterior quantities of interest in the Tobit model

The two primary posterior constructs of interest in the Tobit model are the expected value of the outcome variable under censoring, and the marginal effect of a regressor on this expectation (see KPT p. 220 for details):

$$\begin{aligned} p\left(E\left(y_p \mid \mathbf{x_p}, y_p^* > 0\right)\right) &= \int_{\boldsymbol{\theta}} \left(\mathbf{x_p}'\boldsymbol{\beta} + \sigma \frac{\phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)} \right) d\,\boldsymbol{\theta} \\ p\left(\frac{\partial E\left(y_p \mid \mathbf{x_p}\right)}{\partial x_{pj}} \mid y_p^* > 0 \right) &= \int_{\boldsymbol{\theta}} \beta_j \Phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right) d\,\boldsymbol{\theta} \end{aligned} \tag{39}$$

where $\boldsymbol{\theta} = \left[\boldsymbol{\beta}' \quad \sigma^2\right]'$ and as for the Probit model $\mathbf{x_p}$ is set to the sample mean for the derivation of marginal effects. If x_{pj} is an indicator variable, its marginal effect is expressed more meaningfully as

$$\begin{aligned} &p\left(E\left(y_p \mid y_p^* > 0, \mathbf{x}_{\mathbf{p},-j}, x_{pj} = 1\right) - E\left(y_p \mid y_p^* > 0, \mathbf{x}_{\mathbf{p},-j}, x_{pj} = 0\right)\right) = \\ &\int_{\boldsymbol{\theta}} \left(\left(\mathbf{x_p}'\boldsymbol{\beta} + \sigma \frac{\phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)} \mid x_{pj} = 1 \right) - \left(\mathbf{x_p}'\boldsymbol{\beta} + \sigma \frac{\phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{\mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)} \mid x_{pj} = 0 \right) \right) d\,\boldsymbol{\theta} \end{aligned} \tag{40}$$

See the Matlab script `mod5_tobit_adoption_predict` for an example.

Ignoring censoring

One might be tempted to simply ignore the censoring in the dependent variable and treat the threshold observations (usually "zeros") in the same way as all other outcome values. However, the resulting simple regression estimates are biased, with the bias increasing with the degree of censoring.

Running a basic regression implies that you are setting the censoring threshold to $-\infty$ (i.e. no censoring). This would imply

$$E\left(y_p \mid \mathbf{x_p}, y_p^* > -\infty\right) = \mathbf{x_p}'\boldsymbol{\beta} + \sigma \frac{\phi\left(\frac{\infty + \mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{\infty + \mathbf{x_p}'\boldsymbol{\beta}}{\sigma}\right)} = \mathbf{x_p}'\boldsymbol{\beta} + \frac{0}{1} = \mathbf{x_p}'\boldsymbol{\beta} \tag{41}$$

However, if the censoring threshold is not $-\infty$ but, say, some value *a*, this expectation (and thus $\boldsymbol{\beta}$) will be biased upwards, since the Inverse Mills Ratio (IMR) $\frac{\phi\left(\frac{a+\mathbf{x}_p'\boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{a+\mathbf{x}_p'\boldsymbol{\beta}}{\sigma}\right)}$ is always positive. The good news is that the IMR *de*creases in $\mathbf{x}_p'\boldsymbol{\beta}$. Thus, as the unconditional expectation $\mathbf{x}_p'\boldsymbol{\beta}$ increases, the bias diminishes. Intuitively, this means that if the unconditional mean is far to the right from the censoring threshold, the censoring effect is small and ignoring it is less damaging.

In the horse auction example, however, we have 55% censoring at zero, with many net bids close to this threshold. As illustrated in scripts `mod5_tobit_adoption_naive` and `mod5_tobit_adoption_naive_predict`, this leads to biased posteriors for coefficients and predictions. In addition, predictive densities now enter the negative realm, which obviously doesn't make sense from a policy perspective (i.e. if you want to answer the question "what kind of net bid range can a horse with certain features achieve"…)

The Chib Method for data-augmented models

A slightly modified version of the Chib method for deriving the marginal likelihood can also be implemented for models with data augmentation.

Building on the 3-parameter-block example, assume that instead of a third block our original GS used data augmentation, i.e. included draws of non-parameter elements $\mathbf{z}^r$, r=1…R from $p\left(\mathbf{z}\,|\,\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\mathbf{y}\right)$. In this case, the 3-block method can be applied directly, with $\mathbf{z}$ replacing $\boldsymbol{\theta}_3$ and omission of the last step, i.e. evaluation of $p\left(\overline{\boldsymbol{\theta}}_3\,|\,\overline{\boldsymbol{\theta}}_1,\overline{\boldsymbol{\theta}}_2,\mathbf{y}\right)$. Conceptually, we start again with

$$p\left(\overline{\boldsymbol{\theta}}_1,\overline{\boldsymbol{\theta}}_2\,|\,\mathbf{y}\right)=p\left(\overline{\boldsymbol{\theta}}_1\,|\,\mathbf{y}\right)p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{y}\right) \tag{42}$$

where now

$$\begin{aligned} p\left(\overline{\boldsymbol{\theta}}_1\,|\,\mathbf{y}\right)&=\iint p\left(\overline{\boldsymbol{\theta}}_1,\boldsymbol{\theta}_2,\mathbf{z}\,|\,\mathbf{y}\right)d\boldsymbol{\theta}_2 d\mathbf{z}=\iint p\left(\overline{\boldsymbol{\theta}}_1\,|\,\boldsymbol{\theta}_2,\mathbf{z},\mathbf{y}\right)p\left(\boldsymbol{\theta}_2,\mathbf{z}\,|\,\mathbf{y}\right)d\boldsymbol{\theta}_2 d\mathbf{z} \quad \text{and} \\ p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{y}\right)&=\int p\left(\overline{\boldsymbol{\theta}}_2,\mathbf{z}\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{y}\right)d\mathbf{z}=\int p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{z},\mathbf{y}\right)p\left(\mathbf{z}\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{y}\right)d\mathbf{z} \end{aligned} \tag{43}$$

We first approximate $p\left(\overline{\boldsymbol{\theta}}_1\,|\,\mathbf{y}\right)$ using output from the original GS, i.e.

$$p\left(\overline{\boldsymbol{\theta}}_1\,|\,\mathbf{y}\right)\approx\frac{1}{R}\sum_{r=1}^{R}p\left(\overline{\boldsymbol{\theta}}_1\,|\,\boldsymbol{\theta}_2^r,\mathbf{z}^r,\mathbf{y}\right) \tag{44}$$

We then use a second (separate GS) to draw sequentially from $p\left(\mathbf{z}\,|\,\overline{\boldsymbol{\theta}}_1,\boldsymbol{\theta}_2,\mathbf{y}\right)$ and $p\left(\boldsymbol{\theta}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{z},\mathbf{y}\right)$. This allows us to evaluate $p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{z}^s,\mathbf{y}\right)$ at each iteration and compute

$$p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{y}\right)\approx\frac{1}{S}\sum_{s=1}^{S}p\left(\overline{\boldsymbol{\theta}}_2\,|\,\overline{\boldsymbol{\theta}}_1,\mathbf{z}^s,\mathbf{y}\right) \tag{45}$$

The final result is then given as

$$\log p(\mathbf{y}) = \log p(\bar{\boldsymbol{\theta}}) + \log p(\mathbf{y} \mid \bar{\boldsymbol{\theta}}) - \left(\log p(\bar{\boldsymbol{\theta}}_1 \mid \mathbf{y}) + \log p(\bar{\boldsymbol{\theta}}_2 \mid \bar{\boldsymbol{\theta}}_1, \mathbf{y})\right). \quad (46)$$

There are a few important things to remember when applying the Chib method to data-augmented models:

1. By collecting the draws of $\mathbf{z}$ in the original GS, you can save time for the first step of the GS, which evaluates the marginal posterior of one of the parameters by averaging evaluations of the conditional posterior. This conditioning is over draws of the remaining parameters AND the augmented data. If you have saved your $\mathbf{z}$'s, you won't have to draw them again in this step.

2. When evaluating the *LHF* at the posterior mean of all parameters, we need to work with the *unconditional* likelihood $p(\mathbf{y} \mid \boldsymbol{\theta})$, not the conditional version $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{z})$. For more complicated *LHF*s you may need to work with numerical approximation of integrals or importance sampling (see Koop Ch. 4) to evaluate it.

3. Always save all components that enter the expression for the marginal likelihood separately: log(priors), logLHF, and log(posterior). That way, if the result looks "weird", you can pinpoint the problem faster to one of the components. Also, it is often illustrative to examine the relative magnitudes of these components for two competing models.

4. Remember that for any multi-block model the Chib routine will usually take quite a bit longer than the original Gibbs Sampler, as it needs to implement several rounds of reduced GS's with more and more parameters set at their posterior mean.

Examples:

Scripts `mod_5_probit_Fair_chib` and function `gs_probit_chib` implement this approach for the probit model using the full Fair (1978) data on extramarital affairs. This can be compared to scripts `mod_5_probit_Fair_nokids` and `mod_5_probit_Fair_nokids_chib`, which repeats the Fair analysis, but without the "kids" indicator. As can be seen from the Chib output, the restricted model receives more support in this case with a log BF of 2.9 (about 18 times more likely). Much of this is driven by avoiding a vague prior in the restricted model.

Similarly, `mod_5_tobit_adoption_chib` and function `gs_tobit_chib` implement the Chib method for the tobit model, using the horse adoption data. In this case, a model that omits the "training" variable, which had a strong, positive effect in the original specification, is found to be exponentially less likely than the full model (see scripts `mod5_tobit_adoption_notraining`, and `mod5_tobit_adoption_notraining_chib`).

References

Chib, Siddhartha and Bradley P. Carlin. 1999. "On MCMC sampling in hierarchical longitudinal models." *Statistics and Computing* 9, 17-26.

Fair, Ray C.. 1978. "A Theory on Extramarital Affairs." *Journal of Political Economy* 86, 45-61.

Dyk, David A. van and Xiao-Li Meng. 2001. "The Art of Data Augmentation." *Journal of Computational and Graphical Statistics* 10, 1-50.

Swamy, P.A.V.B. 1970. "Efficient Inference in a Random Coefficient Regression Model." *Econometrica* 38, 311-323.