Lesson 5: Gibbs sampling

Lesson 5.1

So far, we have demonstrated MCMC for a single parameter. What if we seek the posterior distribution of multiple parameters, and that posterior distribution does not have a standard form? One option is to perform Metropolis-Hastings (M-H) by sampling candidates for all parameters at once, and accepting or rejecting all of those candidates together. While this is possible, it can get complicated. Another (simpler) option is to sample the parameters one at a time.

As a simple example, suppose we have a joint posterior distribution for two parameters θ and φ, written $p(\theta, \varphi \mid y) \propto g(\theta, \varphi)$. If we knew the value of φ, then we would just draw a candidate for θ and use $g(\theta, \varphi)$ to compute our Metropolis-Hastings ratio, and possibly accept the candidate. Before moving on to the next iteration, if we don't know φ, then we can perform a similar update for it. Draw a candidate for φ using some proposal distribution and again use $g(\theta, \varphi)$ to compute our Metropolis-Hastings ratio. Here we pretend we know the value of θ by substituting its current iteration from the Markov chain. Once we've drawn for both θ and φ, that completes one iteration and we begin the next iteration by drawing a new θ. In other words, we're just going back and forth, updating the parameters one at a time, plugging the current value of the other parameter into $g(\theta, \varphi)$.

This idea of one-at-a-time updates is used in what we call *Gibbs sampling*, which also produces a stationary Markov chain (whose stationary distribution is the posterior). If you recall, this is the namesake of `JAGS`, "just another Gibbs sampler."

Full conditional distributions

Before describing the full Gibbs sampling algorithm, there's one more thing we can do. Using the chain rule of probability, we have $p(\theta, \varphi \mid y) = p(\theta \mid \varphi, y) \cdot p(\varphi \mid y)$. Notice that the only difference between $p(\theta, \varphi \mid y)$ and $p(\theta \mid \varphi, y)$ is multiplication by a factor that doesn't involve θ. Since the $g(\theta, \varphi)$ function above, when viewed as a function of θ is proportional to both these expressions, we might as well have replaced it with $p(\theta \mid \varphi, y)$ in our update for θ.

This distribution $p(\theta \mid \varphi, y)$ is called the full conditional distribution for θ. Why use it instead of $g(\theta, \varphi)$? In some cases, the full conditional distribution is a standard distribution we know how to sample. If that happens, we no longer need to draw a candidate and decide whether to accept it. In fact, if we treat the full conditional distribution as a candidate proposal distribution, the resulting Metropolis-Hastings acceptance probability becomes exactly 1.

Gibbs samplers require a little more work up front because you need to find the full conditional distribution for each parameter. The good news is that all full conditional distributions have the same starting point: the full joint posterior distribution. Using the example above, we have

$$p(\theta \mid \varphi, y) \propto p(\theta, \varphi \mid y)$$

where we simply now treat φ as a known number. Likewise, the other full conditional is $p(\varphi \mid \theta, y) \propto p(\theta, \varphi \mid y)$ where here, we consider θ to be a known number. We always start with the full posterior distribution. Thus, the process of finding full conditional distributions is the same as finding the posterior distribution of each parameter, pretending that all other parameters are known.

Gibbs sampler

The idea of Gibbs sampling is that we can update multiple parameters by sampling just one parameter at a time, cycling through all parameters and repeating. To perform the update for one particular parameter, we substitute in the current values of all other parameters.

Here is the algorithm. Suppose we have a joint posterior distribution for two parameters θ and φ, written $p(\theta, \varphi \mid y)$. If we can find the distribution of each parameter at a time, i.e., $p(\theta \mid \varphi, y)$ and $p(\varphi \mid \theta, y)$, then we can take turns sampling these distributions like so:

1. Using φ_{i-1}, draw θ_i from $p(\theta \mid \varphi = \varphi_{i-1}, y)$.
2. Using θ_i, draw φ_i from $p(\varphi \mid \theta = \theta_i, y)$.

Together, steps 1 and 2 complete one cycle of the Gibbs sampler and produce the draw for (θ_i, φ_i) in one iteration of a MCMC sampler. If there are more than two parameters, we can handle that also. One Gibbs cycle would include an update for each of the parameters.

In the following segments, we will provide a concrete example of finding full conditional distributions and constructing a Gibbs sampler.

Lesson 5.2

Normal likelihood, unknown mean and variance

Let's return to the example at the end of Lesson 2 where we have normal likelihood with unknown mean and unknown variance. The model is

$$\begin{aligned} y_i \mid \mu, \sigma^2 &\overset{\text{iid}}{\sim} N(\mu, \sigma^2), \quad i = 1, \dots, n \\ \mu &\sim N(\mu_0, \sigma_0^2) \\ \sigma^2 &\sim IG(\nu_0, \beta_0) \, . \end{aligned}$$

We chose a normal prior for μ because, in the case where σ^2 is known, the normal is the conjugate prior for μ. Likewise, in the case where μ is known, the inverse-gamma is the conjugate prior for σ^2. This will give us convenient full conditional distributions in a Gibbs sampler.

Let's first work out the form of the full posterior distribution. When we begin analyzing data, the `JAGS` software will complete this step for us. However, it is extremely valuable to see and understand how this works.

$$\begin{aligned} p(\mu, \sigma^2 \mid y_1, y_2, \dots, y_n) &\propto p(y_1, y_2, \dots, y_n \mid \mu, \sigma^2) p(\mu) p(\sigma^2) \\ &= \prod_{i=1}^{n} N(y_i \mid \mu, \sigma^2) \times N(\mu \mid \mu_0, \sigma_0^2) \times IG(\sigma^2 \mid \nu_0, \beta_0) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y_i - \mu)^2}{2\sigma^2} \right] \times \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left[-\frac{(\mu - \mu_0)^2}{2\sigma_0^2} \right] \times \frac{\beta_0^{\nu_0}}{\Gamma(\nu_0)} (\sigma^2)^{-(\nu_0+1)} \exp\left[-\frac{\beta_0}{\sigma^2} \right] I_{\sigma^2>0}(\sigma^2) \\ &\propto (\sigma^2)^{-n/2} \exp\left[-\frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2\sigma^2} \right] \exp\left[-\frac{(\mu - \mu_0)^2}{2\sigma_0^2} \right] (\sigma^2)^{-(\nu_0+1)} \exp\left[-\frac{\beta_0}{\sigma^2} \right] I_{\sigma^2>0}(\sigma^2) \end{aligned}$$

From here, it is easy to continue on to find the two full conditional distributions we need. First let's look at μ, assuming σ^2 is known (in which case it becomes a constant and is absorbed into the normalizing constant):

$$\begin{aligned} p(\mu \mid \sigma^2, y_1, \dots, y_n) &\propto p(\mu, \sigma^2 \mid y_1, \dots, y_n) \\ &\propto \exp\left[-\frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2\sigma^2} \right] \exp\left[-\frac{(\mu - \mu_0)^2}{2\sigma_0^2} \right] \\ &\propto \exp\left[-\frac{1}{2} \left(\frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2\sigma^2} + \frac{(\mu - \mu_0)^2}{2\sigma_0^2} \right) \right] \\ &\propto N\left(\mu \mid \frac{n\bar{y}/\sigma^2 + \mu_0/\sigma_0^2}{n/\sigma^2 + 1/\sigma_0^2}, \frac{1}{n/\sigma^2 + 1/\sigma_0^2} \right) , \end{aligned}$$

which we derived in the supplementary material of the last course. So, given the data and σ^2, μ follows this normal distribution.

Now let's look at σ^2, assuming μ is known:

$$\begin{aligned} p(\sigma^2 \mid \mu, y_1, \dots, y_n) &\propto p(\mu, \sigma^2 \mid y_1, \dots, y_n) \\ &\propto (\sigma^2)^{-n/2} \exp\left[-\frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2\sigma^2} \right] (\sigma^2)^{-(\nu_0+1)} \exp\left[-\frac{\beta_0}{\sigma^2} \right] I_{\sigma^2>0}(\sigma^2) \\ &\propto (\sigma^2)^{-(\nu_0 + n/2 + 1)} \exp\left[-\frac{1}{\sigma^2} \left(\beta_0 + \frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2} \right) \right] I_{\sigma^2>0}(\sigma^2) \\ &\propto IG\left(\sigma^2 \mid \nu_0 + \frac{n}{2}, \beta_0 + \frac{\sum_{i=1}^{n} (y_i - \mu)^2}{2} \right) . \end{aligned}$$

These two distributions provide the basis of a Gibbs sampler to simulate from a Markov chain whose stationary distribution is the full posterior of both μ and σ^2. We simply alternate draws between these two parameters, using the most recent draw of one parameter to update the other.

We will do this in `R` in the next segment.

Lesson 5.3

Gibbs sampler in R

To implement the Gibbs sampler we just described, let's return to our running example where the data are the percent change in total personnel from last year to this year for $n = 10$ companies. We'll still use a normal likelihood, but now we'll relax the assumption that we know the variance of growth between companies, σ^2, and estimate that variance. Instead of the t prior from earlier, we will use the conditionally conjugate priors, normal for μ and inverse-gamma for σ^2.

The first step will be to write functions to simulate from the full conditional distributions we derived in the previous segment. The full conditional for μ, given σ^2 and data is

$$N\left(\mu \mid \frac{n\bar{y}/\sigma^2 + \mu_0/\sigma_0^2}{n/\sigma^2 + 1/\sigma_0^2}, \frac{1}{n/\sigma^2 + 1/\sigma_0^2}\right)$$

```
update_mu = function(n, ybar, sig2, mu_0, sig2_0) {
  sig2_1 = 1.0 / (n / sig2 + 1.0 / sig2_0)
  mu_1 = sig2_1 * (n * ybar / sig2 + mu_0 / sig2_0)
  rnorm(n=1, mean=mu_1, sd=sqrt(sig2_1))
}
```

The full conditional for σ^2 given μ and data is

$$IG\left(\sigma^2 \mid \nu_0 + \frac{n}{2}, \beta_0 + \frac{\sum_{i=1}^{n}(y_i - \mu)^2}{2}\right)$$

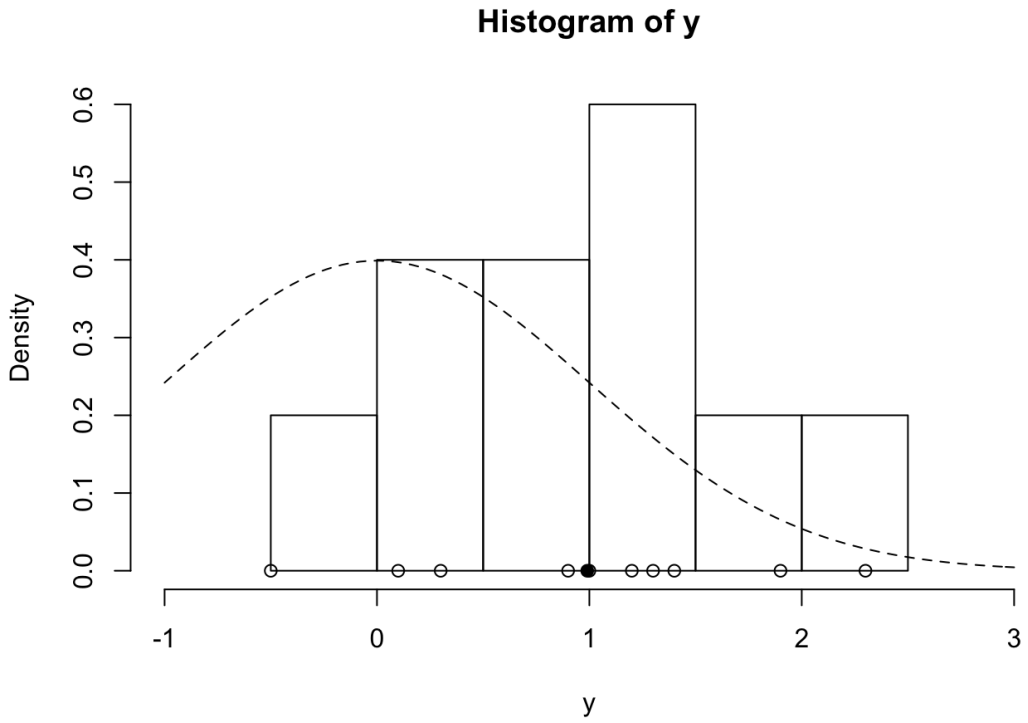
```
update_sig2 = function(n, y, mu, nu_0, beta_0) {
  nu_1 = nu_0 + n / 2.0
  sumsq = sum( (y - mu)^2 ) # vectorized
  beta_1 = beta_0 + sumsq / 2.0
  out_gamma = rgamma(n=1, shape=nu_1, rate=beta_1) # rate for gamma is shape for inv-gamma
  1.0 / out_gamma # reciprocal of a gamma random variable is distributed inv-gamma
}
```

With functions for drawing from the full conditionals, we are ready to write a function to perform Gibbs sampling.

```
gibbs = function(y, n_iter, init, prior) {
  ybar = mean(y)
  n = length(y)

  ## initialize
  mu_out = numeric(n_iter)
  sig2_out = numeric(n_iter)

  mu_now = init$mu

  ## Gibbs sampler
  for (i in 1:n_iter) {
    sig2_now = update_sig2(n=n, y=y, mu=mu_now, nu_0=prior$nu_0, beta_0=prior$beta_0)
    mu_now = update_mu(n=n, ybar=ybar, sig2=sig2_now, mu_0=prior$mu_0, sig2_0=prior$sig2_0)

    sig2_out[i] = sig2_now
    mu_out[i] = mu_now
  }

  cbind(mu=mu_out, sig2=sig2_out)
}
```

Now we are ready to set up the problem in R .

```
y = c(1.2, 1.4, -0.5, 0.3, 0.9, 2.3, 1.0, 0.1, 1.3, 1.9)
ybar = mean(y)
n = length(y)

## prior
prior = list()
prior$mu_0 = 0.0
prior$sig2_0 = 1.0
prior$n_0 = 2.0 # prior effective sample size for sig2
prior$s2_0 = 1.0 # prior point estimate for sig2
prior$nu_0 = prior$n_0 / 2.0 # prior parameter for inverse-gamma
prior$beta_0 = prior$n_0 * prior$s2_0 / 2.0 # prior parameter for inverse-gamma

hist(y, freq=FALSE, xlim=c(-1.0, 3.0)) # histogram of the data
curve(dnorm(x=x, mean=prior$mu_0, sd=sqrt(prior$sig2_0)), lty=2, add=TRUE) # prior for mu
points(y, rep(0,n), pch=1) # individual data points
points(ybar, 0, pch=19) # sample mean
```



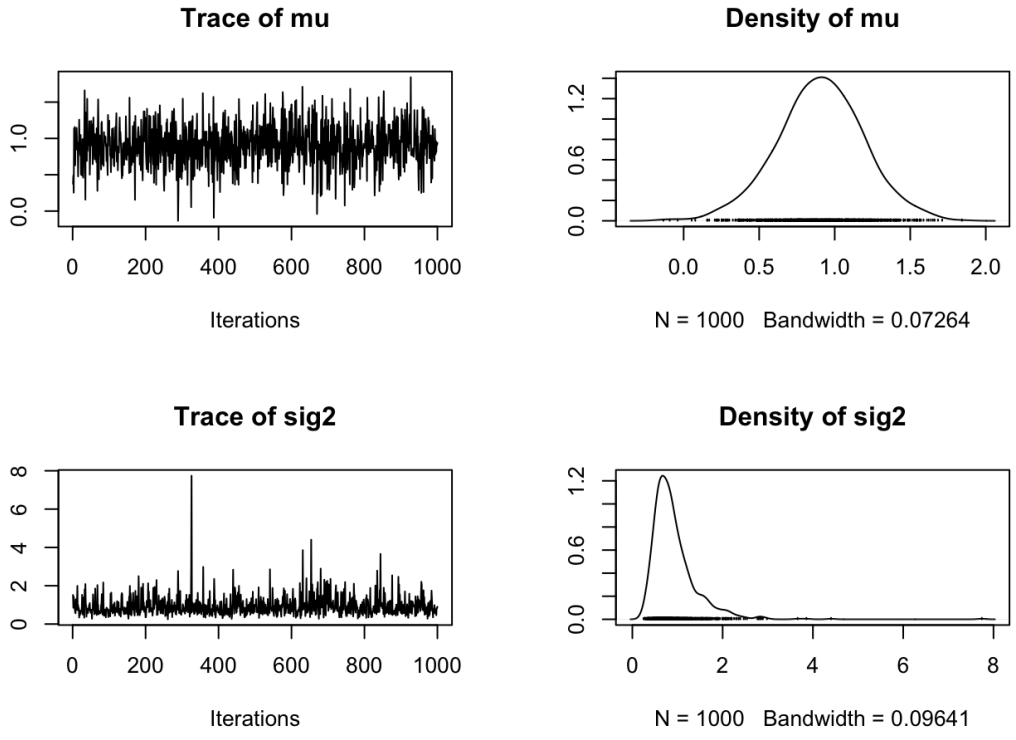
Finally, we can initialize and run the sampler!

```
set.seed(53)

init = list()
init$mu = 0.0

post = gibbs(y=y, n_iter=1e3, init=init, prior=prior)
```

```
head(post)
```

```
##             mu      sig2
## [1,] 0.3746992 1.5179144
## [2,] 0.4900277 0.8532821
## [3,] 0.2536817 1.4325174
## [4,] 1.1378504 1.2337821
## [5,] 1.0016641 0.8409815
## [6,] 1.1576873 0.7926196
```

```
library("coda")
plot(as.mcmc(post))
```




```
summary(as.mcmc(post))
```

```
##
## Iterations = 1:1000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 1000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##        Mean     SD Naive SE Time-series SE
## mu   0.9051 0.2868  0.00907        0.00907
## sig2 0.9282 0.5177  0.01637        0.01810
##
## 2. Quantiles for each variable:
##
##        2.5%    25%    50%   75% 97.5%
## mu   0.3024 0.7244 0.9089 1.090 1.481
## sig2 0.3577 0.6084 0.8188 1.094 2.141
```

As with the Metropolis-Hastings example, these chains appear to have converged. In the next lesson, we will discuss convergence in more detail.