

Lecture 24 MCMC Calibration of Log-AR(1) Model

Instead of presenting a general framework, we will illustrate basic strategies in MCMC algorithms via an example of Log-AR(1) model, which is a special case of *stochastic volatility* (SV) models. SV models are extensively studied in financial econometrics. Similar examples appear in Jun Liu's book: sections 1.6, 3.3, 4.5, 9.6.2, etc. There are many related terms in the literature, e.g., state-space models, filtering problems, hidden Markov models (HMM).

The Log-AR(1) model, given by

$$y_t = e^{h_t/2} \epsilon_t^{(1)} \tag{24.1}$$

$$h_t = \alpha + \beta h_{t-1} + \eta \epsilon_t^{(2)}, \tag{24.2}$$

$t = 1, \dots, T$, is a hierarchical model consisting of three layers: the observed return data $y = (y_1, \dots, y_T)$ (after removing the drift), the latent logarithmic volatility $h = (h_1, \dots, h_T)$, and the hyper-parameter $\theta = (\alpha, \beta, \eta)$. As a single source for randomness, the vectors $(\epsilon_t^{(1)}, \epsilon_t^{(2)})$, $t = 1, \dots, T$ are iid, so are the components $\epsilon_t^{(1)}$ and $\epsilon_t^{(2)}$ for each t , with the common distribution $N(0, 1)$. MCMC is a Bayesian inference scheme, which provides estimates of parameter θ and latent state h based on observed data y .

A simulation study before using real data contains two opposite directions to perform Monte Carlo simulation: a “top-down” path and a “bottom-up” path. The top-down path generates a synthetic return data set y by assigning initial values to θ , and simulating the volatility sequence h via (24.2), then simulating y via (24.1) conditioning on h . Note that the top-down path is a direct Monte Carlo practice, not MCMC. The bottom-up path aims at several conditional distributions: $p(\theta|y)$ as a basis for the Bayesian inference on θ , and $p(h|y)$ for generating realizations of h . Both $p(\theta|y)$ and $p(h|y)$ are marginals of the (joint) posterior distribution $p(\theta, h|y)$. However, $p(\theta, h|y)$ is not only analytically intractable, but also difficult to perform direct Monte Carlo simulation.

An ingenious idea, called *the Gibbs sampler*, is to alternate the conditional distributions $p(\theta|h, y)$ and $p(h|\theta, y)$ iteratively. Simulating from $p(\theta|h, y)$ is easy, but generating samples from $p(h|\theta, y)$ turns out to be much harder, due to the high dimensionality of h .

Question: why does the Gibbs sampler lead to $p(\theta, h|y)$ eventually?

To simulate from $p(h|\theta, y)$, we apply the Gibbs sampler again by cycling through a sequence of conditional distributions $p(h_t|h_{-t}, \theta, y)$, $t = 1, \dots, T$, where h_{-t} denotes the rest of h vector with h_t removed. Hence sampling from $p(h|\theta, y)$ is carried out by repeated single-site updating of h_t over different t . It is clear that h is a Markov chain. More

importantly, h also satisfies a “two-sided” Markov properties, i.e. the long vector h_{-t} in the conditioning can be replaced by (h_{t-1}, h_{t+1}) (the immediate past and future):

$$p(h_t|h_{-t}, \theta, y) = p(h_t|h_{t-1}, h_{t+1}, \theta, y).$$

Such local dependence is a key to make the computation simpler and faster. Moreover,

$$p(h_t|h_{t-1}, h_{t+1}, \theta, y) \propto p(y_t|h_t) p(h_t|h_{t-1}, \theta) p(h_{t+1}|h_t, \theta),$$

where the notation \propto indicates the specified conditional density up to a normalization constant. *Metropolis algorithms*, another special class of MCMC, turn out to be a right tool to generate MCMC samples without dealing with those complicated normalization constants. For Log-AR(1) model, the convergence of the proposed “Metropolis within Gibbs” can be verified by invoking the general theory given in Tierney (1994).

The MCMC algorithm proposed by Jacquier, Polson and Rossi (1994) (1999) consists of the following steps:

Step 1 Initialize h and θ .

Step 2 Sample h_t from $p(h_t|h_{t-1}, h_{t+1}, \theta, y)$ for $t = 1, \dots, T$.

Step 3 Sample θ from $p(\theta|h, y)$.

Step 4 Go back to Step 2 and continue.

It is usually referred to as a single-move sampler because the components of h are updated one by one sequentially. Each iteration of the MCMC consists of updating T components of h , and it will run a large number N iterations before the sampler terminates. Step 3 follows the standard Bayesian linear model theory [Gelman et al. (1998), Chapter 8]. Starting from a non-informative conjugate prior for θ :

$$p(\alpha, \beta, \eta) \propto \eta^{-2},$$

and observing that θ and y are conditionally independent given h , the posterior for θ given h is given by

$$p(\theta|h) = \mathcal{N}_2(\mu, (H^\top H)^{-1}\eta^2) IG((T-2)/2, Ts^2/2), \quad (24.3)$$

where \mathcal{N}_2 denotes the bivariate normal distribution, and IG the inverse gamma distribution; $\mu = (H^\top H)^{-1}H^\top h_{2,T}$ with $h_{2,T} = (h_2, \dots, h_T)^\top$ (the notation A^\top represents the transpose of matrix A), H is an $(T-1) \times 2$ matrix whose row t is $(1, h_t)$, $t = 1, \dots, T-1$, and Ts^2 is the standard sum of squared errors. In Step 2, we need to simulate, for each $t = 1, \dots, T$, from the single-site conditional distribution $p(h_t|h_{t-1}, h_{t+1}, \theta, y)$, expressed as

$$p(h_t|h_{t-1}, h_{t+1}, \theta, y) \propto \exp \left\{ -\frac{1}{2} \left[h_t + y_t^2/e^{h_t} + (h_t - m_t)^2/b^2 \right] \right\}, \quad (24.4)$$

where $m_t = [\alpha(1-\beta) + \beta(h_{t-1} + h_{t+1})]/(1+\beta^2)$ and $b^2 = \eta^2/(1+\beta^2)$. To apply the Metropolis algorithm locally at each t to (24.4), denote the updated value of h after iteration i by $h^{(i)}$ and assume that the current value within iteration $i + 1$ is $(h_1^{(i+1)}, \dots, h_{t-1}^{(i+1)}, h_t^{(i)}, \dots, h_T^{(i)})$. First, simulate $h_t^{(i+1)}$ from a *proposal density* $q(h_t^{(i+1)} | h_t^{(i)}) \sim N(h_t^{(i)}, c_0 \hat{\eta})$, where $\hat{\eta}$ is an estimate of η obtained so far and c_0 is a constant chosen judiciously. Note that this proposal kernel is symmetric, i.e. $q(z|z') = q(z'|z)$ for all $z, z' \in \mathcal{R}$. Next, $h_t^{(i+1)}$ sampled from the above q is accepted with probability

$$\min \left(\frac{p(h_t^{(i+1)} | h_{t-1}^{(i+1)}, h_{t+1}^{(i)}, \theta^{(i)}, y)}{p(h_t^{(i)} | h_{t-1}^{(i+1)}, h_{t+1}^{(i)}, \theta^{(i)}, y)}, 1 \right).$$

References

- Gelman, A., Carlin, J.B., Stern, H.S. and Rubin, D.B. (1998). *Bayesian Data Analysis*. Chapman & Hall.
- Jacquier, E., Polson, N.G. and Rossi, P.E. (1994). Bayesian analysis of stochastic volatility models (with discussion). *Journal of Business and Economic Statistics* **12**, 371-417.
- Jacquier, E., Polson, N.G. and Rossi, P.E. (1999). Stochastic volatility: univariate and multivariate extensions. Preprint.
- Tierney, L. (1994). Markov chains for exploring posterior distributions (with discussion). *Annals of Statistics* **22**, 1701-1786.