STAT 450: Bayesian Statistics - Homework 8

Problem 1: LeBron James' 3-Point Shooting Percentage

```
LBJ3points <- readRDS("LBJ3points.rds")   # load dataset
```

(a)

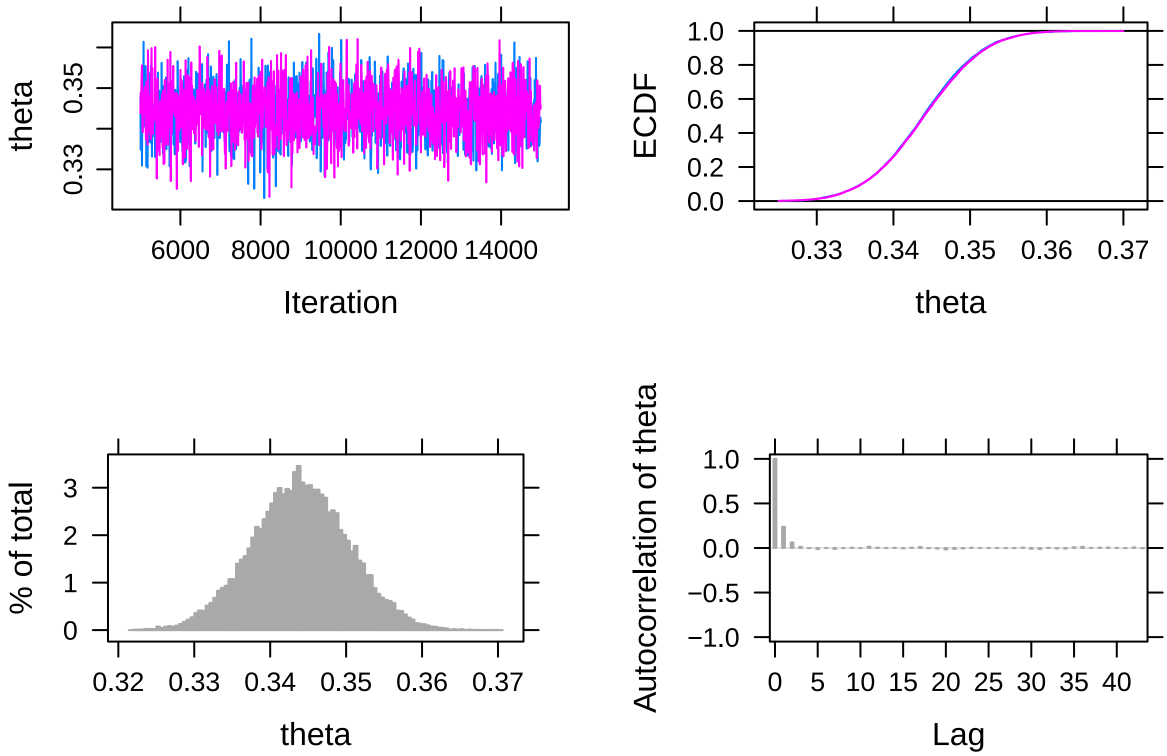
```
mtext <- "
model
  {
  for(j in 1:17){
    y[j] ~ dbinom(theta, n[j])
  }
  theta ~ dbeta(a, b)
  y_pred ~ dbinom(theta, 320)
  }
"
dat <- list(y = LBJ3points$Made, n = LBJ3points$Attempted, a = 1, b = 1)

start_val <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
              )

out <- run.jags(model = mtext,
                monitor = c("theta","y_pred"),
                data = dat,
                n.chains = 2,
                inits = start_val,
                sample = 10000)
```

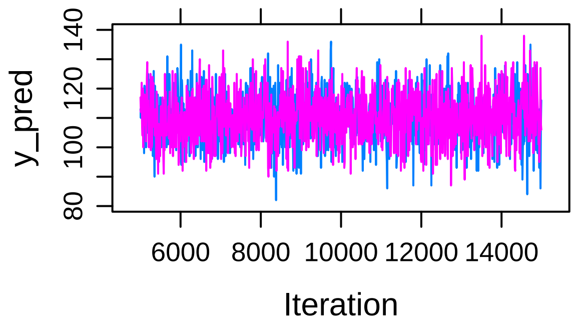
```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:17:32 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 17
##    Unobserved stochastic nodes: 2
##    Total graph size: 39
## . Reading parameter file inits1.txt
```

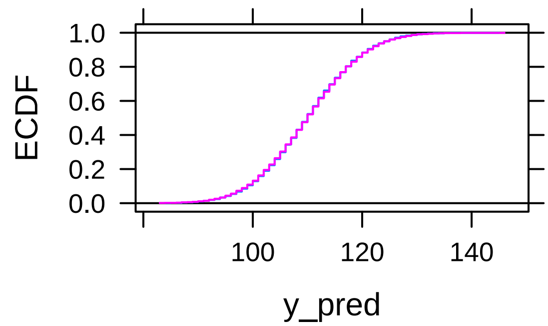
```
## . Reading parameter file inits2.txt
## . Initializing model
## . Adapting 1000
## -------------------------------------------------| 1000
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . Updating 0
## . Deleting model
## .
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Calculating summary statistics...
## Calculating the Gelman-Rubin statistic for 2 variables....
## Finished running the simulation
```

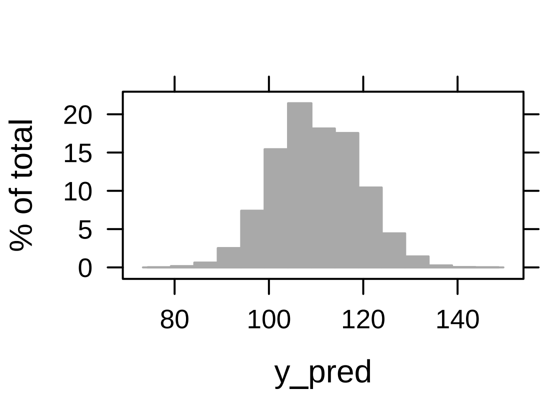
```r
plot(out)
```

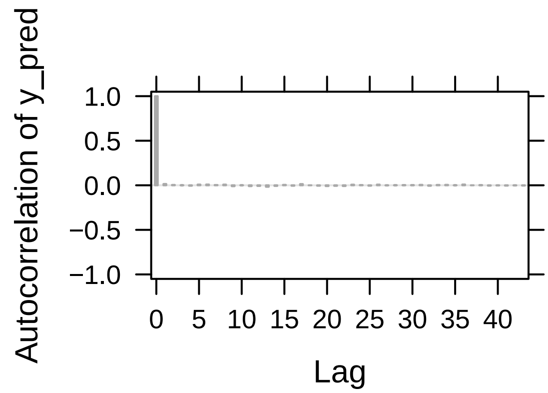
```
## Generating plots...
```

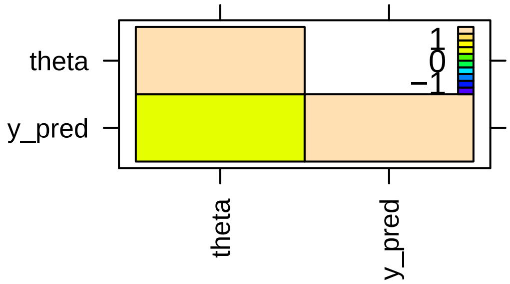












95% credible interval for theta and posterior predictions

```
parameters <- rbind(out$mcmc[[1]], out$mcmc[[2]])
kable(apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975))))
```

	theta	y_pred
2.5%	0.3314949	93
97.5%	0.3565012	127

(b)

```
mtext <- "
model{
  for(j in 1:17){
    y[j] ~ dbinom(theta[j], n[j])
    theta[j] ~ dbeta(a,b)
  }
}
"
```

```r
dat <- list(y = LBJ3points$Made,
            n = LBJ3points$Attempted,
            a = 1, b = 1)

start_val <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
              )


out <- run.jags(model = mtext,
                monitor = "theta",
                data = dat,
                n.chains = 2,
                inits = start_val,
                sample = 10000)
```

```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:17:35 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 17
##    Unobserved stochastic nodes: 17
##    Total graph size: 53
## . Reading parameter file inits1.txt
## . Reading parameter file inits2.txt
## . Initializing model
## . Adapting 1000
## -------------------------------------------------| 1000
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . Updating 0
## . Deleting model
## .
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Calculating summary statistics...
## Calculating the Gelman-Rubin statistic for 17 variables....
## Finished running the simulation
```

```r
parameters <- rbind(out$mcmc[[1]], out$mcmc[[2]])
```

```r
m <- matrix(1:17, 5)
```

```
## Warning in matrix(1:17, 5): data length [17] is not a sub-multiple or multiple
## of the number of rows [5]
```

```r
for (i in 1:ncol(m)) {
 cat(kable(
   apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975)))[, m[, i]],
   'latex', booktabs=TRUE) %>% kable_styling(full_width = T), "\\newline")
}
```

	theta[1]	theta[2]	theta[3]	theta[4]	theta[5]
2.5%	0.2356278	0.2983262	0.2903312	0.2708958	0.2693702
97.5%	0.3544312	0.4048961	0.3845028	0.3735881	0.3635286

	theta[6]	theta[7]	theta[8]	theta[9]	theta[10]
2.5%	0.2979268	0.2886659	0.2765157	0.2892278	0.3466759
97.5%	0.3936642	0.3812821	0.3860883	0.4410923	0.4666602

	theta[11]	theta[12]	theta[13]	theta[14]	theta[15]
2.5%	0.3268736	0.3060445	0.2564087	0.3134492	0.3219759
97.5%	0.4342044	0.4064084	0.3648174	0.4144924	0.4142270

	theta[16]	theta[17]	theta[1]	theta[2]	theta[3]
2.5%	0.2904719	0.3027610	0.2356278	0.2983262	0.2903312
97.5%	0.3922880	0.3977981	0.3544312	0.4048961	0.3845028

(c)

```r
mtext <- "
model{
  for(j in 1:17){
    y[j] ~ dbinom(theta[j], n[j])
    theta[j] ~ dbeta(a,b)
  }

  mu ~ dbeta(10, 20)
  nu ~ dlnorm(0, 1/4)

  a <- mu*nu
  b <- nu*(1 - mu)

  theta_new ~ dbeta(a,b)
  ytilde ~ dbinom(theta_new, 320)
}
"

dat <- list(y = LBJ3points$Made,
            n = LBJ3points$Attempted)

start_val <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
```

```r
             )


out <- run.jags(model = mtext,
                monitor = c("theta","a","b","mu","nu","theta_new","ytilde"),
                data = dat,
                n.chains = 2,
                inits = start_val,
                sample = 10000)
```

```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:17:44 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 17
##    Unobserved stochastic nodes: 21
##    Total graph size: 65
## . Reading parameter file inits1.txt
## . Reading parameter file inits2.txt
## . Initializing model
## . Adapting 1000
## -------------------------------------------------| 1000
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . . . . . . . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . Updating 0
## . Deleting model
## .
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Calculating summary statistics...
## Calculating the Gelman-Rubin statistic for 23 variables....
## Finished running the simulation
```

95% credible interval for the thetas, other parameters, and predictions

```r
parameters <- rbind(out$mcmc[[1]], out$mcmc[[2]])

m <- matrix(1:23, 5)
```

```
## Warning in matrix(1:23, 5): data length [23] is not a sub-multiple or multiple
## of the number of rows [5]
```

```r
for (i in 1:ncol(m)) {
 cat(kable(
   apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975)))[, m[, i]],
   'latex', booktabs=TRUE) %>% kable_styling(full_width = T), "\\newline")
}
```

	theta[1]	theta[2]	theta[3]	theta[4]	theta[5]
2.5%	0.2825873	0.3118976	0.3067575	0.2983979	0.2953011
97.5%	0.3627394	0.3824072	0.3732602	0.3678791	0.3638971

	theta[6]	theta[7]	theta[8]	theta[9]	theta[10]
2.5%	0.3106608	0.3055645	0.3024278	0.3106269	0.3293528
97.5%	0.3772274	0.3722451	0.3738451	0.3911898	0.4127965

	theta[11]	theta[12]	theta[13]	theta[14]	theta[15]
2.5%	0.3234886	0.3155780	0.2906395	0.3193348	0.3224000
97.5%	0.3990245	0.3846601	0.3656440	0.3885046	0.3891459

	theta[16]	theta[17]	a	b	mu
2.5%	0.3080289	0.3135494	41.36392	79.02756	0.326824
97.5%	0.3760820	0.3805420	882.73547	1694.70875	0.361615

	nu	theta_new	ytilde	theta[1]	theta[2]
2.5%	120.5159	0.2931940	87	0.2825873	0.3118976
97.5%	2579.6118	0.3974731	134	0.3627394	0.3824072

(d)

The CI in part c is larger. This would seem surprising at first, since we would usually expect a hierarchical model to have less certainty than a combined model. But this result in our context actually makes more sense because we are using informative priors which introduces more variability to the posterior predictions since they don't utilize as much information from the data as the cases where we use flat priors.

Problem 2: Radon Levels in Minnesota Counties

```r
radonData <- readRDS("radon.rds")   # load dataset
```

(a)

```r
mtext <- "
model
  {
  for(j in 1:n){
    y[j] ~ dexp(theta)
  }
  theta ~ dgamma(1, 1)
  }
"
dat <- list(y = radonData$radon,
            n = length(radonData$radon))
```

```r
start_val <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
              )

out <- run.jags(model = mtext,
                monitor = "theta",
                data = dat,
                n.chains = 2,
                inits = start_val,
                sample = 10000)
```

```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:17:56 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 919
##    Unobserved stochastic nodes: 1
##    Total graph size: 922
## . Reading parameter file inits1.txt
## . Reading parameter file inits2.txt
## . Initializing model
## . Adaptation skipped: model is not in adaptive mode.
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . . Updating 0
## . Deleting model
## .
## Note: the model did not require adaptation
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Calculating summary statistics...
## Calculating the Gelman-Rubin statistic for 1 variables....
## Finished running the simulation
```

95% credible interval and mean for theta

```r
parameters <- rbind(out$mcmc[[1]], out$mcmc[[2]])
kable(apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975))))  %>% kable_styling()
```

	theta
2.5%	0.1965109
97.5%	0.2234942

```r
apply(parameters, 2, mean)
```

```
##     theta
## 0.2098209
```

(b)

```r
mtext_im <- "
model{
  for(j in 1:n){
      y[j] ~ dexp(theta[county[j]])
  }

  for(i in 1:k){
    theta[i] ~ dgamma(1,1)
  }
}
"

dat_im <- list(y = radonData$radon,
               n = length(radonData$radon),
               county = as.numeric(radonData$county),
               k = n_distinct(radonData$county)
               )


start_val_im <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
              )


out_im <- run.jags(model = mtext_im,
                   monitor = "theta",
                   data = dat_im,
                   n.chains = 2,
                   inits = start_val_im,
                   sample = 10000)
```

```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:17:57 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 919
##    Unobserved stochastic nodes: 85
##    Total graph size: 1926
## . Reading parameter file inits1.txt
## . Reading parameter file inits2.txt
```

```
## . Initializing model
## . Adaptation skipped: model is not in adaptive mode.
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . Updating 0
## . Deleting model
## .
## Note: the model did not require adaptation
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Note: Summary statistics were not produced as there are >50 monitored
## variables
## [To override this behaviour see ?add.summary and ?runjags.options]
## FALSEFinished running the simulation
```

```
parameters <- rbind(out_im$mcmc[[1]], out_im$mcmc[[2]])
```

95% credible intervals for the thetas

```
m <- matrix(1:n_distinct(radonData$county), 5)

for (i in 1:ncol(m)) {
 cat(kable(
   apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975)))[, m[, i]],
   'latex', booktabs=TRUE) %>% kable_styling(full_width = T), "\\newline")
}
```

	theta[1]	theta[2]	theta[3]	theta[4]	theta[5]
2.5%	0.1733221	0.2554955	0.0971188	0.1197472	0.1003178
97.5%	1.1057900	0.4407871	0.7889123	0.4937419	0.6396127

	theta[6]	theta[7]	theta[8]	theta[9]	theta[10]
2.5%	0.0689965	0.0777002	0.0686465	0.1825852	0.0622856
97.5%	0.5496287	0.2195083	0.4234030	0.6018494	0.2931443

	theta[11]	theta[12]	theta[13]	theta[14]	theta[15]
2.5%	0.0952124	0.0545961	0.1208249	0.0669462	0.1207380
97.5%	0.5005984	0.3448886	0.5682443	0.1887668	0.7608299

	theta[16]	theta[17]	theta[18]	theta[19]	theta[20]
2.5%	0.1201818	0.0766869	0.1802253	0.1652346	0.0569786
97.5%	1.4109390	0.4781313	0.5445839	0.2703782	0.4572456

	theta[21]	theta[22]	theta[23]	theta[24]	theta[25]
2.5%	0.0950008	0.1617704	0.0912701	0.0555315	0.0680976
97.5%	0.3378735	0.7428886	1.0806500	0.1966296	0.1907314

	theta[26]	theta[27]	theta[28]	theta[29]	theta[30]
2.5%	0.1803509	0.0854438	0.1553692	0.1115202	0.1815141
97.5%	0.2641956	0.4033040	0.8129081	0.8972059	0.5770422

	theta[31]	theta[32]	theta[33]	theta[34]	theta[35]
2.5%	0.0491150	0.0959933	0.0460024	0.0810608	0.2835561
97.5%	0.2556956	0.6011901	0.2896451	0.6650488	1.1847537

	theta[36]	theta[37]	theta[38]	theta[39]	theta[40]
2.5%	0.0216815	0.2586816	0.0745858	0.0745115	0.0392988
97.5%	0.2513941	0.9228481	0.4714532	0.3973232	0.2528610

	theta[41]	theta[42]	theta[43]	theta[44]	theta[45]
2.5%	0.0732578	0.0502028	0.0531769	0.1107445	0.1006934
97.5%	0.2839461	1.1311135	0.1897570	0.4666069	0.2918213

	theta[46]	theta[47]	theta[48]	theta[49]	theta[50]
2.5%	0.1096874	0.0868541	0.1638715	0.0820678	0.0188560
97.5%	0.6004238	0.9914347	0.5900860	0.2388514	0.4247716

	theta[51]	theta[52]	theta[53]	theta[54]	theta[55]
2.5%	0.0406576	0.0488443	0.0898719	0.1524920	0.0962704
97.5%	0.2536894	0.3896447	0.7203508	0.3438637	0.3712749

	theta[56]	theta[57]	theta[58]	theta[59]	theta[60]
2.5%	0.1003544	0.2047254	0.0571433	0.0744452	0.0753878
97.5%	0.8009600	0.9480596	0.3600087	0.4752891	0.8759128

	theta[61]	theta[62]	theta[63]	theta[64]	theta[65]
2.5%	0.1933709	0.0506663	0.0759628	0.0794837	0.0566146
97.5%	0.3846066	0.2685269	0.5865605	0.2547533	0.6560429

	theta[66]	theta[67]	theta[68]	theta[69]	theta[70]
2.5%	0.1379992	0.0782875	0.1553110	0.0985795	0.1362739
97.5%	0.3808476	0.2251370	0.5949958	0.6346976	0.2961233

	theta[71]	theta[72]	theta[73]	theta[74]	theta[75]
2.5%	0.0964766	0.0462161	0.2710908	0.1273150	0.0669337
97.5%	0.3252513	0.5348295	0.3881614	0.7953466	0.5578494

	theta[76]	theta[77]	theta[78]	theta[79]	theta[80]
2.5%	0.0481293	0.0697996	0.1292648	0.1195768	0.1580101
97.5%	0.3020557	0.2925101	0.6746982	0.7728183	0.2802939

	theta[81]	theta[82]	theta[83]	theta[84]	theta[85]
2.5%	0.0352659	0.0233255	0.0951279	0.1007816	0.0814168
97.5%	0.2844516	0.5434632	0.2778953	0.2892907	0.9553273

Means of the thetas

```
apply(parameters, 2, mean)
```

```
##  theta[1]  theta[2]  theta[3]  theta[4]  theta[5]  theta[6]  theta[7]  theta[8]
## 0.5362657 0.3422697 0.3639023 0.2747645 0.3126868 0.2529891 0.1391145 0.2063438
##  theta[9] theta[10] theta[11] theta[12] theta[13] theta[14] theta[15] theta[16]
## 0.3612504 0.1567261 0.2577238 0.1697705 0.3023875 0.1196634 0.3702195 0.5879738
## theta[17] theta[18] theta[19] theta[20] theta[21] theta[22] theta[23] theta[24]
## 0.2333111 0.3392552 0.2148719 0.2072272 0.1977381 0.3997910 0.4502400 0.1152557
```

```
## theta[25] theta[26] theta[27] theta[28] theta[29] theta[30] theta[31] theta[32]
## 0.1214570 0.2201477 0.2153148 0.4184152 0.4075866 0.3521067 0.1319541 0.2943202
## theta[33] theta[34] theta[35] theta[36] theta[37] theta[38] theta[39] theta[40]
## 0.1415720 0.3016505 0.6565996 0.1059012 0.5390579 0.2309618 0.2039448 0.1234785
## theta[41] theta[42] theta[43] theta[44] theta[45] theta[46] theta[47] theta[48]
## 0.1609536 0.4080397 0.1109391 0.2573783 0.1839294 0.3041124 0.4147870 0.3462583
## theta[49] theta[50] theta[51] theta[52] theta[53] theta[54] theta[55] theta[56]
## 0.1505990 0.1522014 0.1250856 0.1800789 0.3255366 0.2390406 0.2104200 0.3678347
## theta[57] theta[58] theta[59] theta[60] theta[61] theta[62] theta[63] theta[64]
## 0.5107219 0.1756520 0.2327807 0.3644328 0.2808763 0.1377363 0.2683229 0.1550024
## theta[65] theta[66] theta[67] theta[68] theta[69] theta[70] theta[71] theta[72]
## 0.2732147 0.2439810 0.1416399 0.3406859 0.3080089 0.2094706 0.1929697 0.2222867
## theta[73] theta[74] theta[75] theta[76] theta[77] theta[78] theta[79] theta[80]
## 0.3268243 0.3906356 0.2527719 0.1471876 0.1614670 0.3479942 0.3744397 0.2145548
## theta[81] theta[82] theta[83] theta[84] theta[85]
## 0.1301535 0.1933472 0.1739793 0.1833019 0.3942317
```

(c)

```
mtext_hm <- "
model{
  for(j in 1:n){
      y[j] ~ dexp(theta[county[j]])
  }

  for(i in 1:k){
    theta[i] ~ dgamma(a,b)
  }

  mu ~ dgamma(0.01, 0.05)
  sigma ~ dunif(0, 1)

  a <- mu^3 / sigma^2
  b <- mu / sigma^2

}
"

dat_hm <- list(y = radonData$radon,
               n = length(radonData$radon),
               county = as.numeric(radonData$county),
               k = n_distinct(radonData$county)
               )


start_val_hm <- list(
              list( ".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45081),
              list(".RNG.name" = "base::Wichmann-Hill", ".RNG.seed" = 45082)
              )


out_hm <- run.jags(model = mtext_hm,
                   monitor = c("theta", "a", "b"),
                   data = dat_hm,
```

```r
                   n.chains = 2,
                   inits = start_val_hm,
                   sample = 10000)
```

```
## Calling the simulation...
## Welcome to JAGS 4.3.2 on Mon May 20 07:18:02 2024
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 919
##    Unobserved stochastic nodes: 87
##    Total graph size: 1937
## . Reading parameter file inits1.txt
## . Reading parameter file inits2.txt
## . Initializing model
## . Adapting 1000
## -------------------------------------------------| 1000
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
## . Updating 4000
## -------------------------------------------------| 4000
## ************************************************** 100%
## . . . . Updating 10000
## -------------------------------------------------| 10000
## ************************************************** 100%
## . . . . . . Updating 0
## . Deleting model
## .
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Note: Summary statistics were not produced as there are >50 monitored
## variables
## [To override this behaviour see ?add.summary and ?runjags.options]
## FALSEFinished running the simulation
```

```r
parameters <- rbind(out_hm$mcmc[[1]], out_hm$mcmc[[2]])
```

95% credible intervals for the thetas, a, and b

```r
m <- matrix(1:n_distinct(radonData$county), 5)

for (i in 1:ncol(m)) {
 cat(kable(
   apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975)))[, m[, i]],
   'latex', booktabs=TRUE) %>% kable_styling(full_width = T), "\\newline")
}
```

	theta[1]	theta[2]	theta[3]	theta[4]	theta[5]
2.5%	0.1409439	0.2257829	0.1277318	0.1349507	0.1306269
97.5%	0.3649637	0.3726055	0.3388855	0.3229623	0.3316196

	theta[6]	theta[7]	theta[8]	theta[9]	theta[10]
2.5%	0.1195928	0.1068191	0.1169876	0.1584039	0.1096339
97.5%	0.3147726	0.2255590	0.2994506	0.3583622	0.2669395
	theta[11]	theta[12]	theta[13]	theta[14]	theta[15]
2.5%	0.1284248	0.1106807	0.1370430	0.0963432	0.1333838
97.5%	0.3164950	0.2826806	0.3338951	0.2067971	0.3424031
	theta[16]	theta[17]	theta[18]	theta[19]	theta[20]
2.5%	0.1271137	0.1211317	0.1613246	0.1672238	0.1157098
97.5%	0.3508060	0.3106421	0.3559895	0.2602339	0.3015612
	theta[21]	theta[22]	theta[23]	theta[24]	theta[25]
2.5%	0.1262678	0.1459105	0.1243969	0.0938099	0.0971327
97.5%	0.2828610	0.3581912	0.3425589	0.2203787	0.2099231
	theta[26]	theta[27]	theta[28]	theta[29]	theta[30]
2.5%	0.1793713	0.1231150	0.1410037	0.1304539	0.1606151
97.5%	0.2572843	0.2989202	0.3574343	0.3457945	0.3564239
	theta[31]	theta[32]	theta[33]	theta[34]	theta[35]
2.5%	0.1008272	0.1284809	0.1034677	0.1237222	0.1640819
97.5%	0.2554265	0.3288866	0.2670118	0.3281388	0.4039450
	theta[36]	theta[37]	theta[38]	theta[39]	theta[40]
2.5%	0.0948040	0.1707338	0.1208777	0.1189898	0.0980978
97.5%	0.2601543	0.4015957	0.3079044	0.2968003	0.2554414
	theta[41]	theta[42]	theta[43]	theta[44]	theta[45]
2.5%	0.1125661	0.1171816	0.0928225	0.1334762	0.1248808
97.5%	0.2612971	0.3319811	0.2156636	0.3162983	0.2662140
	theta[46]	theta[47]	theta[48]	theta[49]	theta[50]
2.5%	0.1335828	0.1234228	0.1525160	0.1114810	0.1037591
97.5%	0.3324693	0.3395399	0.3521769	0.2377178	0.2939266
	theta[51]	theta[52]	theta[53]	theta[54]	theta[55]
2.5%	0.0980556	0.1093022	0.1268640	0.1563347	0.1255352
97.5%	0.2549378	0.2935940	0.3319063	0.2961334	0.2926237
	theta[56]	theta[57]	theta[58]	theta[59]	theta[60]
2.5%	0.1296016	0.1532950	0.1111011	0.1213825	0.1224386
97.5%	0.3362861	0.3745397	0.2850173	0.3106830	0.3328986
	theta[61]	theta[62]	theta[63]	theta[64]	theta[65]
2.5%	0.1821746	0.1031122	0.1217807	0.1130339	0.1169842
97.5%	0.3256145	0.2593622	0.3207413	0.2479634	0.3221450
	theta[66]	theta[67]	theta[68]	theta[69]	theta[70]
2.5%	0.1459804	0.1081730	0.1500207	0.1277628	0.1454329
97.5%	0.3057574	0.2313988	0.3487634	0.3279228	0.2725874

	theta[71]	theta[72]	theta[73]	theta[74]	theta[75]
2.5%	0.1251950	0.1131226	0.2528817	0.1349394	0.1199154
97.5%	0.2778651	0.3080352	0.3603422	0.3469588	0.3157826

	theta[76]	theta[77]	theta[78]	theta[79]	theta[80]
2.5%	0.1038050	0.1130604	0.1378820	0.1349206	0.1606935
97.5%	0.2699461	0.2660052	0.3434074	0.3462176	0.2657313

	theta[81]	theta[82]	theta[83]	theta[84]	theta[85]
2.5%	0.1013468	0.1082225	0.1218950	0.1255007	0.1233586
97.5%	0.2653728	0.3050636	0.2573754	0.2668189	0.3391528

```
cat(kable(
   apply(parameters, 2, function(x) quantile(x, c(0.025, 0.975)))[, 86:87],
   'latex', booktabs=TRUE) %>% kable_styling(full_width = T), "\\newline")
```

	a	b
2.5%	7.823484	36.67395
97.5%	35.114245	172.61305

Means of the thetas, a, and b

```
apply(parameters, 2, mean)
```

```
##    theta[1]   theta[2]   theta[3]   theta[4]   theta[5]   theta[6]   theta[7]
##   0.2360357  0.2937195  0.2188965  0.2181582  0.2170545  0.2062845  0.1624429
##    theta[8]   theta[9]  theta[10]  theta[11]  theta[12]  theta[13]  theta[14]
##   0.1982701  0.2446037  0.1817159  0.2113706  0.1883294  0.2223398  0.1483146
##   theta[15]  theta[16]  theta[17]  theta[18]  theta[19]  theta[20]  theta[21]
##   0.2240690  0.2229417  0.2047470  0.2457734  0.2111399  0.1983925  0.1967651
##   theta[22]  theta[23]  theta[24]  theta[25]  theta[26]  theta[27]  theta[28]
##   0.2366394  0.2177829  0.1524623  0.1498127  0.2166214  0.2021602  0.2333536
##   theta[29]  theta[30]  theta[31]  theta[32]  theta[33]  theta[34]  theta[35]
##   0.2219660  0.2459040  0.1709465  0.2147829  0.1777581  0.2126684  0.2635634
##   theta[36]  theta[37]  theta[38]  theta[39]  theta[40]  theta[41]  theta[42]
##   0.1703631  0.2670885  0.2043538  0.1984111  0.1699959  0.1807373  0.2101457
##   theta[43]  theta[44]  theta[45]  theta[46]  theta[47]  theta[48]  theta[49]
##   0.1494956  0.2147879  0.1895643  0.2199908  0.2168479  0.2385414  0.1702255
##   theta[50]  theta[51]  theta[52]  theta[53]  theta[54]  theta[55]  theta[56]
##   0.1885322  0.1709877  0.1920174  0.2154165  0.2203557  0.2014647  0.2189280
##   theta[57]  theta[58]  theta[59]  theta[60]  theta[61]  theta[62]  theta[63]
##   0.2469402  0.1899763  0.2049306  0.2138576  0.2477699  0.1742092  0.2088140
##   theta[64]  theta[65]  theta[66]  theta[67]  theta[68]  theta[69]  theta[70]
##   0.1752344  0.2064398  0.2180791  0.1652490  0.2358728  0.2159610  0.2045364
##   theta[71]  theta[72]  theta[73]  theta[74]  theta[75]  theta[76]  theta[77]
##   0.1945355  0.1999294  0.3040319  0.2257788  0.2063625  0.1800321  0.1828286
##   theta[78]  theta[79]  theta[80]  theta[81]  theta[82]  theta[83]  theta[84]
##   0.2264384  0.2244872  0.2097288  0.1762139  0.1959036  0.1841663  0.1897782
##   theta[85]          a          b
##   0.2155487 16.8135861 81.4487512
```