Assignment 4

Mary Silva

3/13/2019

1A

Using $\theta_1 = 1.5$ and $\theta_2 = 2$ we draw a sample of size 1000 using the independence Metropolis Hastings algorithm with gamma distribution as the proposal density.

```
theta_1 = 1.5 # true value theta1
theta_2 = 2 # true value theta2
mean_z1 = sqrt(theta_2/theta_1)
mean_z2 = sqrt(theta_1/theta_2) + 1/(2*theta_2)

# hyperparams
b = 2.5
a = mean_z1*b
#M-H Algorithm
MH_alg1 = function(N){
  MH_samples = rep(NA, N)
  count = 0
  current_z = 1.0
  for(i in 1:N){
    curr_p = pdf_z(current_z)
    z_new = rgamma(1, a, b)
    p_new = pdf_z(z_new)

    accept = exp(p_new + dgamma(current_z,a,b,log = T) -
                   p_new - dgamma(z_new,a,b,log = T))
    if(runif(1) < accept){
      current_z = z_new
      count = count + 1
    }
    MH_samples[i] = current_z
  }
  return(list(MH_samples=MH_samples,count=count))
}
```

After trying several hyperparameters for different Gamma distributions, the best sample obtains a mean, $E(Z)$, of

```
## [1] 1.763996
```

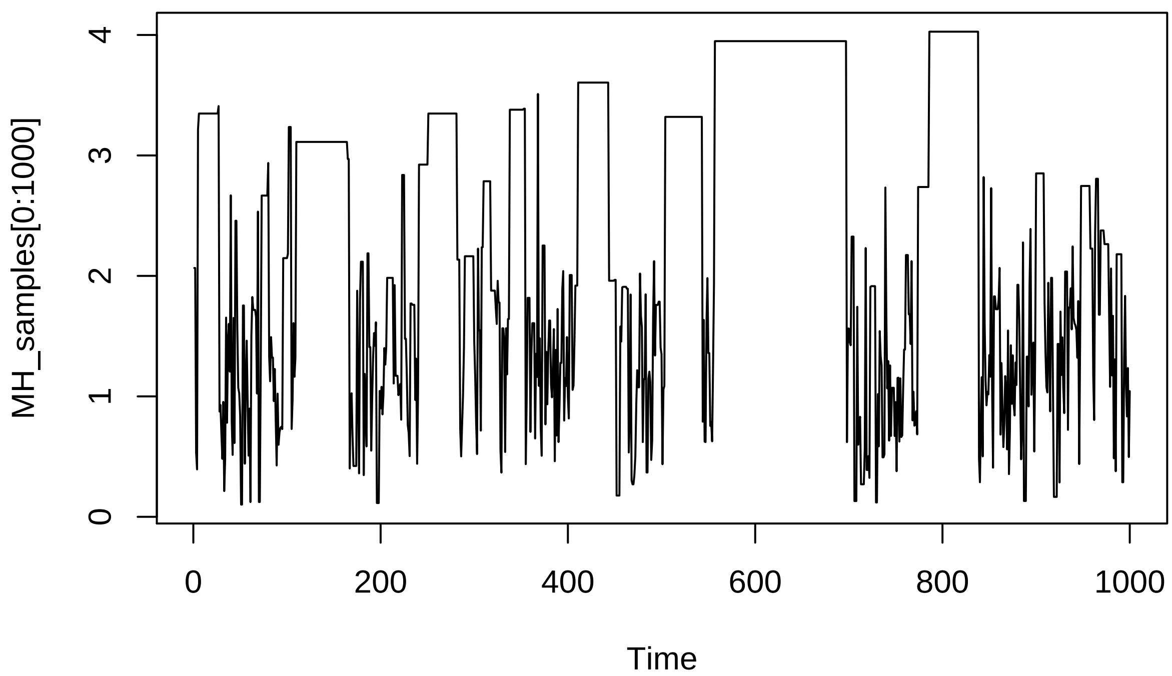
$E(1/Z)$

```
## [1] 1.537128
```

and an accuracy of

```
## [1] 0.513
```

The traceplot for the samples for Metropolis-Hastings is shown below:



1B

The density of $W = log(Z)$ is given by

$$f_W(w) \propto \exp\left\{-\frac{3}{2}w - \theta_1 exp\{w\} - \frac{\theta_2}{\exp(w)}\right\}\exp(w)$$

We draw a sample of size 1000 using the random-walk Metropolis algorithm with this density.

```
v = 0.01
MH_RW = function(N){
  N = N
  MH_RW = rep(NA, N)
  a_count = 0
  z_curr = 1.0
  for (i in 1:N) {
    p_curr = pdf_z2(z_curr)
    z_new = exp(log(z_curr) + rnorm(1,0,sqrt(v)))
    p_new = pdf_z2(z_new)
    acceptance = exp(p_new - p_curr)
    if(runif(1) < acceptance){
      z_curr = z_new
      a_count = a_count+1
    }
    MH_RW[i] = z_curr
  }
  return(list(MH_RW=MH_RW, a_count=a_count))
}
```

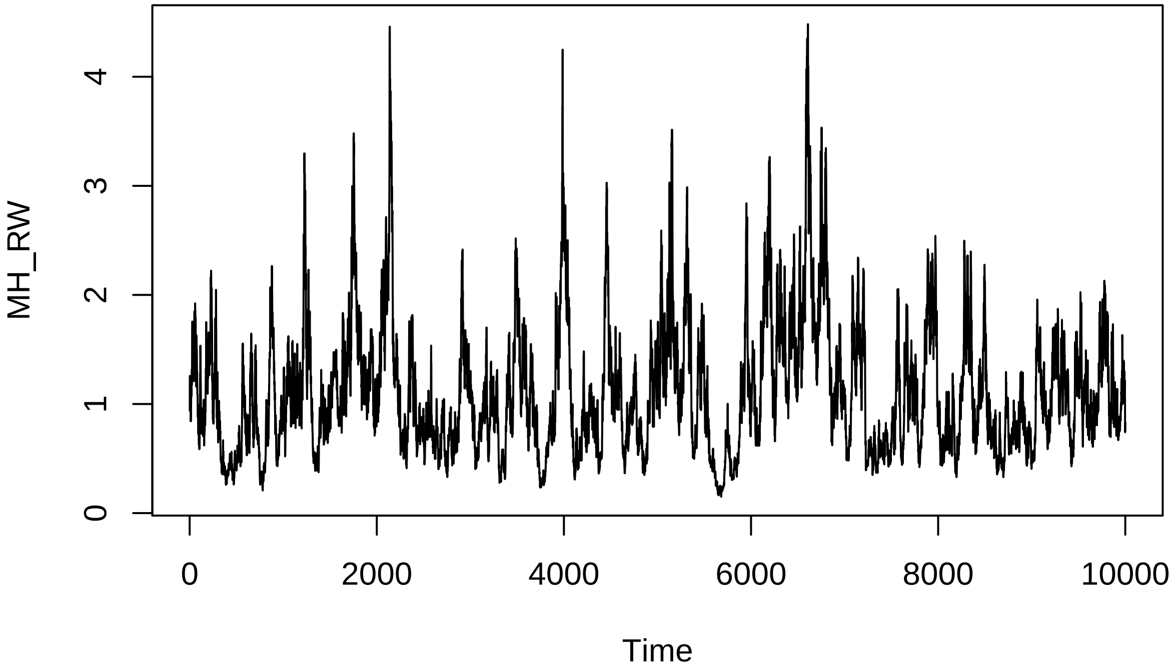
The mean for the samples, $E(W_{samples})$, is

```
## [1] 1.114488
```

And the accuracy is

```
## [1] 0.9363
```

If we use 10000 metropolis hastings random ralk samples, the traceplot is shown below



2A

$$x_i|\nu, \theta \sim Gamma(\nu, \theta)$$
$$\nu \sim Gamma(a, b)$$
$$\theta \sim Gamma(\alpha, \beta)$$

The joint posterior for θ and ν

$$\pi(\theta, \nu, \boldsymbol{x}) \propto \frac{\left(\prod_{i=1}^{n} x_i\right)^{\nu-1} \nu^{a-1} e^{-b\nu}}{\left(\Gamma(\nu)\right)^n} \theta^{a+n\nu-1} \exp\left\{-\theta\left(\beta + \sum_{i=1}^{n} x_i\right)\right\}$$

The full conditionals:

$$\pi(\theta|\nu, \boldsymbol{x}) \propto \theta^{a+n\nu-1} \exp\left\{-\theta\left(\beta + \sum_{i=1}^{n} x_i\right)\right\}$$

thus, $\theta|\nu, \boldsymbol{x} \sim Gamma(n\nu, \beta + \sum x_i)$.

$$\pi(\nu|\theta, \boldsymbol{x}) \propto \theta^{n\nu} \frac{\left(\prod_{i=1}^{n} x_i\right)^{\nu-1} \nu^{a-1} e^{-b\nu}}{\left(\Gamma(\nu)\right)^n}$$

which is not a recognizable distribution. We use a Metropolis within Gibbs algorithm to sample from the full conditionals, using a random walk proposal on $log(\nu)$. I tried various hyperparameters appropriate for this data.

```r
sample = NULL
N = 1000
sample$theta = rep(NA,N)
sample$nu = rep(NA,N)
alpha = 3
beta = 2
v = 0.05
theta_curr = 2
nu_curr = 3
set.seed(2)
for(i in 1:N){
  theta_curr = rgamma(1, n*nu_curr + alpha, beta + sum_x)
  nu_new = exp(log(nu_curr) + rnorm(1,0,sqrt(v)))
  pnu_curr = nu_condit(nu_curr, theta_curr)
  pnu_new = nu_condit(nu_new, theta_curr)
  accept = exp(pnu_new - pnu_curr)
  if(runif(1) < accept)
    nu_curr = nu_new

  sample$theta[i] = theta_curr
  sample$nu[i] = nu_new
}
```

The effective sample size for θ is

```
##     var1
## 37.15345
```

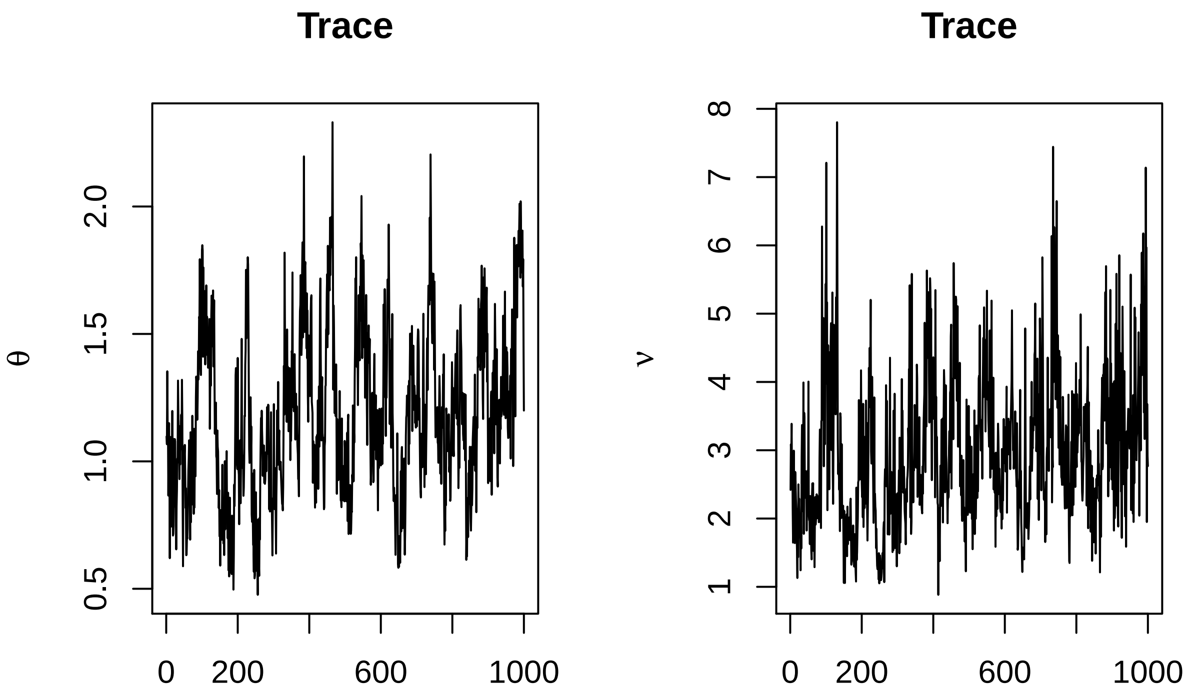
The effective sample size for ν is

```
##     var1
## 62.45584
```

The table below summarizes the results

parameter	mean	95% Credible Interval
θ	1.13	(0.635,1.846)
ν	2.807	(1.669, 4.55)

The traceplots are below



2B

Now we develop a Metropolis-Hastings algorithm that jointly proposes $\log(\nu)$ and $\log(\theta)$ using a Gaussian random walk centered on the current value of the parameters. Tune the variance-covariance matrix of the proposal using a test run that proposes the parameters independently:

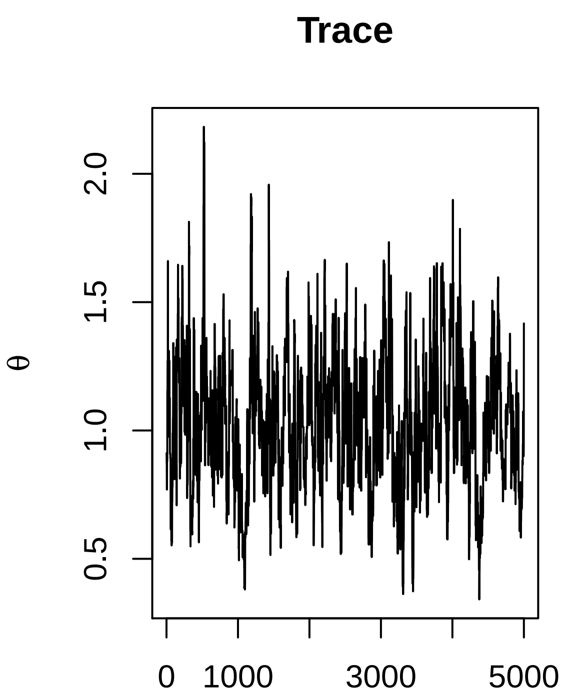
```r
V = 0.05*diag(2)
theta_curr = 2
nu_curr = 3
N_test = 5000
for(i in 1:N_test){
  nu_new = exp(log(nu_curr) + rnorm(1,0,sqrt(V[1,1])))
  theta_new = exp(log(theta_curr) + rnorm(1,0, V[2,2]))
  p_curr = pcurr(nu_curr, theta_curr)
  p_new = pcurr(nu_new, theta_new)
  accept = exp(p_new - p_curr)
  if(runif(1) < accept){
    nu_curr = nu_new
    theta_curr = theta_new
  }
  sample$theta[i] = theta_curr
  sample$nu[i] = nu_curr
}
```

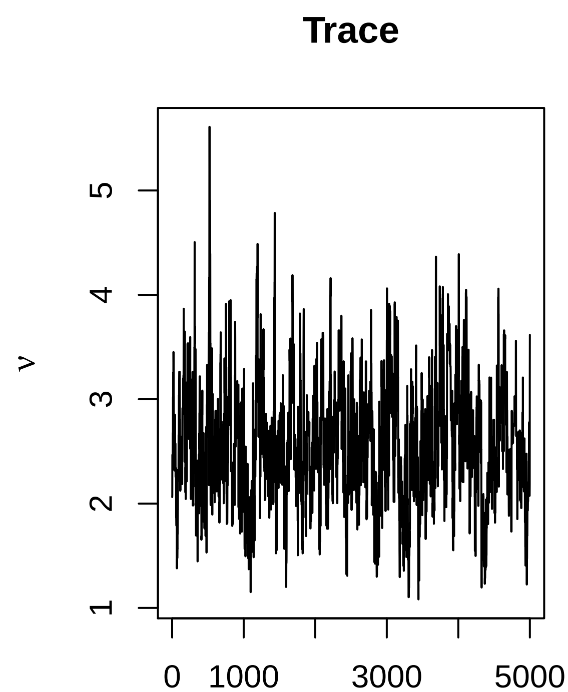
```r
for(i in N_test+1:10000){
  new = mvrnorm(1, c(log(nu_curr), log(theta_curr)), V)
  nu_new = exp(new[1])
  theta_new = exp(new[2])
  p_curr = pcurr2(nu_curr, theta_curr)
  p_new = pcurr2(nu_new, theta_new)
  acceptance = exp(p_new - p_curr)
```

```
  if(runif(1) < acceptance){
    nu_curr = nu_new
    theta_curr = theta_new
  }
  sample$theta[i] = theta_curr
  sample$nu[i] = nu_curr
}
```

parameter	mean	95% Credible Interval
θ	1.11	(0.59,1.79)
ν	2.807	(1.49, 4.38)

The trace plot for these samples are below





2C

Now we are going to develop a Metropolis algorithm that jointly proposes $\log \nu$ and $\log \theta$ using independent proposals based on Laplace approximation of the posterior distribution of $\log \nu$ and $\log \theta$.

We let $t = \log \theta$ and $v = \log \nu$, then the posterior becomes

$$
\begin{aligned}
\pi(\theta, \nu | \boldsymbol{x}) &\propto \exp\left\{(\nu - 1)\sum_{i=1}^{n} \log x_i + (a-1)\log \nu - b\nu - n \log \Gamma(\nu)\right\} \\
&\times \exp\left\{(\alpha + n\nu - 1)\log \theta - \theta\left(\beta + \sum_{i=1}^{n} x_i\right)\right\} \\
\Rightarrow \pi(t, v|\boldsymbol{x}) &\propto \exp\left\{(e^v - 1) sum_{i=1}^n \log x_i + av - be^v - n\log\Gamma(e^v)\right\} \\
&\times \exp\left\{(a + ne^v)t - e^t\left(\beta + sum_{i=1}^n x_i\right)\right\}
\end{aligned}
$$

Now, we let

$$h(t,v) = (e^v = 1)\sum_{i=1}^{n} x_i + av - be^v - n\log\Gamma(e^v)\exp\left\{(a+ne^v)t - e^t\left(\beta + sum_{i=1}^{n} x_i\right)\right\}$$

Then we use the definition of Laplace approximation

```
h = function(w) {
  a1 = (exp(w[2]) - 1) * sum_logx + 3 * w[2] - exp(w[2])
  return(-(a1 - n * lgamma(exp(w[2])) +
             (2 + n * exp(w[2])) * w[1] - exp(w[1]) * (2 + sum_x)))
}
laplace = optim(c(0,1), h, hessian = T)
```

The laplace maximum for the parameters are

```
## [1] 0.09685737 1.00165219
```

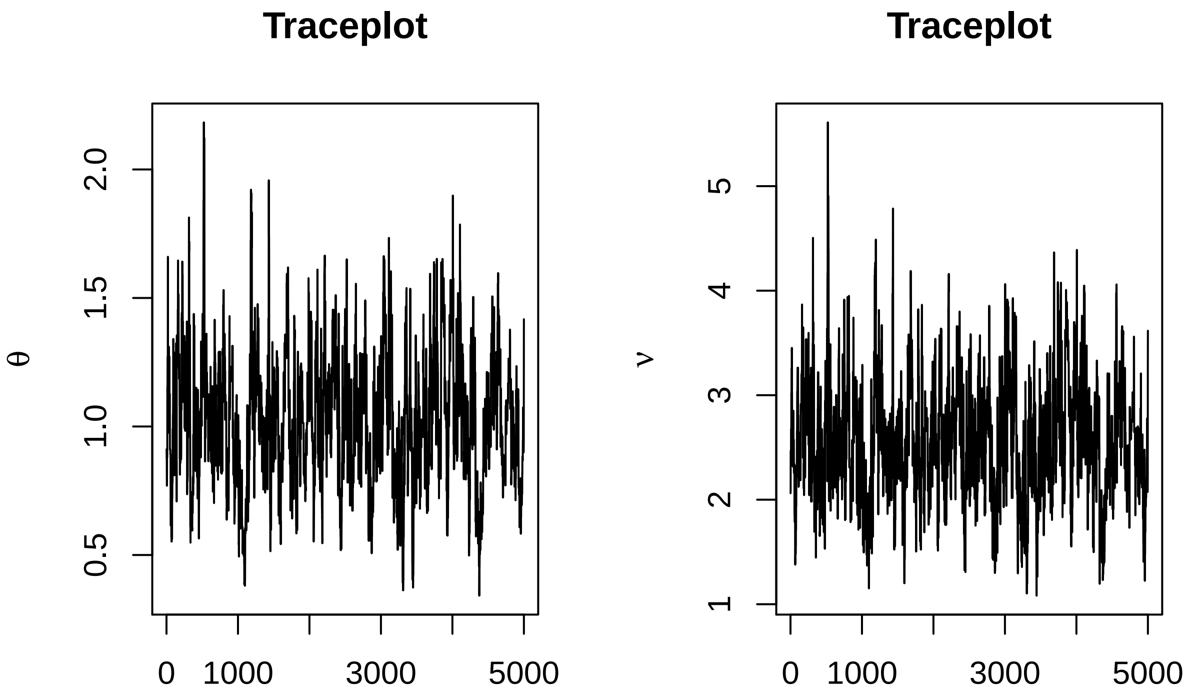
and the hessian obtained at the maximum is

```
##           [,1]      [,2]
## [1,]  70.06943 -68.06943
## [2,] -68.06943  85.06425
```

Now we update the variance-covariance matrix then resume the Metropolis sampling algorithm

```
for(i in N_test+1:N){
  nu_new = exp(log(nu_curr) + rnorm(1,0,sqrt(V[1,1])))
  theta_new = exp(log(theta_curr) + rnorm(1,0,sqrt(V[2,2])))
  p_curr = pcurr(nu_curr = nu_curr, theta_curr = theta_curr)
  p_new =pcurr(nu_curr = nu_new, theta_curr = theta_new)

  accept = exp(p_new - p_curr)
  if(runif(1) < accept){
    nu_curr = nu_new
    theta_curr = theta_new
  }
  sample$theta[i] = theta_curr
  sample$nu[i] = nu_curr
}
```



The effective sample size associated with θ is

```
##     var1
## 152.8612
```

The effective sample size associated with ν is

```
##     var1
## 182.2601
```

parameter	mean	95% Credible Interval
θ	1.05	(0.56,1.58)
ν	2.55	(1.49, 3.79)