**ISyE 6420**
Fall 2019
Brani Vidakovic

# An Example of Gibbs Sampler

**Gibbs Sampler.** Suppose that $\boldsymbol{\theta} = (\theta_1, \dots, \theta_p)$ is a multidimensional parameter of interest. Suppose that we can simulate from the conditional densities $\pi(\theta_i|\boldsymbol{\theta}_{-i})$, where $\boldsymbol{\theta}_{-i}$ denotes the parameter vector $\boldsymbol{\theta}$ without $i$th component. If, in the process of simulation, the current state of $\boldsymbol{\theta}$ is $\boldsymbol{\theta}^n = (\theta_1^n, \theta_2^n, \dots, \theta_p^n)$, the Gibbs sampler produces $\boldsymbol{\theta}^{n+1}$ in the following way:

Draw $\theta_1^{n+1}$ from $\pi(\theta_1|\theta_2^n, \theta_3^n, \dots, \theta_p^n)$
Draw $\theta_2^{n+1}$ from $\pi(\theta_2|\theta_1^{n+1}, \theta_3^n, \dots, \theta_p^n)$
Draw $\theta_3^{n+1}$ from $\pi(\theta_3|\theta_1^{n+1}, \theta_2^{n+1}, \theta_4^n, \dots, \theta_p^n)$
...
Draw $\theta_{p-1}^{n+1}$ from $\pi(\theta_{p-1}|\theta_1^{n+1}, \theta_2^{n+1}, \dots, \theta_{p-2}^{n+1}, \theta_p^n)$
Draw $\theta_p^{n+1}$ from $\pi(\theta_p|\theta_1^{n+1}, \theta_2^{n+1}, \dots, \theta_{p-1}^{n+1})$

Here, we assumed a fixed updating order. This may not always be the case, since it is possible to generalize the Gibbs Sampler in a number of ways.

**Finding the Full Conditionals.** The full conditionals, needed for implementation of the Gibbs sampler are *conceptually* easy to find. From the joint distribution of all variables, only expressions that contain the particular variable are entering to the conditional distribution. The difficulty is (as always) in finding normalizing constants.

Suppose $\boldsymbol{\theta} = (\boldsymbol{\theta}_s, \boldsymbol{\theta}_{-s})$ and we are interested in the conditional for $\boldsymbol{\theta}_s$, given $\boldsymbol{\theta}_{-s}$. The full conditional is

$$\pi(\boldsymbol{\theta}_s|\boldsymbol{\theta}_{-s}) = \frac{\pi(\boldsymbol{\theta}_s, \boldsymbol{\theta}_{-s})}{\int \pi(\boldsymbol{\theta}_s, \boldsymbol{\theta}_{-s}) d\boldsymbol{\theta}_s} \propto \pi(\boldsymbol{\theta}_s, \boldsymbol{\theta}_{-s}).$$

**Example.** We illustrate Gibbs sampler and finding the full conditionals in the model:

$$\begin{aligned} Y_1, Y_2, \dots, Y_n &\sim \mathcal{N}(\mu, 1/\tau) \\ \mu &\sim \mathcal{N}(\mu_0, 1/\tau_0) \\ \tau &\sim \mathcal{G}a(a, b), \end{aligned}$$

where $\tau, \tau_0$ are precision parameters (reciprocals of variances), and $b$ is a rate parameter. Note that in the above, the normal distribution is parametrized with mean and variance, as usual.

The joint distribution is

$$
\begin{aligned}
f(y, \mu, \tau) &= \left\{ \prod_{i=1}^{n} f(y_i|\mu, \tau) \right\} \pi(\mu)\pi(\tau) \\
&\propto \tau^{n/2} \exp\left\{ -\tau/2 \sum_{i=1}^{n} (y_i - \mu)^2 \right\} \ \exp\{-(\mu - \mu_0)^2/2\} \ \tau^{a-1} \exp\{-b\tau\}.
\end{aligned}
$$

To find the full conditional for $\mu$ we select the terms from $f(y, \mu, \tau)$ that contain $\mu$ and normalize. Indeed,

$$
\begin{aligned}
\pi(\mu|\tau, y) &= \frac{\pi(\mu, \tau|y)}{\pi(\tau|y)} \\
&= \frac{\pi(\mu, \tau, y)}{\pi(\tau, y)} \propto \pi(\mu, \tau, y).
\end{aligned}
$$

Thus,

$$
\begin{aligned}
\pi(\mu|\tau, y) &\propto \exp\left\{ -\frac{\tau}{2} \sum_{i=1}^{n} (y_i - \mu)^2 \right\} \exp\left\{ -\frac{\tau_0}{2} (\mu - \mu_0)^2 \right\} \\
&\propto \exp\left\{ -\frac{1}{2} (\tau_0 + n\tau) \left( \mu - \frac{\tau \sum y_i + \tau_0 \mu_0}{\tau_0 + n\tau} \right)^2 \right\},
\end{aligned}
$$

which a kernel of normal $\mathcal{N}(\frac{\tau \sum y_i + \tau_0\mu_0}{\tau_0 + n\tau}, \frac{1}{\tau_0 + n\tau})$ distribution. Similarly,

$$
\begin{aligned}
\pi(\tau|\mu, y) &\propto \tau^{n/2} \exp\left\{ -\tau/2 \sum_{i=1}^{n} (y_i - \mu)^2 \right\} \ \tau^{a-1} \exp\{-b\tau\} \\
&= \tau^{n/2+a-1} \exp\left\{ -\tau \left[ b + \frac{1}{2} \sum_{i=1}^{n} (y_i - \mu)^2 \right] \right\},
\end{aligned}
$$

which is a kernel of gamma $\mathcal{G}a(a + n/2, b + \frac{1}{2} \sum_{i=1}^{n} (y_i - \mu)^2)$ distribution.

**Numerical Example:** As a specific example we simulate $n = 100$ observations from normal $\mathcal{N}(1, 2^2)$ distribution as our sample, and start a Gibbs sampler with initials $\mu = 1/2$ and $\tau = 1/2$.

The prior on $\mu$ is normal $\mathcal{N}(\mu_0, 1/\tau_0)$ for $\mu_0 = 1/2$ and $\tau_0 = 1/100$. The prior on precision $\tau$ is gamma $\mathcal{G}a(a, b)$ for $a = 1/2$ and $b = 2$. We simulate $NN = 10000$ observations from the posteriors and burn in first 200 values. What is the MCMC estimator for $\mu$ and $\tau$?

Below are versions of code in Octave, Python, and R.

```
close all
clear all
 randn("state",1)
```

```
  randg("state",1)
  n=100;
  y = 2 * randn(1,n) + 1;
%-------------------------------------------
NN = 10000;
mus = []; taus = [];
sumdata = sum(y);
%hyperparameters
mu0=0.5; tau0 = 1/100;
a= 1/2; b= 2;
% start, initial values
mu = 0.5;   tau=0.5;
for i = 1 : NN
  newmu  = sqrt(1/(tau0+n*tau)) * randn + (tau * sumdata+tau0*mu0)/(tau0+n*tau);
  par   = b+1/2 * sum ( (y - newmu).^2);
  newtau = gamrnd(a + n/2, 1/par); %par is rate
  mus = [mus newmu];
  taus = [taus newtau];
  mu=newmu;
  tau=newtau;
end

burn =200;
mus = mus(burn+1:end);
taus=taus(burn+1:end);

mean(mus)
mean(taus)
figure(1)
subplot(211)
hist(mus, 40)
subplot(212)
hist(taus, 40)
```

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(100)
n=100
y = np.random.normal(1,2,n) #construct the sample of size 100
#-------------------------------------------
NN = 10000;
mus = np.array([])
taus = np.array([])
sumdata = np.sum(y)
#hyperparameters
mu0=0.5
tau0 = 1/100
a= 1/2
b= 2
# start, initial values
```

```
mu =  0.5
tau = 0.5
for i in range(NN):
  newmu  = np.random.normal((tau * sumdata+tau0*mu0)/(tau0+n*tau),
                           np.sqrt(1/(tau0+n*tau)))
  par   = b+1/2 * np.sum ( (y -  newmu)**2)
  newtau = np.random.gamma(a + n/2, 1/par); #par is rate
  mus = np.append(mus, newmu)
  taus = np.append(taus, newtau)
  mu = newmu
  tau = newtau

burn =200
mus  = mus[burn:NN]
taus = taus[burn:NN]

print(np.mean(mus))      #0.7933565847999285
print(np.mean(taus))      #0.2629318615281096
plt.subplot(2,1,1)
plt.hist(mus, 40)
plt.subplot(2,1,2)
plt.hist(taus, 40)
plt.show()
```

```
n <-100
set.seed(2)
y <- 2 * rnorm(100) + 1
#-----------------------------------------
NN <- 10000
mus <- c()
taus <- c()
sumdata <- sum(y)
#hyperparameters
mu0 <- 0.5
tau0 <- 1/100
a <- 1/2
b <-  2
# start, initial values
mu <- 0.5
tau <- 0.5
for (i in 1 : NN){
   newmu <- rnorm(1, (tau * sumdata+tau0*mu0)/(tau0+n*tau),
                 sqrt(1/(tau0+n*tau)) )
   rat  <- b + 1/2 * sum ( (y - newmu)^2)
   newtau <- rgamma(1,shape=a + n/2,
                   rate=rat)
   mus <-  c(mus, newmu)
   taus <- c(taus, newtau)
   mu <- newmu
```

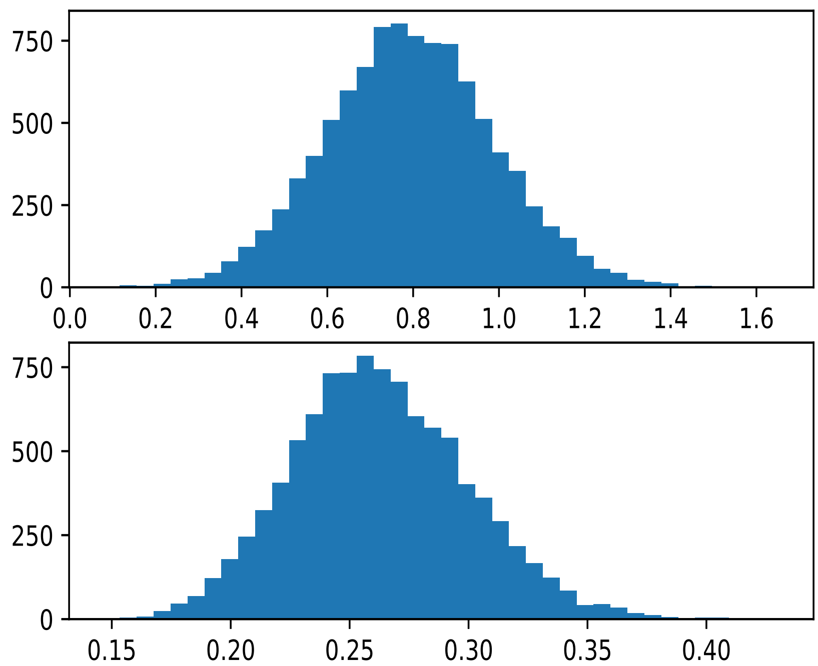


Figure 1: Figure from Python code. Top: histogram of $\mu$s, bottom: histogram of $\tau$s.

```
   tau<- newtau
}

burn <- 200
mus <- mus[burn:NN]
taus <- taus[burn:NN]

mean(mus)
  #[1] 0.9404001

mean(taus)
  #[1] 0.186471

par(mfrow=c(1,2))
hist(mus, 40)
hist(taus, 40)
```