Project 2 for Bayesian Inference: SOLUTIONS
Spring 2009

1. Let $\pi(x, y)$ denote a bivariate density which is uniform over the region bounded by the lines $x = 0$, $x = 1$, $x + y = 1$, and $x + y = 2$.

   (a) Identify the conditional densities $\pi(y|x)$, $\pi(x|y)$ for $0 < x < 1$ and $0 < y < 2$. (A sketch of the region will help). Design and implement a Gibbs sample to sample from π, by drawing from these conditional densities.

   (b) Generate histogram estimates of the marginal densities $\pi(x)$ and $\pi(y)$, and verify that your algorithm produces estimates that agree with the theoretical marginal densities (which you will have to work out!)

   (c) Suppose that you used a similar algorithm to sample from a density that was uniform on the region bounded by $x = 0$, $x = 1$, $x + y = 1$, and $x + y = 1.1$. By considering the shape of this region, suggest why your Gibbs sampler would not explore the target density efficiently.

ANSWER

(a) We are given a uniform density $\pi(x, y)$ on the set

$$A := \{(x, y) \in \mathbb{R} \times \mathbb{R} : \ 0 \le x \le 1, \ y \ge 0, \ 1 \le x + y \le 2\}.$$

It is easy to see that, for each $0 \le x \le 1$, the conditional density $\pi_{21}(y|x)$ is uniform on the set

$$\begin{aligned} A_x &:= \{y \in \mathbb{R} : \ (x, y) \in A\} \\ &= \{y \in \mathbb{R} : \ 1 - x \le y \le 2 - x\} = [1 - x, 2 - x]. \end{aligned}$$

Similarly, for each $0 \le y \le 2$, the conditional density $\pi_{12}(x|y)$ is uniform on the set

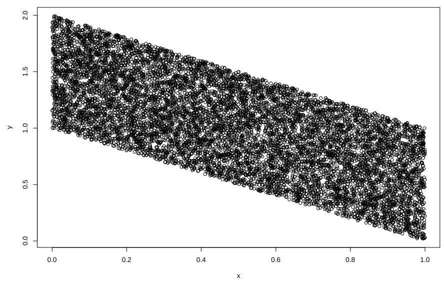
$$\begin{aligned} A_y &:= \{x \in \mathbb{R} : \ (x, y) \in A\} \\ &= \{x \in \mathbb{R} : \ 0 \le x \le 1, \ 1 - y \le x \le 2 - y\} \\ &= \{x \in \mathbb{R} : \ \max(0, 1 - y) \le x \le \min(1, 2 - y)\} \\ &= [\max(0, 1 - y), \ \min(1, 2 - y)]. \end{aligned}$$

We obtain one sample from the UNIFORM$[a, b]$ law by using the command `runif(1,a,b)` in R. The code is:

```
N=10000;
x=c(1:N); y=c(1:N);x[1]=1/2; y[1]=1
for(i in 1:(N-1)){
x[i+1] = runif(1, max(0,1-y[i]), min(1,2-y[i]))
y[i+1] = runif(1, 1-x[i+1], 2-x[i+1])   }
```

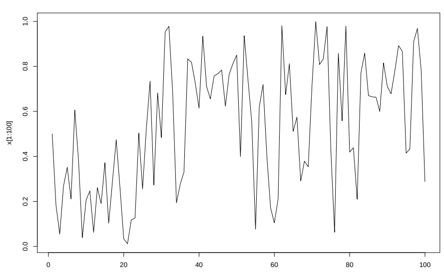
Here is a scatter plot of y vs. x:

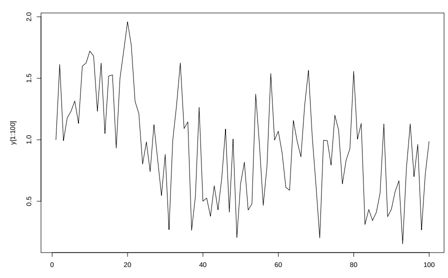
```
plot(x,y)
dev.copy(postscript,'figs/scatterxy.ps')
dev.off()
```



Let's plot the first 100 values of x and y:

```
plot(x[1:100],type='l')
dev.copy(postscript,'figs/xplot100.ps'); dev.off()
plot(y[1:100],type='l')
dev.copy(postscript,'figs/yplot100.ps'); dev.off()
```





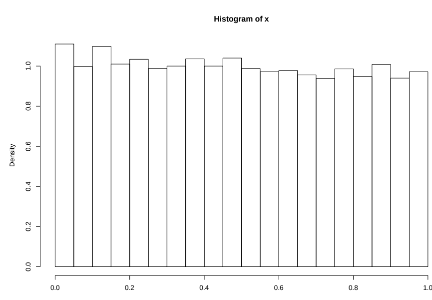
(b) We compute the marginal densities as follows:

$$
\begin{aligned}
\pi_1(x) &= \int_{\mathbb{R}} \pi(x,y)dy \\
&= \mathbb{1}(0 \le x \le 1) \int_{\mathbb{R}} \mathbb{1}(1-x \le y \le 2-x)dy \\
&= \mathbb{1}(0 \le x \le 1) \\
\pi_2(y) &= \int_{\mathbb{R}} \pi(x,y)dx \\
&= \int_{\mathbb{R}} \mathbb{1}(0 \le x \le 1,\ 1-y \le x \le 2-y)dx \\
&= \int_{\mathbb{R}} \mathbb{1}(\max(0,1-y) \le x \le \min(1,2-y))dx \\
&= \max\big(0, \min(1,2-y) - \max(0,1-y)\big).
\end{aligned}
$$

Thus π_1 is uniform between 0 and 1, whereas π_2 is a "tent" map. We plot the histogram for x:

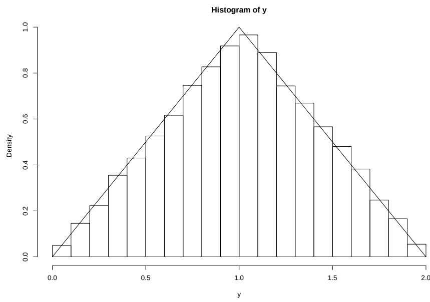
```
hist(x,probability=1)
dev.copy(postscript,'figs/histx.ps')
dev.off()
```

It looks fairly "OK".



We next plot the histogram for y against its theoretical marginal density:

```
p2=function(t){pmax(0, pmin(1,2-t)-pmax(0,1-t))}
hist(y,probability=1)
plot(p2,0,2,add=TRUE)
dev.copy(postscript,'figs/histy.ps')
dev.off()
```



(c) Suppose now that the $\pi(x, y)$ is uniform on

$$A := \{(x, y) \in \mathbb{R} \times \mathbb{R} : \ 0 \le x \le 1, \ y \ge 0, \ 1 \le x + y \le 1.1\}.$$

Then for each $0 \le x \le 1$, the conditional density $\pi_{21}(y|x)$ is uniform on the set

$$A_x = [1 - x, \ 1.1 - x].$$

For each $0 \le y \le 1.1$, the conditional density $\pi_{12}(x|y)$ is uniform on the set

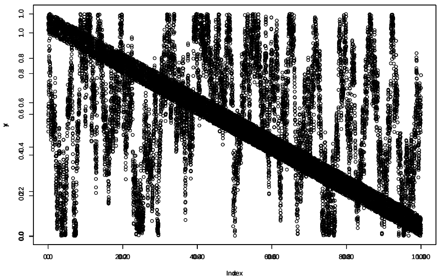
$$A_y = [\max(0, 1 - y), \ \min(1, 1.1 - y)].$$

The code is:

```
N=10000;
x=c(1:N); y=c(1:N);x[1]=0.51; y[1]=0.51
for(i in 1:(N-1)){
x[i+1] = runif(1, max(0,1-y[i]), min(1,1.1-y[i]))
y[i+1] = runif(1, 1-x[i+1], 1.1-x[i+1])   }
```

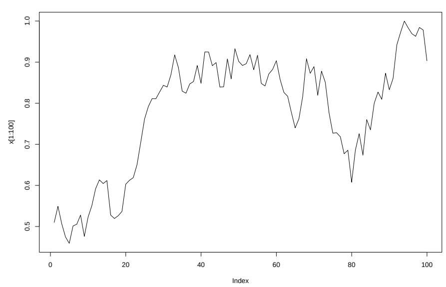
Here is a scatter plot:

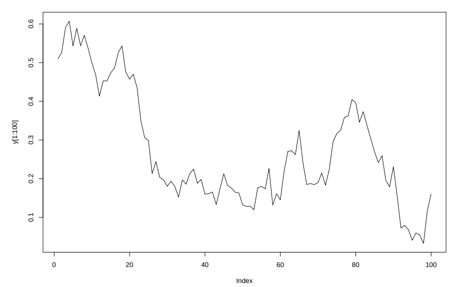
```
plot(x,y)
dev.copy(postscript,'figs/scatterxy2.ps')
dev.off()
```



Let's plot the first 100 values of x and y:

```
plot(x[1:100],type='l')
dev.copy(postscript,'figs/xplot100_2.ps'); dev.off()
plot(y[1:100],type='l')
dev.copy(postscript,'figs/yplot100_2.ps'); dev.off()
```





We see a slower convergence than in the previous case. This is natural because of the shape of the region.

2. Design and implement a Gibbs sampler to simulate from the posterior density $\pi(\mu, \psi|\mathbf{x})$ where $\mathbf{x}$ is a random sample of size n from a Normal distribution with unknown mean mu, and variance $\sigma^2 = \frac{1}{\psi}$, using independent normal and Gamma priors for μ and ψ respectively. (Code for this has been given out to the class.)

   (a) Apply your algorithm to investigate $\pi(\mu, \psi|\mathbf{x})$ where $\mathbf{x}$ is a random sample of size 20 from an N(1, 0.5) density (which you will have to generate yourself! - Remember rnorm() uses the standard deviation.) Use initially the improper, vague priors ($\pi(\mu) \propto 1$, $\pi(\psi) \propto \psi^{-1}$) discussed in lectures. By generating a suitably long sequence of iterates from the chain, estimate the posterior mean and variance of the parameters, and the posterior probabilities that $\mu > 1.5$ and $\sigma^2 > 0.75$.

   (b) Repeat these calculations for a sample size of 60 from an N(1, 0.5) distribution.

   (c) Consider how you might use standard results (see chapter 3 of notes!) in order to check whether your Gibbs sampler is working correctly.

ANSWER

Let $\varphi(x) = (2\pi)^{-1/2}e^{-x^2/2}$ be the standard $N(0,1)$ density. Then $\varphi_{\mu,\sigma^2} = (1/\sigma)\varphi((x-\mu)/\sigma)$ is the $N(\mu,\sigma^2)$ density. With

$$\psi = 1/\sigma^2$$

we find

$$\varphi_{\mu,1/\psi}(x) = (2\pi)^{-1/2}\psi^{1/2}e^{-\frac{1}{2}(x-\mu^2)\psi} := L(\mu,\psi;x)$$

When we have n data points $x_1, \ldots, x_n$, presumably coming as independent samplings from an $N(\mu, 1/\psi)$ density, we evaluate the joint likelihood as

$$\begin{aligned} L(\mu,\psi;\boldsymbol{x}) &= \prod_{i=1}^{n} L(\mu,\psi;x_i), \qquad \boldsymbol{x} = (x_1,\ldots,x_n). \\ &= (2\pi)^{-n/2}\psi^{n/2}\exp\left\{-\frac{\psi}{2}\sum_{i=1}^{n}(x_i-\mu)^2\right\} \\ &\propto \psi^{n/2}\exp\left\{-\frac{\psi}{2}((n-1)s^2(\boldsymbol{x}) + n(\overline{\boldsymbol{x}}-\mu)^2)\right\} \end{aligned}$$

where

$$\overline{\boldsymbol{x}} = \frac{x_1+\cdots+x_n}{n}, \quad s^2(\boldsymbol{x}) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu)^2,$$

are the sample mean and (unbiased version of the) variance of the data vector $\boldsymbol{x}$.

We are being told to use the following prior density:

$$\pi(\mu, \psi) \propto \psi^{-1}, \quad \mu > 0, \psi > 0.$$

The posterior density therefore becomes

$$f(\mu, \psi|\boldsymbol{x}) = \pi(\mu, \psi)L(\mu, \psi; \boldsymbol{x}) \propto \psi^{n/2-1} \exp\left\{-\frac{\psi}{2}[(n-1)s^2(\boldsymbol{x})+n(\overline{\boldsymbol{x}}-\mu)^2]\right\}.$$

We shall write $f(\mu, \psi)$, omitting the $\boldsymbol{x}$.

We compute the marginal distribution of ψ given μ and find that it is GAMMA$[a, b]$ with

$$a = n/2, \quad b = \tfrac{1}{2}[(n-1)s^2(\boldsymbol{x}) + n(\overline{\boldsymbol{x}} - \mu)^2].$$

The marginal distribution of μ given ψ is $N(\overline{\boldsymbol{x}}, 1/n\psi)$.

INPUT: A vector $\boldsymbol{x}$ of size n with sample mean `xbar` and sample variance `s2`.

```
N= 30000
mu=c(1:N)
psi=c(1:N)
mu[1]=1
psi[1]=2
for(i in 1:(N-1))
{
a = n/2
b = (1/2)*((n-1)*s2+n*(xbar-mu[i])^2)
psi[i+1]=rgamma(1,a,b)
mu[i+1]=rnorm(1,xbar,sqrt(1/(n*psi[i+1])))
}
```

OUTPUT: The trajectory $(\mu, \psi) = ((\mu_1, \psi_1), \ldots, (\mu_N, \psi_N))$ of a 2-dimensional Markov chain.

(a) To apply the algorithm, we first generate data:

```
n=20
x=rnorm(n,1,sqrt(0.5))
xbar = mean(x)
s2 = var(x)
```

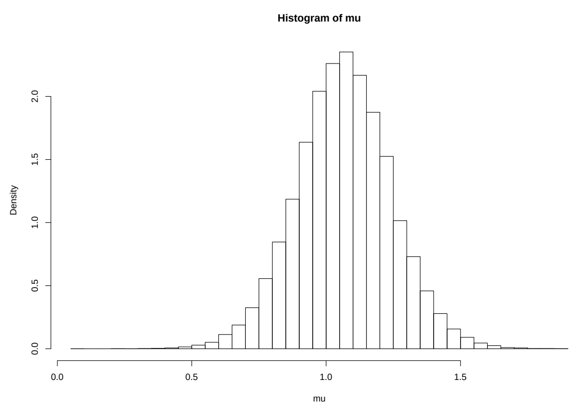
Applying the algorithm, we find:

mean(mu) = 1.066220 mean(psi) = 1.779348 var(mu) = 0.03132511 var(psi) = 0.3333739

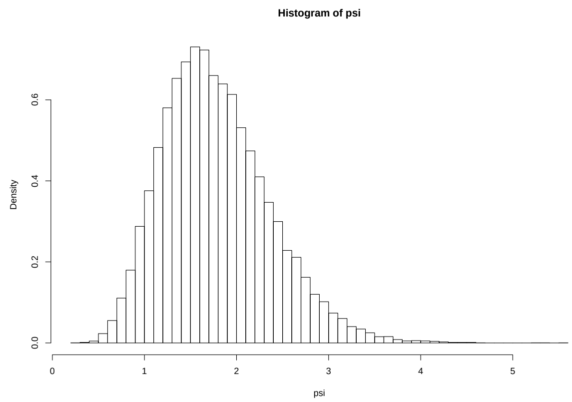
We have almost identifyied the parameters. Remember that the true parameters where $\mu = 1$, $\psi = 1/0.5 = 2$.

But what we have actually obtained is not single parameters, but posterior distributions. Let's sample the historgrams for μ and σ which should serve as proxies for the posterior densities:

```
hist(mu,probability=1,breaks=50)
dev.copy(postscript,'figs/muhist.ps')
dev.off()
```




```
hist(psi,probability=1,breaks=50)
dev.copy(postscript,'figs/psihist.ps')
dev.off()
```



Estimation of the posterior probability that $\mu > 1.5$:

```
sum(mu>1.5)/N
```

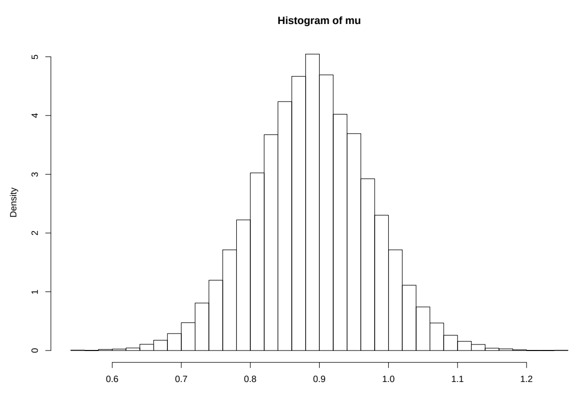
The answer is: 0.0091

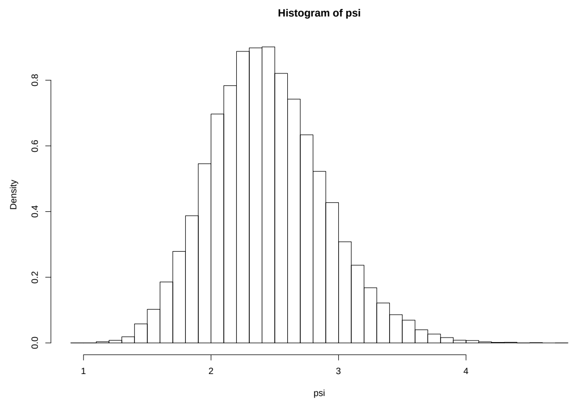
Estimation of the posterior probability that $\sigma^2 > 0.75$, i.e. that $\psi < 4/3$:

```
sum(psi<4/3)/N
```

The answer is: 0.2323667

(b) We now generate $n = 60$ i.i.d. samples from $N(1, 0.5)$ and run the algorithm on them. We obtain:





mean(mu) = 0.8891722 mean(psi) = 2.442945 var(mu) = 0.007085651 var(psi) = 0.2022883

3. Let $\mathbf{x} = (5.25, 4.80, 4.55, 5.8, 5.3, 4.38, 3.08, 5.60)$ denote a random sample from a $\Gamma(\alpha, \beta)$ distribution for which $\sum x_i = 38.76$ and $\prod x_i = 266, 274$. Assume that *a priori* $\alpha \sim U(1, 15)$ and $\beta \sim Exp(0.1)$. Implement in R the Metropolis algorithm described in lectures for simulating from the posterior $\pi(\alpha, \beta|\mathbf{x})$. By applying the method of moments to the data, identify suitable initial values for α and β. Use trial and error to identify suitable step-sizes for updates to α and β.

   (a) Estimate the posterior mean and variance of α and β from a suitably long run of the chain. Examine the shape of the marginal histogram for α and estimate its posterior mode.

   (b) Estimate 90% equal-tailed credible intervals for α and β from the output of the chain. (You can do this using the sort() command in R which arranges the elements in a vector in increasing order. The end points of a credible interval can then be obtained from the ordered Markov chain output.)

   (c) How do the marginal distributions of α and β change when the prior for β is selected to be a) $Exp(1)$ and b) $Exp(5)$?

   (d) By plotting the points (α_i, β_i) on a scatter diagram, investigate the dependence of α and β in the posterior distribution.

   (e) The data were generated from a $Gamma(8, 1.5)$ distribution. Simulate random samples of size 20 and size 40 from this distribution using the rgamma(n, alpha, beta) function in R and apply your algorithm to these samples in order to estimate (α, β) using the $Exp(0.1)$ prior for β. Investigate how the posterior marginal densities for the parameters change as the sample size becomes larger.

4. Modify your code for the M-H sampler of the previous question (inference on (α, β) in the $\Gamma(\alpha, \beta)$) distribution) by using a Gibbs update for β instead of the Metropolis step. By examining trace plots of the values of α and β against iterate and/or calculating autocorrelation functions determine whether the Gibbs sampler has superior mixing properties.