

# Case Study 2

Parker Johnson

11/8/2022

## Introduction

Radon levels have been a subject of concern for homeowners in Minnesota, as too high of radon levels can become dangerous and even lead to lung cancer. Therefore, having an understanding of radon levels in Minnesota and how they might differ within each county is important for everyone's well-being. In this study, we estimate county-level average radon concentrations, as well as the state-wide average radon concentration. From this, we then identify counties with unusually high radon concentrations.

## Methodology

For this estimation, a hierarchical model was used. This model seemed appropriate for the study, because a hierarchical model can acknowledge the grouping structure of the counties while still addressing the broader population of Minnesota at the same time.

I will assume that the log radon levels are normally distributed, since log levels can be both positive and negative. In this situation,  $Y_{ij}$  represents the  $i$ -th radon level for county  $j$ . This hierarchical model will need  $\mu_j$ , which is the mean of the log radon levels for county  $j$ . I will assume a common standard deviation across counties, since the spread of log radon values across the state doesn't seem likely to differ much across counties, and this allows for a simpler implementation. I will also work with precision, which is  $\frac{1}{\sigma^2}$ , within this model to allow for easier implementation before transferring this back to standard deviations. Precisions often have gamma priors; therefore, the hierarchical model will have the following elements:

Sampling model: For  $j = 1, \dots, 85$  and  $i = 1, \dots, n_j$ :  $Y_{ij} | \mu_j, \sigma \stackrel{\text{iid}}{\sim} \text{Normal}(\mu_j, \sigma)$

Prior for  $\mu_j$ , Stage 1:  $j = 1, \dots, 85$ :  $\mu_j | \mu, \tau \sim \text{Normal}(\mu, \tau)$

Prior for  $\mu_j$ , Stage 2:  $\mu, \tau \sim \pi(\mu, \tau)$

$1/\sigma^2 | a_\sigma, b_\sigma \sim \text{Gamma}(a_\sigma, b_\sigma)$

Where  $\mu$  and  $\tau$  are random hyperparameters and  $\pi(\mu, \tau)$  denotes an arbitrary joint hyperprior distribution.

To construct a prior on these hyperparameters, a typical approach for Normal models is to assign two independent prior distributions; a Normal distribution for the mean  $\mu$  and a Gamma distribution for the precision  $1/\tau^2$ . Therefore, we now have the following prior for  $\mu_j$  instead of  $\pi(\mu, \tau)$ :

$\mu | \mu_0, \gamma_0 \sim \text{Normal}(\mu_0, \gamma_0)$

$1/\tau^2 | a, b \sim \text{Normal}(a_\tau, b_\tau)$

We now need to choose values for  $a_\sigma, b_\sigma, \mu_0, \gamma_0, a_\tau$ , and  $b_\tau$ . According to [1](https://www.epa.gov/radon/what-epa-action-level-radon-and-what-does-it-mean#:~:text=The%20average%20indoor%20radon%20concentration,related%20lung%20cancers%20a%20year,the average indoor radon concentration for homes in the US is about 1.3 pCi/L. On the log scale, this value is approximately 0.26, so this will be the value for <math>\mu_0</math>. Since I don't know how variable this value is, I will choose <math>\gamma_0 = 1</math> to keep a fair amount of variability. Weakly informative priors are sensible to choose for <math>\tau</math> and <math>\sigma</math>, so I will choose <math>a_\sigma = 1, b_\sigma = 1, a_\tau = 1</math>, and <math>b_\tau = 1</math>. Therefore, the complete hierarchical model is as follows:</a></p></div><div data-bbox=)

Sampling model: For  $j = 1, \dots, 85$  and  $i = 1, \dots, n_j$ :

$$Y_{ij}|\mu_j, \sigma_j \stackrel{\text{iid}}{\sim} \text{Normal}(\mu_j, \sigma_j)$$

Prior for  $\mu_j$ , Stage 1:  $j = 1, \dots, 85$ :

$$\mu_j|\mu, \tau \sim \text{Normal}(\mu, \tau)$$

Prior for  $\mu_j$ , Stage 2: the hyperpriors:

$$\mu \sim \text{Normal}(0.26, 1)$$

$$1/\tau^2 \sim \text{Gamma}(1, 1)$$

Prior for  $\sigma$ :

$$1/\sigma^2 \sim \text{Gamma}(1, 1)$$

With the full model, JAGS can now be used to estimate the posterior distribution in order to estimate county-level and state-wide radon concentrations. The counties with unusually high radon concentrations were found by identifying the top 3 counties for their mean radon values, and by also identifying the top 3 counties for the 97.5th quantile for their radon values. The 97.5th quantile counties were identified on top of the mean radon values because this means that it's still possible for the mean log radon value to be this high in these given counties, so the uncertainty in these counties for a high mean value means that a threat for high radon values could still be present.

## Results

The posterior mean of the log radon level in each county from JAGS can be seen in Figure 1. From these posterior means, the largest estimated mean was in Blue Earth County, with a mean of 1.777 log pCi/L (which is equivalent to 5.909 pCi/L). The smallest estimated mean was in Lake County, with a mean of 0.643 log pCi/L (which is equivalent to 1.903 pCi/L).

The posterior mean of the log radon level across every county is approximately 1.317 log pCi/L (which is equivalent to 3.732 pCi/L), with a standard deviation of approximately 0.0557 log pCi/L. Therefore, there is a 95% chance that the state-wide average radon level is between 3.354 and 4.163 pCi/L.

The counties with the 3 highest mean radon values were Blue Earth, Lac Qui Parle, and Freeborn County, with respective means of 1.777 log pCi/L (or 5.909 pCi/L), 1.741 log pCi/L (or 5.703 pCi/L), and 1.740 log pCi/L (or 5.697 pCi/L). The counties with the 3 highest radon values for their 97.5th quantiles were Lac Qui Parle, Watonwan, and Nicollet, with respective quantile values of 2.416 log pCi/L (or 11.21 pCi/L), 2.317 log pCi/L (or 10.14 pCi/L), and 2.291 log pCi/L (or 0.888 pCi/L). It's worth noting that these 3 counties all had means greater than 1.70 log pCi/L as well.

## Discussion

The state-wide mean log radon concentration is approximately 1.317 log pCi/L (which is equivalent to 3.732 pCi/L), and a table of the average mean log radon concentrations in each county can be seen after the discussion section. From these results, we estimate that the counties with unusually high radon values include Blue Earth, Lac Qui Parle, Freeborn, Watonwan, and Nicollet.

According to <https://www.consumerreports.org/radon/is-it-safe-to-buy-a-home-with-an-elevated-radon-level/>, levels of 4 pCi/L, or 1.386 log pCi/L, or higher are considered hazardous. However, radon levels below this level can still pose a threat to the health of people living in that home. Since the state-wide mean radon concentration is fairly close to this level, efforts should be taken across the state as a whole to reduce radon levels in homes for the general health of Minnesotans. People living in the counties of Blue Earth, Lac Qui Parle, Freeborn, Watonwan, and Nicollet should be particularly cautious about radon gas, and should make sure that their home's radon levels are not hazardous.

One drawback to this design is the possibility for inaccuracies to arise when justifying the prior distribution and model. While a weakly informative prior does not have a large impact on the posterior distribution, it does sway it slightly, so an incorrect prior distribution will result in a slightly inaccurate posterior distribution

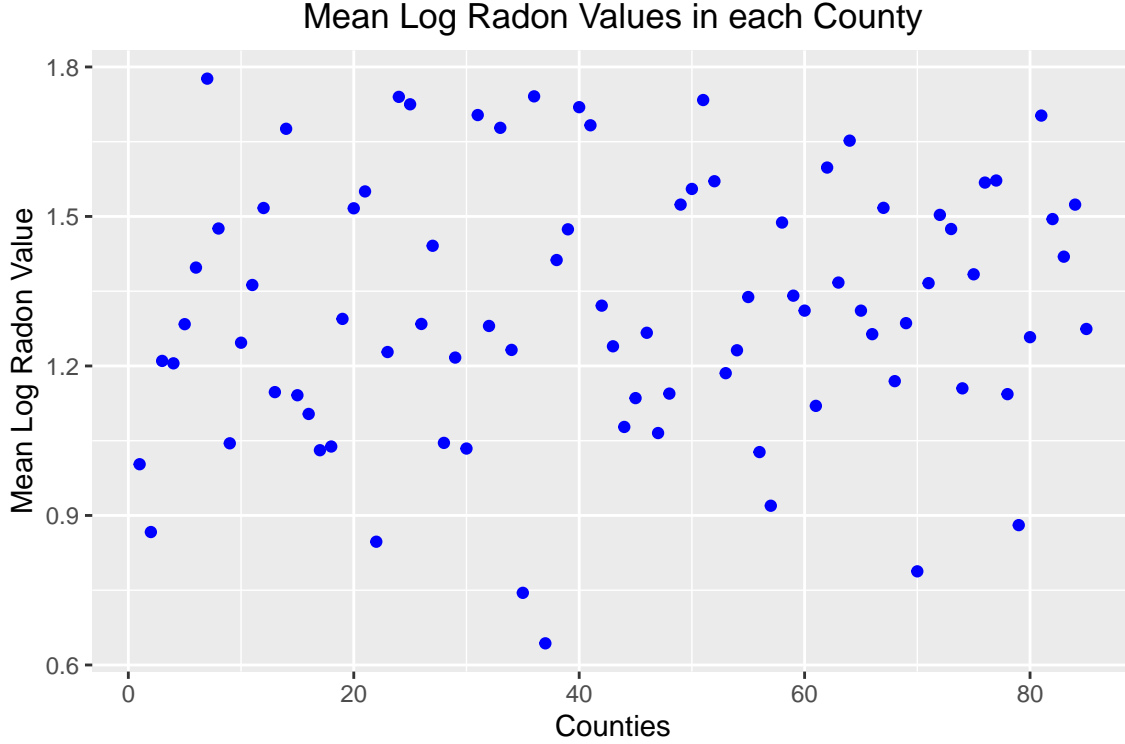


Figure 1: Scatterplot of mean log radon values in each county.

and interpretations. Since the country-wide radon mean level might differ heavily from Minnesota’s mean level, perhaps more research could have been conducted to find a more accurate prior distribution for Minnesota.

Another drawback to this design is that there isn’t a whole lot of observed data for many of the counties. Mahnomen, Murray, and Wilkin only had 1 observed radon level, with Murray and Wilkin’s value being very high. However, with only 1 observation, a good estimation of the mean radon level in those counties can’t be obtained, so it’s hard to know whether the radon levels in these counties is hazardous or not. Additionally, many other counties had less than 10 observations, so these lack of observations adds uncertainty to the posterior means that are obtained and the strength of the county-wide means are weakened. Therefore, if possible, redoing this study with more observations across each county would be beneficial to truly understanding radon levels within each county and across Minnesota.

Table 1: Table of mean log radon values in each county and their quantiles.

	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[1]	AITKIN	1.003	0.288	0.434	0.811	1.004	1.197	1.569
mu_j[2]	ANOKA	0.867	0.105	0.663	0.796	0.867	0.937	1.072
mu_j[3]	BECKER	1.210	0.299	0.620	1.006	1.212	1.417	1.777
mu_j[4]	BELTRAMI	1.205	0.239	0.743	1.044	1.207	1.365	1.677
mu_j[5]	BENTON	1.284	0.280	0.737	1.096	1.284	1.469	1.825
mu_j[6]	BIG STONE	1.397	0.298	0.823	1.195	1.395	1.599	1.983
mu_j[7]	BLUE EARTH	1.777	0.187	1.404	1.650	1.776	1.899	2.152
mu_j[8]	BROWN	1.476	0.283	0.933	1.284	1.471	1.669	2.026
mu_j[9]	CARLTON	1.045	0.213	0.626	0.900	1.046	1.189	1.463
mu_j[10]	CARVER	1.247	0.250	0.761	1.078	1.249	1.414	1.733

	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[11]	CASS	1.362	0.263	0.853	1.188	1.360	1.539	1.884
mu_j[12]	CHIPPEWA	1.517	0.281	0.957	1.333	1.520	1.705	2.066
mu_j[13]	CHISAGO	1.148	0.251	0.659	0.982	1.146	1.316	1.635
mu_j[14]	CLAY	1.676	0.185	1.304	1.553	1.676	1.801	2.041
mu_j[15]	CLEARWATER	1.141	0.279	0.584	0.947	1.141	1.329	1.695
mu_j[16]	COOK	1.104	0.325	0.465	0.888	1.113	1.321	1.736
mu_j[17]	COTTONWOOD	1.031	0.283	0.474	0.842	1.032	1.224	1.574
mu_j[18]	CROW WING	1.038	0.196	0.653	0.909	1.038	1.173	1.418
mu_j[19]	DAKOTA	1.294	0.099	1.102	1.228	1.293	1.363	1.488
mu_j[20]	DODGE	1.516	0.302	0.943	1.308	1.513	1.722	2.132
mu_j[21]	DOUGLAS	1.550	0.218	1.123	1.401	1.549	1.694	1.987
mu_j[22]	FARIBAULT	0.847	0.250	0.357	0.684	0.849	1.018	1.324
mu_j[23]	FILLMORE	1.228	0.324	0.586	1.017	1.226	1.446	1.861
mu_j[24]	FREEBORN	1.740	0.221	1.315	1.588	1.741	1.887	2.177
mu_j[25]	GOODHUE	1.725	0.189	1.362	1.595	1.724	1.854	2.090
mu_j[26]	HENNEPIN	1.284	0.076	1.140	1.233	1.285	1.335	1.432
mu_j[27]	HOUSTON	1.441	0.249	0.961	1.276	1.435	1.613	1.923
mu_j[28]	HUBBARD	1.046	0.263	0.511	0.874	1.046	1.227	1.550
mu_j[29]	ISANTI	1.217	0.303	0.622	1.015	1.215	1.420	1.816
mu_j[30]	ITASCA	1.034	0.208	0.621	0.895	1.037	1.176	1.443
mu_j[31]	JACKSON	1.704	0.258	1.206	1.524	1.710	1.876	2.207
mu_j[32]	KANABEC	1.280	0.274	0.757	1.094	1.279	1.461	1.817
mu_j[33]	KANDIYOHI	1.678	0.281	1.128	1.489	1.679	1.866	2.228
mu_j[34]	KITTSON	1.232	0.296	0.649	1.033	1.233	1.430	1.818
mu_j[35]	KOOCHICHINGO	0.745	0.244	0.253	0.581	0.747	0.913	1.211
mu_j[36]	LAC QUI PARLE	1.741	0.339	1.089	1.508	1.741	1.972	2.416
mu_j[37]	LAKE	0.643	0.227	0.198	0.488	0.647	0.796	1.077
mu_j[38]	LAKE OF THE WOODS	1.413	0.274	0.877	1.230	1.412	1.597	1.946
mu_j[39]	LE SUEUR	1.474	0.263	0.946	1.295	1.477	1.649	1.991
mu_j[40]	LINCOLN	1.719	0.284	1.164	1.531	1.714	1.907	2.284
mu_j[41]	LYON	1.683	0.231	1.241	1.519	1.678	1.843	2.127
mu_j[42]	MAHNOMEN	1.321	0.357	0.621	1.087	1.324	1.557	2.031
mu_j[43]	MARSHALL	1.239	0.221	0.804	1.093	1.237	1.388	1.663
mu_j[44]	MARTIN	1.078	0.237	0.617	0.918	1.077	1.237	1.546
mu_j[45]	MCLEOD	1.135	0.190	0.767	1.004	1.135	1.266	1.506
mu_j[46]	MEEKER	1.267	0.269	0.739	1.085	1.261	1.444	1.802
mu_j[47]	MILLE LACS	1.065	0.332	0.410	0.843	1.062	1.285	1.730
mu_j[48]	MORRISON	1.145	0.216	0.714	1.001	1.146	1.290	1.562
mu_j[49]	MOWER	1.524	0.191	1.147	1.395	1.524	1.653	1.899
mu_j[50]	MURRAY	1.555	0.361	0.856	1.316	1.546	1.796	2.277
mu_j[51]	NICOLLET	1.734	0.284	1.180	1.542	1.729	1.926	2.291
mu_j[52]	NOBLES	1.571	0.304	0.991	1.365	1.563	1.771	2.169
mu_j[53]	NORMAN	1.186	0.300	0.595	0.985	1.189	1.383	1.768
mu_j[54]	OLMSTED	1.231	0.151	0.936	1.130	1.231	1.334	1.526
mu_j[55]	OTTER TAIL	1.338	0.231	0.883	1.184	1.337	1.494	1.792

	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[56]	PENNINGTON	1.027	0.311	0.408	0.828	1.029	1.239	1.636
mu_j[57]	PINE	0.920	0.256	0.428	0.747	0.924	1.094	1.415
mu_j[58]	PIPESTONE	1.488	0.277	0.944	1.303	1.487	1.668	2.040
mu_j[59]	POLK	1.341	0.280	0.790	1.152	1.348	1.530	1.886
mu_j[60]	POPE	1.311	0.317	0.682	1.096	1.313	1.521	1.922
mu_j[61]	RAMSEY	1.120	0.132	0.860	1.030	1.119	1.211	1.379
mu_j[62]	REDWOOD	1.598	0.267	1.086	1.418	1.598	1.773	2.128
mu_j[63]	RENVILLE	1.367	0.302	0.773	1.167	1.366	1.566	1.979
mu_j[64]	RICE	1.652	0.203	1.248	1.518	1.655	1.786	2.050
mu_j[65]	ROCK	1.311	0.325	0.675	1.097	1.310	1.527	1.955
mu_j[66]	ROSEAU	1.264	0.188	0.886	1.139	1.262	1.391	1.629
mu_j[67]	SCOTT	1.517	0.194	1.142	1.388	1.518	1.648	1.900
mu_j[68]	SHERBURNE	1.170	0.228	0.727	1.014	1.168	1.323	1.609
mu_j[69]	SIBLEY	1.286	0.284	0.721	1.098	1.292	1.474	1.843
mu_j[70]	ST LOUIS	0.788	0.073	0.646	0.738	0.787	0.837	0.932
mu_j[71]	STEARNS	1.366	0.144	1.081	1.270	1.366	1.463	1.650
mu_j[72]	STEELE	1.503	0.214	1.077	1.361	1.505	1.642	1.932
mu_j[73]	STEVENS	1.475	0.326	0.852	1.253	1.463	1.693	2.127
mu_j[74]	SWIFT	1.155	0.280	0.603	0.966	1.159	1.341	1.708
mu_j[75]	TODD	1.384	0.300	0.785	1.187	1.386	1.582	1.958
mu_j[76]	TRAVERSE	1.568	0.284	1.018	1.372	1.568	1.760	2.132
mu_j[77]	WABASHA	1.572	0.239	1.109	1.413	1.572	1.731	2.044
mu_j[78]	WADENA	1.143	0.262	0.638	0.966	1.143	1.317	1.669
mu_j[79]	WASECA	0.881	0.285	0.319	0.693	0.887	1.073	1.423
mu_j[80]	WASHINGTON	1.258	0.113	1.039	1.183	1.257	1.332	1.484
mu_j[81]	WATONWAN	1.702	0.309	1.105	1.491	1.702	1.912	2.317
mu_j[82]	WILKIN	1.495	0.358	0.804	1.249	1.489	1.735	2.203
mu_j[83]	WINONA	1.419	0.193	1.044	1.285	1.420	1.551	1.801
mu_j[84]	WRIGHT	1.524	0.194	1.142	1.389	1.523	1.655	1.910
mu_j[85]	YELLOW MEDICINE	1.274	0.320	0.642	1.060	1.276	1.490	1.894

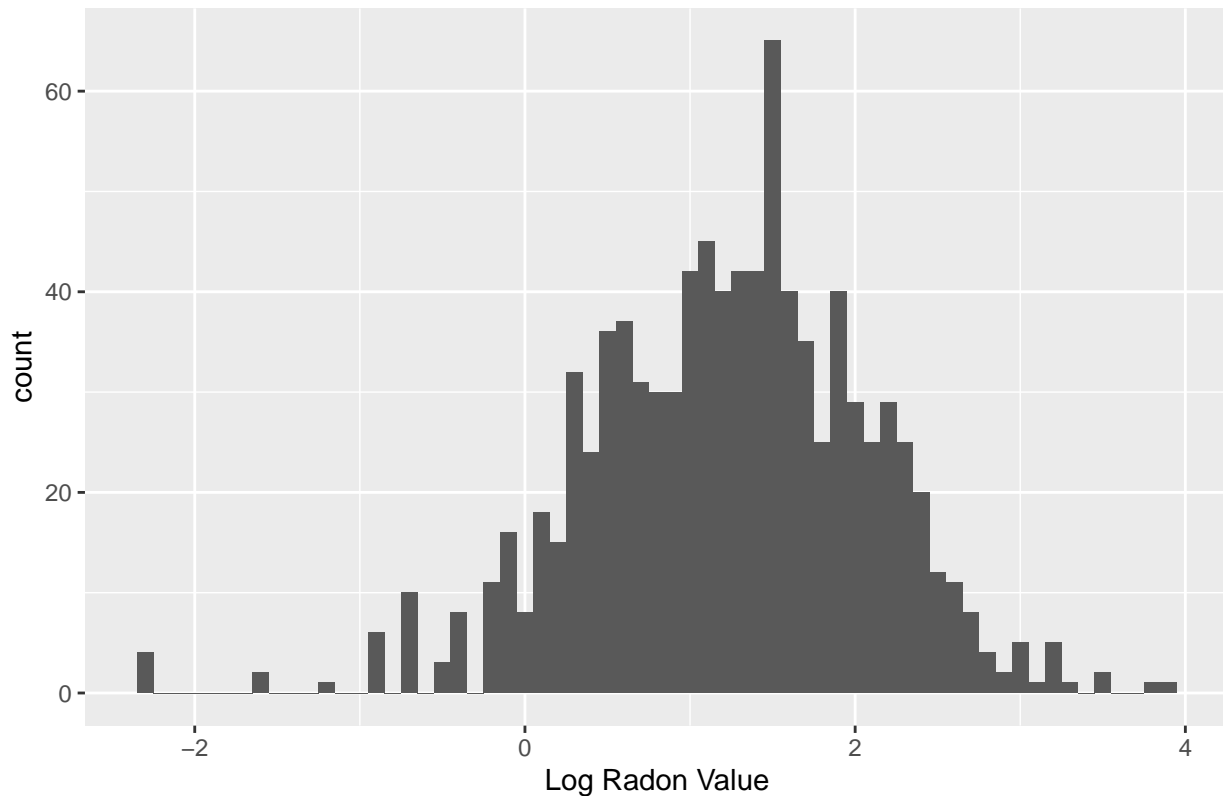
## Technical Appendix

I first loaded the data set and plotted the log radon values to understand how it was distributed. This further justified my choice of choosing a normal model for the log radon values, as the plot below indicates that the log radon values are normally distributed.

```
radon <- read.csv("https://aloy.rbind.io/data/radon.csv")

#Further justification for choosing a normal model, since values look normally
#distributed.
ggplot(data = radon, aes(x = log.radon)) +
  geom_histogram(binwidth = 0.1) +
  labs(x = "Log Radon Value", title = "Log Radon Values for Observed Data") +
  theme(plot.title = element_text(hjust = 0.5))
```

### Log Radon Values for Observed Data



*#Cleaner table of observed data for each county. Will use this for the counties #later.*

```
df_data <- radon %>%
  group_by(county) %>%
  mutate(std = sd(log.radon)) %>%
  summarize(county.name = unique(county.name), Mean = mean(log.radon),
            SD = mean(std))
```

Specifying the prior distributions within the model, I implemented the model in JAGS based on the hierarchical normal sampling example found in Probability and Bayesian Modeling (Albert and Hu, 2020).

```
modelString <- "
model {
  ## sampling
  for (i in 1:N){
    y[i] ~ dnorm(mu_j[county_num[i]], invsigma2)
  }
  ## priors
  for (j in 1:J){
    mu_j[j] ~ dnorm(mu, invtau2)
  }
  invsigma2 ~ dgamma(a_s, b_s)
  sigma <- sqrt(pow(invsigma2, -1))
  ## hyperpriors
  mu ~ dnorm(mu0, g0)
  invtau2 ~ dgamma(a_t, b_t)
  tau <- sqrt(pow(invtau2, -1))
}
```

```

"

#Defining the data
y <- radon$log.radon
county_num <- radon$county
N <- length(y)
J <- length(unique(county_num)) #The 85 counties
the_data <- list("y" = y, "county_num" = county_num,
                "N" = N, "J" = J,
                "mu0" = 0.26, "g0" = 1, #Prior values
                "a_t" = 1, "b_t" = 1,
                "a_s" = 1, "b_s" = 1)

#Function for making the data reproducible
initsfunction <- function(chain){
  .RNG.seed <- c(1,2)[chain]
  .RNG.name <- c("base::Super-Duper",
                "base::Wichmann-Hill")[chain]
  return(list(.RNG.seed=.RNG.seed,
              .RNG.name=.RNG.name))
}

#Matrix of simulated draws
posterior <- run.jags(modelString,
                      n.chains = 1,
                      data = the_data,
                      monitor = c("mu", "tau", "mu_j", "sigma"),
                      adapt = 1000,
                      burnin = 5000,
                      sample = 5000,
                      inits = initsfunction,
                      silent.jags = TRUE)

## 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

```

Diagnostics needed to be conducted to ensure that the JAGS sampler properly converged to a stationary mean. The trace plots indicate that the JAGS sampler converged fairly quickly to white noise, and the ACF plots indicate that there isn't high correlation since it tapers off quickly. The summary statistics are also given below for all of the variables in the model.

(Note: While not all  $\mu_j$ 's were plotted here due to the amount of space this would take up, all trace plots and ACF plots indicated well mixing and convergence.)

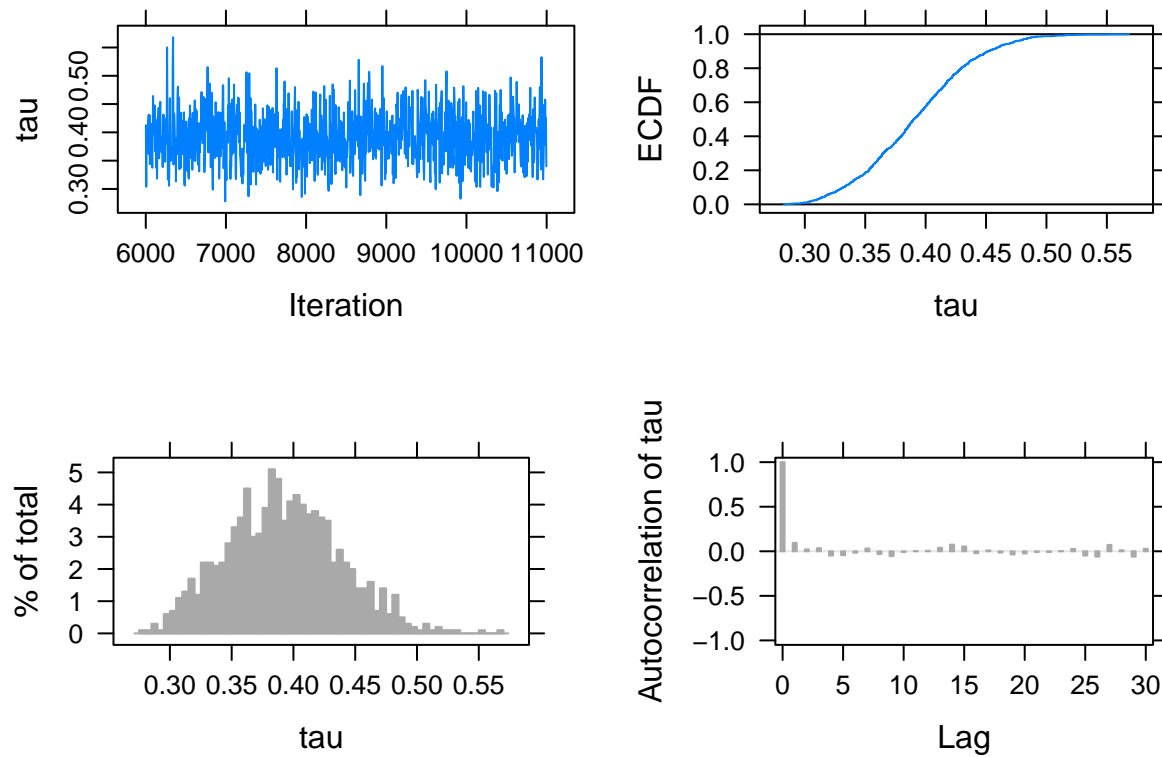
```

plot(posterior, vars = "tau")

## Generating summary statistics and plots (these will NOT be saved for
## reuse)...
## Calculating summary statistics...

## Warning: Convergence cannot be assessed with only 1 chain

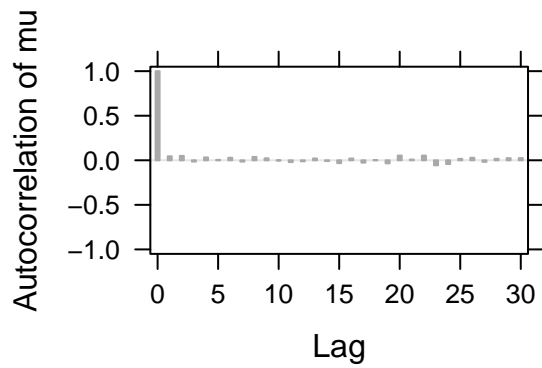
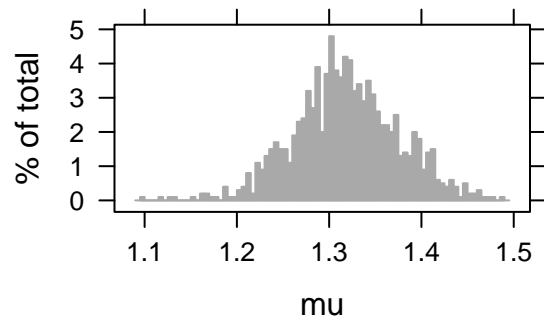
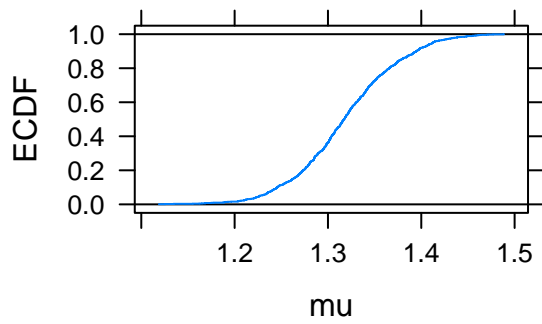
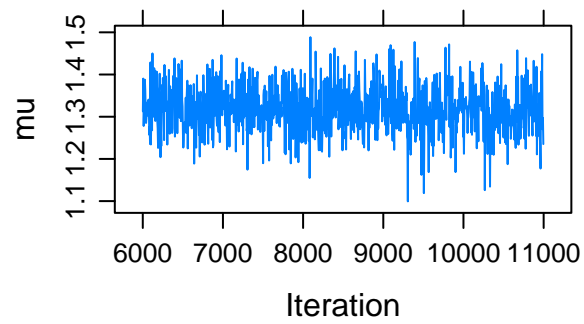
```



```
plot(posterior, vars = "mu[1]")
```

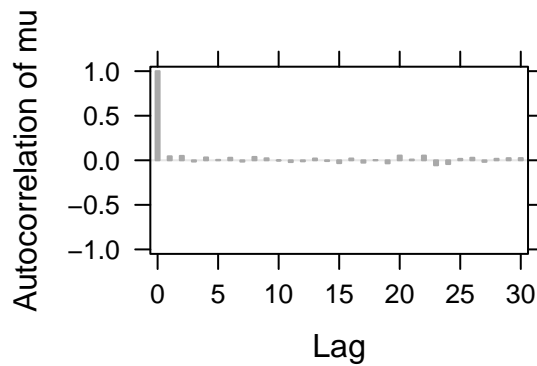
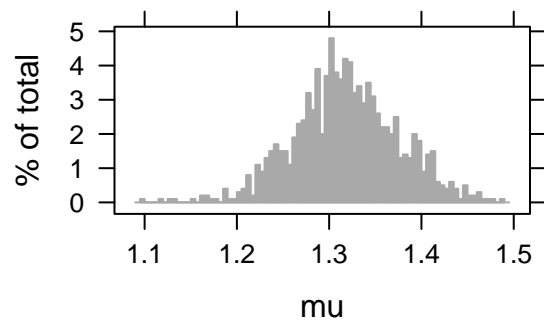
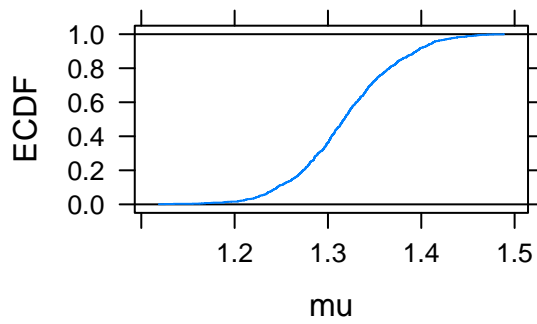
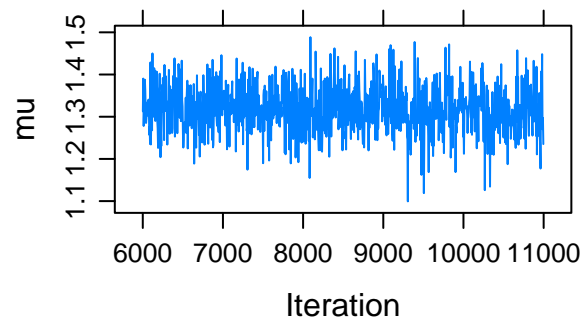
```
## Generating summary statistics and plots (these will NOT be saved for
## reuse)...
## Calculating summary statistics...
## Warning: Convergence cannot be assessed with only 1 chain
```





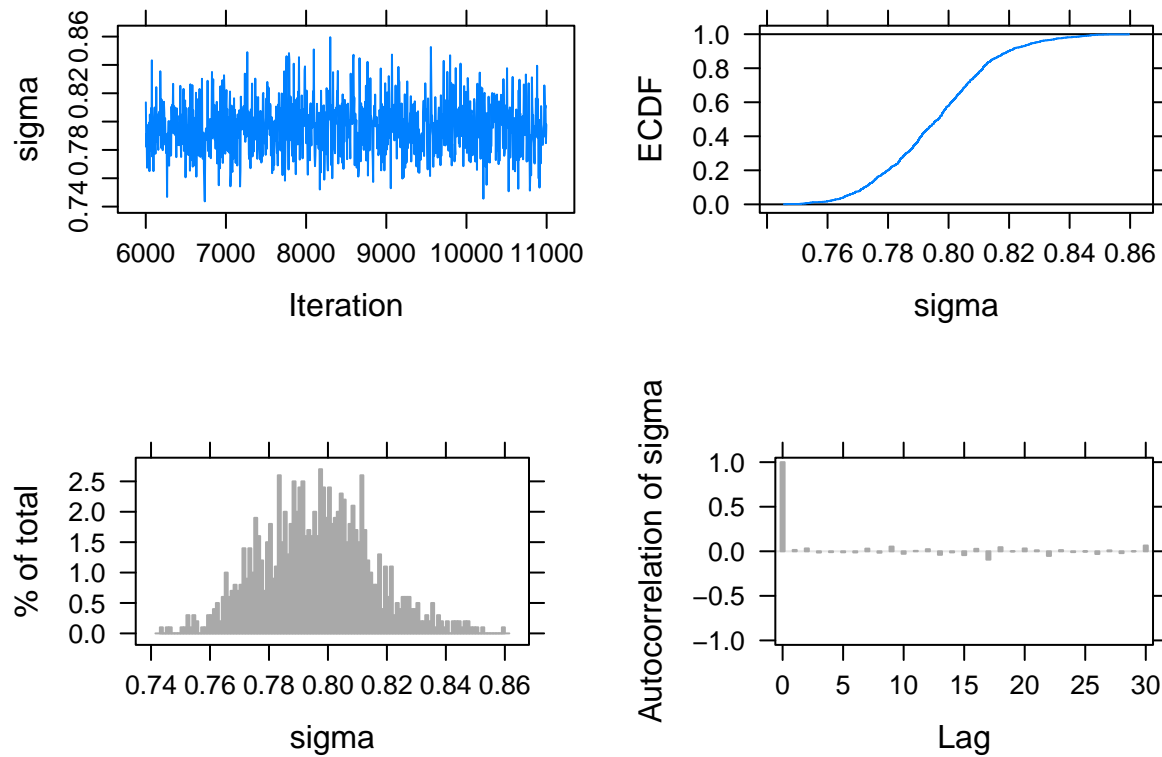
```
plot(posterior, vars = "mu[37]")
```

```
## Generating summary statistics and plots (these will NOT be saved for
## reuse)...
## Calculating summary statistics...
## Warning: Convergence cannot be assessed with only 1 chain
```



```
plot(posterior, vars = "sigma")
```

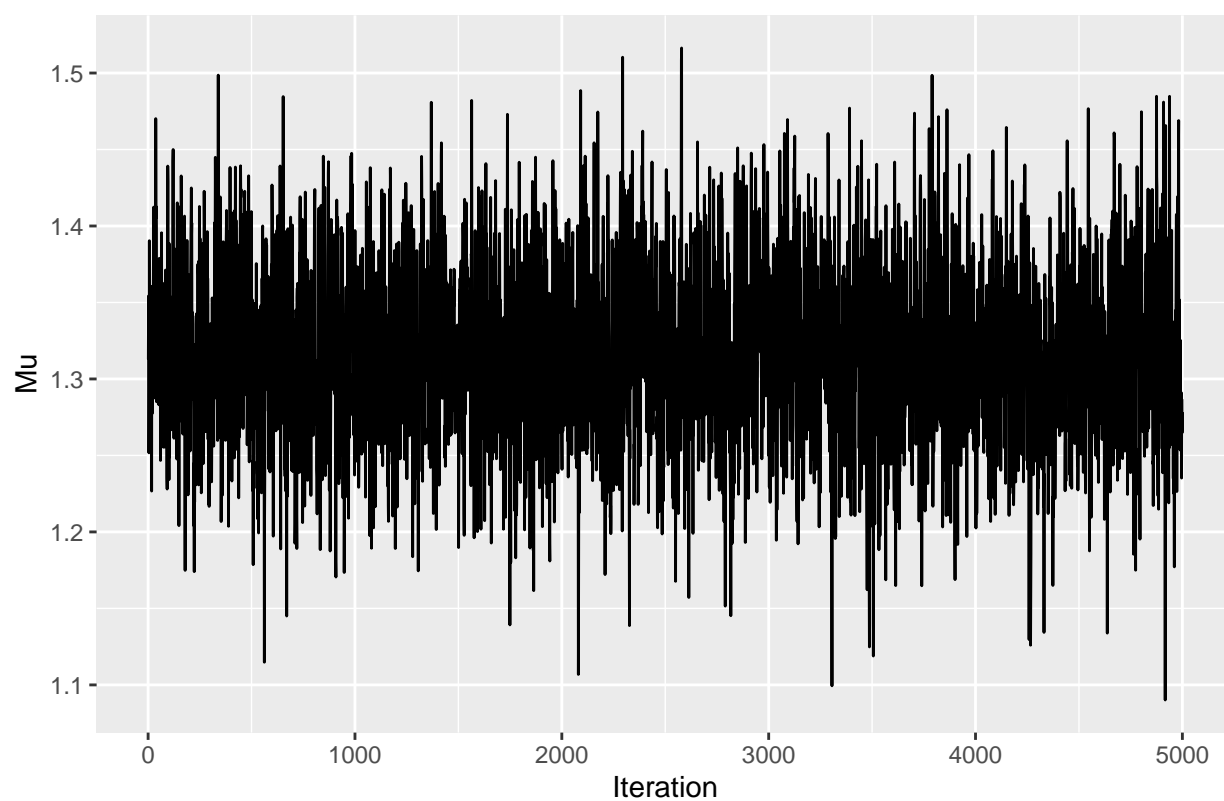
```
## Generating summary statistics and plots (these will NOT be saved for
## reuse)...
## Calculating summary statistics...
## Warning: Convergence cannot be assessed with only 1 chain
```



```
#Trace plot for mu
ggplot() +
  geom_line(aes(x = seq_along(posterior$mcmc[[1]][, 1]), y = posterior$mcmc[[1]][, 1])) +
  labs(x = "Iteration", y = "Mu", title = "Trace Plot for Mu") +
  theme(plot.title = element_text(hjust = 0.5))
```

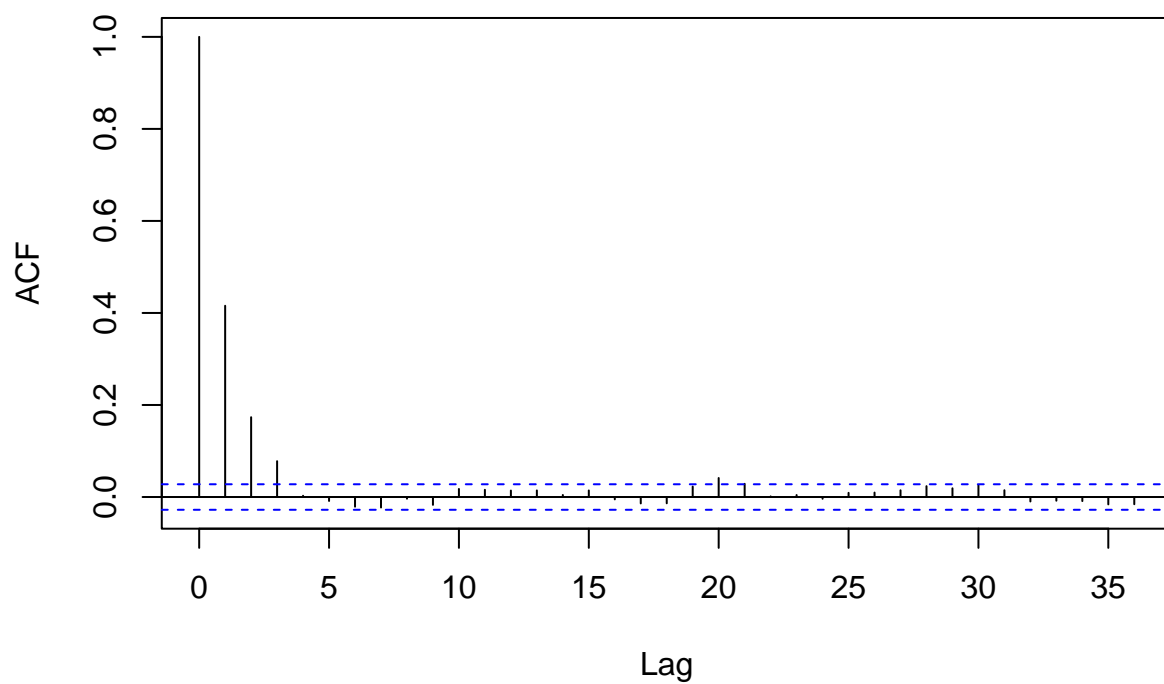
## Don't know how to automatically pick scale for object of type mcmc. Defaulting to continuous.

Trace Plot for Mu



```
#ACF plot for mu  
acf(posterior$mcmc[[1]][, 1], main = "ACF plot for Mu")
```

ACF plot for Mu



```
#Summary statistics
```

```
summary(posterior$mcmc)
```

```
##
## Iterations = 6001:11000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 5000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean      SD Naive SE Time-series SE
## mu      1.3169 0.05574 0.0007883      0.0011897
## tau      0.3929 0.04512 0.0006381      0.0011853
## mu_j[1]  1.0028 0.28837 0.0040781      0.0042123
## mu_j[2]  0.8667 0.10484 0.0014827      0.0014827
## mu_j[3]  1.2101 0.29949 0.0042354      0.0042354
## mu_j[4]  1.2052 0.23891 0.0033787      0.0033787
## mu_j[5]  1.2839 0.28030 0.0039641      0.0040694
## mu_j[6]  1.3974 0.29814 0.0042164      0.0042164
## mu_j[7]  1.7766 0.18734 0.0026494      0.0027881
## mu_j[8]  1.4758 0.28327 0.0040060      0.0042030
## mu_j[9]  1.0448 0.21280 0.0030095      0.0031201
## mu_j[10] 1.2467 0.25034 0.0035404      0.0035404
## mu_j[11] 1.3625 0.26320 0.0037222      0.0036573
## mu_j[12] 1.5170 0.28122 0.0039771      0.0039771
## mu_j[13] 1.1475 0.25082 0.0035471      0.0035471
## mu_j[14] 1.6759 0.18538 0.0026216      0.0026216
## mu_j[15] 1.1413 0.27937 0.0039508      0.0039508
## mu_j[16] 1.1037 0.32487 0.0045944      0.0047006
## mu_j[17] 1.0309 0.28348 0.0040090      0.0042860
## mu_j[18] 1.0383 0.19587 0.0027701      0.0027701
## mu_j[19] 1.2945 0.09871 0.0013959      0.0013959
## mu_j[20] 1.5163 0.30222 0.0042740      0.0044385
## mu_j[21] 1.5502 0.21848 0.0030898      0.0030898
## mu_j[22] 0.8473 0.25013 0.0035374      0.0036461
## mu_j[23] 1.2279 0.32403 0.0045825      0.0046964
## mu_j[24] 1.7400 0.22066 0.0031206      0.0032359
## mu_j[25] 1.7250 0.18864 0.0026677      0.0028509
## mu_j[26] 1.2843 0.07553 0.0010682      0.0010682
## mu_j[27] 1.4412 0.24899 0.0035213      0.0035213
## mu_j[28] 1.0456 0.26319 0.0037221      0.0037221
## mu_j[29] 1.2168 0.30348 0.0042918      0.0042918
## mu_j[30] 1.0343 0.20820 0.0029444      0.0029444
## mu_j[31] 1.7035 0.25777 0.0036454      0.0037352
## mu_j[32] 1.2803 0.27396 0.0038744      0.0039908
## mu_j[33] 1.6779 0.28145 0.0039803      0.0043170
## mu_j[34] 1.2322 0.29606 0.0041869      0.0041869
## mu_j[35] 0.7448 0.24402 0.0034509      0.0036979
## mu_j[36] 1.7410 0.33852 0.0047874      0.0051840
## mu_j[37] 0.6434 0.22718 0.0032128      0.0033953
## mu_j[38] 1.4125 0.27432 0.0038795      0.0038795
## mu_j[39] 1.4742 0.26277 0.0037162      0.0038108
```

```

## mu_j[40] 1.7193 0.28356 0.0040101 0.0043130
## mu_j[41] 1.6830 0.23107 0.0032678 0.0032678
## mu_j[42] 1.3210 0.35658 0.0050428 0.0050428
## mu_j[43] 1.2394 0.22062 0.0031201 0.0030236
## mu_j[44] 1.0776 0.23664 0.0033466 0.0033466
## mu_j[45] 1.1355 0.19029 0.0026910 0.0027580
## mu_j[46] 1.2666 0.26933 0.0038089 0.0038089
## mu_j[47] 1.0654 0.33213 0.0046970 0.0048525
## mu_j[48] 1.1445 0.21621 0.0030576 0.0030576
## mu_j[49] 1.5238 0.19083 0.0026988 0.0026988
## mu_j[50] 1.5552 0.36079 0.0051023 0.0054399
## mu_j[51] 1.7336 0.28405 0.0040170 0.0045264
## mu_j[52] 1.5707 0.30378 0.0042961 0.0044393
## mu_j[53] 1.1855 0.29981 0.0042400 0.0042400
## mu_j[54] 1.2315 0.15058 0.0021296 0.0021296
## mu_j[55] 1.3383 0.23071 0.0032627 0.0033436
## mu_j[56] 1.0272 0.31077 0.0043949 0.0043949
## mu_j[57] 0.9195 0.25576 0.0036170 0.0036170
## mu_j[58] 1.4878 0.27736 0.0039224 0.0039224
## mu_j[59] 1.3410 0.28019 0.0039624 0.0039624
## mu_j[60] 1.3111 0.31738 0.0044885 0.0044885
## mu_j[61] 1.1199 0.13208 0.0018679 0.0018679
## mu_j[62] 1.5981 0.26705 0.0037767 0.0038589
## mu_j[63] 1.3674 0.30154 0.0042645 0.0042645
## mu_j[64] 1.6521 0.20303 0.0028712 0.0029872
## mu_j[65] 1.3110 0.32480 0.0045933 0.0045933
## mu_j[66] 1.2638 0.18779 0.0026558 0.0026558
## mu_j[67] 1.5173 0.19421 0.0027465 0.0027465
## mu_j[68] 1.1695 0.22781 0.0032217 0.0031245
## mu_j[69] 1.2859 0.28421 0.0040193 0.0040193
## mu_j[70] 0.7879 0.07307 0.0010334 0.0010903
## mu_j[71] 1.3662 0.14402 0.0020367 0.0021732
## mu_j[72] 1.5031 0.21414 0.0030284 0.0030284
## mu_j[73] 1.4748 0.32579 0.0046074 0.0048431
## mu_j[74] 1.1551 0.28004 0.0039603 0.0039603
## mu_j[75] 1.3839 0.29971 0.0042385 0.0042385
## mu_j[76] 1.5680 0.28389 0.0040148 0.0041132
## mu_j[77] 1.5721 0.23876 0.0033766 0.0033766
## mu_j[78] 1.1435 0.26213 0.0037071 0.0037071
## mu_j[79] 0.8806 0.28537 0.0040357 0.0039493
## mu_j[80] 1.2578 0.11268 0.0015935 0.0015935
## mu_j[81] 1.7024 0.30885 0.0043678 0.0045966
## mu_j[82] 1.4947 0.35763 0.0050576 0.0050576
## mu_j[83] 1.4193 0.19311 0.0027310 0.0027471
## mu_j[84] 1.5238 0.19449 0.0027506 0.0028431
## mu_j[85] 1.2742 0.32008 0.0045265 0.0045265
## sigma 0.7959 0.01919 0.0002714 0.0002935
##
## 2. Quantiles for each variable:
##
##      2.5%    25%    50%    75%   97.5%
## mu      1.2100 1.2796 1.3163 1.3538 1.4262
## tau     0.3116 0.3609 0.3904 0.4217 0.4863
## mu_j[1] 0.4335 0.8108 1.0038 1.1973 1.5689

```

```

## mu_j[2] 0.6634 0.7963 0.8666 0.9373 1.0716
## mu_j[3] 0.6199 1.0060 1.2120 1.4174 1.7771
## mu_j[4] 0.7428 1.0437 1.2067 1.3652 1.6767
## mu_j[5] 0.7370 1.0957 1.2843 1.4690 1.8254
## mu_j[6] 0.8226 1.1953 1.3946 1.5988 1.9833
## mu_j[7] 1.4041 1.6495 1.7763 1.8991 2.1523
## mu_j[8] 0.9334 1.2836 1.4709 1.6687 2.0259
## mu_j[9] 0.6262 0.9002 1.0461 1.1892 1.4630
## mu_j[10] 0.7607 1.0778 1.2492 1.4140 1.7327
## mu_j[11] 0.8535 1.1881 1.3602 1.5386 1.8844
## mu_j[12] 0.9568 1.3331 1.5196 1.7048 2.0661
## mu_j[13] 0.6592 0.9825 1.1461 1.3164 1.6350
## mu_j[14] 1.3036 1.5531 1.6759 1.8010 2.0414
## mu_j[15] 0.5836 0.9473 1.1412 1.3292 1.6954
## mu_j[16] 0.4652 0.8885 1.1129 1.3211 1.7358
## mu_j[17] 0.4739 0.8423 1.0316 1.2242 1.5742
## mu_j[18] 0.6531 0.9086 1.0379 1.1730 1.4177
## mu_j[19] 1.1020 1.2278 1.2928 1.3633 1.4883
## mu_j[20] 0.9429 1.3085 1.5134 1.7219 2.1323
## mu_j[21] 1.1226 1.4013 1.5488 1.6938 1.9867
## mu_j[22] 0.3566 0.6844 0.8492 1.0178 1.3240
## mu_j[23] 0.5862 1.0165 1.2258 1.4460 1.8608
## mu_j[24] 1.3147 1.5876 1.7414 1.8873 2.1765
## mu_j[25] 1.3618 1.5947 1.7242 1.8539 2.0905
## mu_j[26] 1.1398 1.2328 1.2851 1.3352 1.4320
## mu_j[27] 0.9613 1.2756 1.4353 1.6130 1.9229
## mu_j[28] 0.5113 0.8745 1.0457 1.2274 1.5504
## mu_j[29] 0.6215 1.0149 1.2154 1.4198 1.8158
## mu_j[30] 0.6214 0.8948 1.0368 1.1755 1.4429
## mu_j[31] 1.2061 1.5244 1.7096 1.8759 2.2073
## mu_j[32] 0.7571 1.0944 1.2790 1.4613 1.8172
## mu_j[33] 1.1277 1.4895 1.6789 1.8661 2.2283
## mu_j[34] 0.6486 1.0334 1.2333 1.4298 1.8183
## mu_j[35] 0.2532 0.5807 0.7466 0.9133 1.2114
## mu_j[36] 1.0887 1.5080 1.7413 1.9725 2.4164
## mu_j[37] 0.1981 0.4881 0.6474 0.7956 1.0768
## mu_j[38] 0.8773 1.2303 1.4118 1.5970 1.9460
## mu_j[39] 0.9458 1.2952 1.4774 1.6492 1.9909
## mu_j[40] 1.1638 1.5315 1.7144 1.9075 2.2844
## mu_j[41] 1.2407 1.5188 1.6777 1.8425 2.1272
## mu_j[42] 0.6206 1.0866 1.3241 1.5570 2.0312
## mu_j[43] 0.8037 1.0927 1.2375 1.3883 1.6629
## mu_j[44] 0.6166 0.9180 1.0771 1.2366 1.5459
## mu_j[45] 0.7665 1.0043 1.1355 1.2656 1.5060
## mu_j[46] 0.7392 1.0853 1.2614 1.4443 1.8016
## mu_j[47] 0.4098 0.8435 1.0616 1.2846 1.7303
## mu_j[48] 0.7144 1.0012 1.1458 1.2904 1.5622
## mu_j[49] 1.1467 1.3952 1.5240 1.6534 1.8994
## mu_j[50] 0.8558 1.3161 1.5459 1.7958 2.2773
## mu_j[51] 1.1801 1.5420 1.7287 1.9263 2.2913
## mu_j[52] 0.9906 1.3645 1.5631 1.7710 2.1688
## mu_j[53] 0.5952 0.9852 1.1890 1.3834 1.7684
## mu_j[54] 0.9358 1.1299 1.2314 1.3343 1.5264
## mu_j[55] 0.8825 1.1838 1.3367 1.4936 1.7918

```

```
## mu_j[56] 0.4084 0.8278 1.0290 1.2389 1.6356
## mu_j[57] 0.4277 0.7465 0.9239 1.0936 1.4154
## mu_j[58] 0.9444 1.3030 1.4869 1.6681 2.0404
## mu_j[59] 0.7904 1.1516 1.3477 1.5298 1.8856
## mu_j[60] 0.6825 1.0964 1.3127 1.5215 1.9219
## mu_j[61] 0.8598 1.0303 1.1192 1.2109 1.3792
## mu_j[62] 1.0856 1.4175 1.5984 1.7734 2.1281
## mu_j[63] 0.7726 1.1675 1.3660 1.5656 1.9790
## mu_j[64] 1.2475 1.5180 1.6548 1.7855 2.0499
## mu_j[65] 0.6747 1.0966 1.3102 1.5266 1.9555
## mu_j[66] 0.8858 1.1389 1.2619 1.3912 1.6293
## mu_j[67] 1.1417 1.3882 1.5178 1.6479 1.9001
## mu_j[68] 0.7273 1.0144 1.1682 1.3227 1.6092
## mu_j[69] 0.7207 1.0983 1.2922 1.4737 1.8433
## mu_j[70] 0.6464 0.7378 0.7873 0.8371 0.9321
## mu_j[71] 1.0808 1.2702 1.3659 1.4635 1.6502
## mu_j[72] 1.0769 1.3610 1.5048 1.6423 1.9320
## mu_j[73] 0.8525 1.2533 1.4628 1.6927 2.1267
## mu_j[74] 0.6025 0.9657 1.1590 1.3410 1.7080
## mu_j[75] 0.7849 1.1871 1.3860 1.5821 1.9580
## mu_j[76] 1.0184 1.3718 1.5683 1.7595 2.1323
## mu_j[77] 1.1094 1.4132 1.5716 1.7315 2.0441
## mu_j[78] 0.6378 0.9656 1.1431 1.3173 1.6686
## mu_j[79] 0.3189 0.6934 0.8865 1.0731 1.4230
## mu_j[80] 1.0391 1.1828 1.2570 1.3322 1.4840
## mu_j[81] 1.1045 1.4915 1.7023 1.9121 2.3165
## mu_j[82] 0.8044 1.2488 1.4892 1.7349 2.2034
## mu_j[83] 1.0442 1.2853 1.4201 1.5513 1.8012
## mu_j[84] 1.1419 1.3894 1.5230 1.6554 1.9098
## mu_j[85] 0.6417 1.0601 1.2764 1.4895 1.8943
## sigma    0.7597 0.7829 0.7954 0.8086 0.8354
```

To further confirm that this JAGS model was sufficient for analysis, further diagnostics were run, including the Geweke diagnostic and checking the effective sample size for each variable. Since all Geweke values were less than an absolute value of 3, this indicates that each chain converged, and since all variables had effective sample sizes above 1000, the sampler ran long enough after convergence.

```
geweke.diag(posterior$mcmc[[1]]) #None above absolute value of 3, so good
```

```
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##      mu      tau  mu_j[1]  mu_j[2]  mu_j[3]  mu_j[4]  mu_j[5]
## 2.7631367 -0.4950004 0.3416592 0.6396625 2.0334775 1.1693911 0.0821291
## mu_j[6]  mu_j[7]  mu_j[8]  mu_j[9]  mu_j[10] mu_j[11] mu_j[12]
## 0.6207804 -1.4041347 1.0264425 -0.0352510 -0.2962793 0.6754019 1.8939670
## mu_j[13] mu_j[14] mu_j[15] mu_j[16] mu_j[17] mu_j[18] mu_j[19]
## 1.2672327 0.9856976 1.9657689 1.6500755 0.2193230 0.8320952 1.5092567
## mu_j[20] mu_j[21] mu_j[22] mu_j[23] mu_j[24] mu_j[25] mu_j[26]
## -0.4014087 0.3496608 2.6531607 1.0242747 1.7216999 -1.1665997 -1.1614876
## mu_j[27] mu_j[28] mu_j[29] mu_j[30] mu_j[31] mu_j[32] mu_j[33]
## 0.8305031 -0.0008746 0.4551890 -1.4946198 0.6371079 -1.6416665 1.3020836
## mu_j[34] mu_j[35] mu_j[36] mu_j[37] mu_j[38] mu_j[39] mu_j[40]
## 1.5957822 0.4630229 -1.1055419 0.9460089 1.3475046 -0.3093434 1.3485105
```



```
## mu_j[41] mu_j[42] mu_j[43] mu_j[44] mu_j[45] mu_j[46] mu_j[47]
## 1.4316186 1.4969981 -0.5384299 1.8296554 0.0322160 1.6151112 0.4497178
## mu_j[48] mu_j[49] mu_j[50] mu_j[51] mu_j[52] mu_j[53] mu_j[54]
## 0.6994350 1.6856955 0.4531544 -0.0979712 0.1347944 0.5949917 -0.5630359
## mu_j[55] mu_j[56] mu_j[57] mu_j[58] mu_j[59] mu_j[60] mu_j[61]
## 0.4017213 -0.2976374 0.0205106 2.1237267 1.7734834 0.3862862 2.7952651
## mu_j[62] mu_j[63] mu_j[64] mu_j[65] mu_j[66] mu_j[67] mu_j[68]
## 0.7478526 0.7486450 1.5095193 1.0582221 -0.4843325 1.1948891 1.0019241
## mu_j[69] mu_j[70] mu_j[71] mu_j[72] mu_j[73] mu_j[74] mu_j[75]
## 1.1354129 -0.8731207 -0.1492714 -1.2050853 0.7616964 2.1981069 1.3632904
## mu_j[76] mu_j[77] mu_j[78] mu_j[79] mu_j[80] mu_j[81] mu_j[82]
## 1.9643369 0.6222181 0.3671058 -0.0094087 1.3619833 1.2443010 1.0590092
## mu_j[83] mu_j[84] mu_j[85] sigma
## 0.6725282 -0.3719174 0.4512854 -0.2654067
```

```
effectiveSize(posterior$mcmc[[1]]) #All above 1000, good
```

```
## mu tau mu_j[1] mu_j[2] mu_j[3] mu_j[4] mu_j[5] mu_j[6]
## 2195.014 1448.975 4686.541 5000.000 5000.000 5000.000 4744.543 5000.000
## mu_j[7] mu_j[8] mu_j[9] mu_j[10] mu_j[11] mu_j[12] mu_j[13] mu_j[14]
## 4514.694 4542.240 4651.766 5000.000 5178.847 5000.000 5000.000 5000.000
## mu_j[15] mu_j[16] mu_j[17] mu_j[18] mu_j[19] mu_j[20] mu_j[21] mu_j[22]
## 5000.000 4776.630 4374.664 5000.000 5000.000 4636.253 5000.000 4706.330
## mu_j[23] mu_j[24] mu_j[25] mu_j[26] mu_j[27] mu_j[28] mu_j[29] mu_j[30]
## 4760.563 4650.161 4378.028 5000.000 5000.000 5000.000 5000.000 5000.000
## mu_j[31] mu_j[32] mu_j[33] mu_j[34] mu_j[35] mu_j[36] mu_j[37] mu_j[38]
## 4762.353 4712.598 4250.495 5000.000 4354.304 4264.192 4476.896 5000.000
## mu_j[39] mu_j[40] mu_j[41] mu_j[42] mu_j[43] mu_j[44] mu_j[45] mu_j[46]
## 4754.708 4322.439 5000.000 5000.000 5324.276 5000.000 4760.233 5000.000
## mu_j[47] mu_j[48] mu_j[49] mu_j[50] mu_j[51] mu_j[52] mu_j[53] mu_j[54]
## 4684.676 5000.000 5000.000 4398.710 3937.984 4682.692 5000.000 5000.000
## mu_j[55] mu_j[56] mu_j[57] mu_j[58] mu_j[59] mu_j[60] mu_j[61] mu_j[62]
## 4761.010 5000.000 5000.000 5000.000 5000.000 5000.000 5000.000 4789.305
## mu_j[63] mu_j[64] mu_j[65] mu_j[66] mu_j[67] mu_j[68] mu_j[69] mu_j[70]
## 5000.000 4619.434 5000.000 5000.000 5000.000 5316.058 5000.000 4491.399
## mu_j[71] mu_j[72] mu_j[73] mu_j[74] mu_j[75] mu_j[76] mu_j[77] mu_j[78]
## 4391.719 5000.000 4525.145 5000.000 5000.000 4763.557 5000.000 5000.000
## mu_j[79] mu_j[80] mu_j[81] mu_j[82] mu_j[83] mu_j[84] mu_j[85] sigma
## 5221.261 5000.000 4514.630 5000.000 4941.454 4679.808 5000.000 4275.639
```

To fully check the adequacy of this model, we need to check how well the data matches the model. This can be done through posterior predictive checking, in which we simulate a number of replicated datasets from the posterior predictive distribution and see how the observed sample compares to the replications. If the observed data resembles the replications, we can say that the observed data is consistent with predicted data from the model.

To check some function that will help us detect possible discrepancies between the observed data and the simulated prediction samples, I used the monimum value as the checking function. The histogram is shown below of the observed minimum in blue versus the minimum of each simulated prediction samples. The observed minimum of our data could plausibly have been generated from this distribution, so the model appears to be adequate for our data.

```
post_df <- as.data.frame(posterior$mcmc[[1]])

postpred_sim <- function(j){
  mu_j = rnorm(nrow(post_df), post_df$mu, post_df$tau) # generate mu_j
```

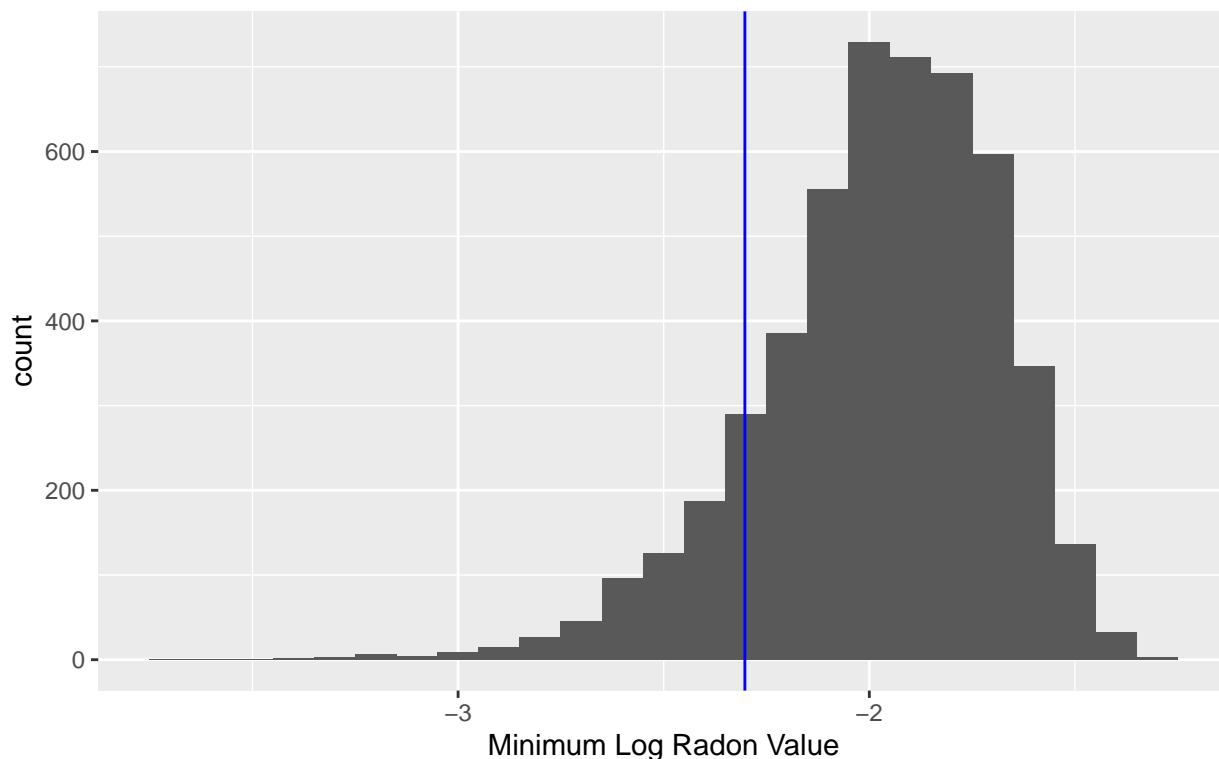
```

  y_pred = rnorm(nrow(post_df), mu_j, post_df$sigma)
}
set.seed(597)
ypred <- t(sapply(1:5000, postpred_sim)) #Doing this 5000 times
postpred_min <- apply(ypred, 1, min)
#Finding the minimum value for each replicated dataset

#Plotting the observed minimum versus the minimum from each replicated dataset
ggplot(data = NULL, aes(x = postpred_min)) +
  geom_histogram(binwidth = 0.1) +
  geom_vline(xintercept = min(radon$log.radon), color = "blue") +
  labs(x = "Minimum Log Radon Value", title = "Posterior Predictive Distribution of
  Minimum Log Radon Values") +
  theme(plot.title = element_text(hjust = 0.5))

```

Posterior Predictive Distribution of  
Minimum Log Radon Values



I then made a clean table of the summary statistics for each county.

```

df_summary <- data.frame(unclass(summary(posterior$mcmc))) #Data frame of summary
df_summary2 <- df_summary[-c(1, 2, 88), ] %>% #Data frame of just counties
  select(-c(start, end, thin, nchain)) #Removing unnecessary columns

#Getting column names from data frame made earlier
county_names <- df_data %>%
  select(county.name)

#Adding county names to summary data frame and removing more columns
county_summary <- df_summary2 %>%

```

```

mutate(county = county_names$county.name) %>%
select(-c(statistics.Naive.SE, statistics.Time.series.SE))

#Reordering and renaming the data frame
county_summary <- county_summary[, c(8, 1, 2, 3, 4, 5, 6, 7)] %>%
  rename(Mean = statistics.Mean, SD = statistics.SD,
    `2.5th quantile` = quantiles.2.5., `25th quantile` = quantiles.25.,
    `50th quantile` = quantiles.50., `75th quantile` = quantiles.75.,
    `97.5th quantile` = quantiles.97.5.)
knitr::kable(county_summary, digits = 3)

```

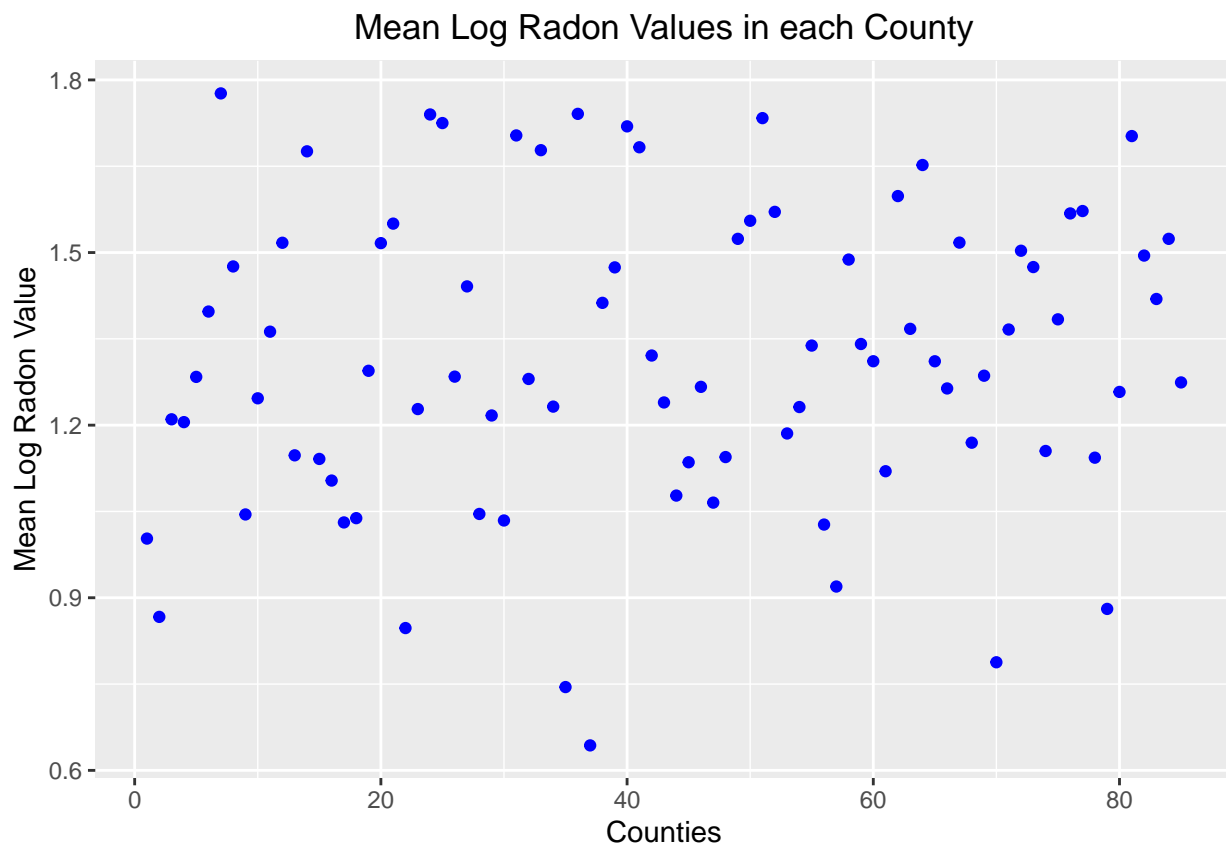
	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[1]	AITKIN	1.003	0.288	0.434	0.811	1.004	1.197	1.569
mu_j[2]	ANOKA	0.867	0.105	0.663	0.796	0.867	0.937	1.072
mu_j[3]	BECKER	1.210	0.299	0.620	1.006	1.212	1.417	1.777
mu_j[4]	BELTRAMI	1.205	0.239	0.743	1.044	1.207	1.365	1.677
mu_j[5]	BENTON	1.284	0.280	0.737	1.096	1.284	1.469	1.825
mu_j[6]	BIG STONE	1.397	0.298	0.823	1.195	1.395	1.599	1.983
mu_j[7]	BLUE EARTH	1.777	0.187	1.404	1.650	1.776	1.899	2.152
mu_j[8]	BROWN	1.476	0.283	0.933	1.284	1.471	1.669	2.026
mu_j[9]	CARLTON	1.045	0.213	0.626	0.900	1.046	1.189	1.463
mu_j[10]	CARVER	1.247	0.250	0.761	1.078	1.249	1.414	1.733
mu_j[11]	CASS	1.362	0.263	0.853	1.188	1.360	1.539	1.884
mu_j[12]	CHIPPEWA	1.517	0.281	0.957	1.333	1.520	1.705	2.066
mu_j[13]	CHISAGO	1.148	0.251	0.659	0.982	1.146	1.316	1.635
mu_j[14]	CLAY	1.676	0.185	1.304	1.553	1.676	1.801	2.041
mu_j[15]	CLEARWATER	1.141	0.279	0.584	0.947	1.141	1.329	1.695
mu_j[16]	COOK	1.104	0.325	0.465	0.888	1.113	1.321	1.736
mu_j[17]	COTTONWOOD	1.031	0.283	0.474	0.842	1.032	1.224	1.574
mu_j[18]	CROW WING	1.038	0.196	0.653	0.909	1.038	1.173	1.418
mu_j[19]	DAKOTA	1.294	0.099	1.102	1.228	1.293	1.363	1.488
mu_j[20]	DODGE	1.516	0.302	0.943	1.308	1.513	1.722	2.132
mu_j[21]	DOUGLAS	1.550	0.218	1.123	1.401	1.549	1.694	1.987
mu_j[22]	FARIBAULT	0.847	0.250	0.357	0.684	0.849	1.018	1.324
mu_j[23]	FILLMORE	1.228	0.324	0.586	1.017	1.226	1.446	1.861
mu_j[24]	FREEBORN	1.740	0.221	1.315	1.588	1.741	1.887	2.177
mu_j[25]	GOODHUE	1.725	0.189	1.362	1.595	1.724	1.854	2.090
mu_j[26]	HENNEPIN	1.284	0.076	1.140	1.233	1.285	1.335	1.432
mu_j[27]	HOUSTON	1.441	0.249	0.961	1.276	1.435	1.613	1.923
mu_j[28]	HUBBARD	1.046	0.263	0.511	0.874	1.046	1.227	1.550
mu_j[29]	ISANTI	1.217	0.303	0.622	1.015	1.215	1.420	1.816
mu_j[30]	ITASCA	1.034	0.208	0.621	0.895	1.037	1.176	1.443
mu_j[31]	JACKSON	1.704	0.258	1.206	1.524	1.710	1.876	2.207
mu_j[32]	KANABEC	1.280	0.274	0.757	1.094	1.279	1.461	1.817
mu_j[33]	KANDIYOHI	1.678	0.281	1.128	1.489	1.679	1.866	2.228
mu_j[34]	KITTSO	1.232	0.296	0.649	1.033	1.233	1.430	1.818
mu_j[35]	KOOCHICHINGO	0.745	0.244	0.253	0.581	0.747	0.913	1.211
mu_j[36]	LAC QUI PARLE	1.741	0.339	1.089	1.508	1.741	1.972	2.416
mu_j[37]	LAKE	0.643	0.227	0.198	0.488	0.647	0.796	1.077

	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[38]	LAKE OF THE WOODS	1.413	0.274	0.877	1.230	1.412	1.597	1.946
mu_j[39]	LE SUEUR	1.474	0.263	0.946	1.295	1.477	1.649	1.991
mu_j[40]	LINCOLN	1.719	0.284	1.164	1.531	1.714	1.907	2.284
mu_j[41]	LYON	1.683	0.231	1.241	1.519	1.678	1.843	2.127
mu_j[42]	MAHNOMEN	1.321	0.357	0.621	1.087	1.324	1.557	2.031
mu_j[43]	MARSHALL	1.239	0.221	0.804	1.093	1.237	1.388	1.663
mu_j[44]	MARTIN	1.078	0.237	0.617	0.918	1.077	1.237	1.546
mu_j[45]	MCLEOD	1.135	0.190	0.767	1.004	1.135	1.266	1.506
mu_j[46]	MEEKER	1.267	0.269	0.739	1.085	1.261	1.444	1.802
mu_j[47]	MILLE LACS	1.065	0.332	0.410	0.843	1.062	1.285	1.730
mu_j[48]	MORRISON	1.145	0.216	0.714	1.001	1.146	1.290	1.562
mu_j[49]	MOWER	1.524	0.191	1.147	1.395	1.524	1.653	1.899
mu_j[50]	MURRAY	1.555	0.361	0.856	1.316	1.546	1.796	2.277
mu_j[51]	NICOLLET	1.734	0.284	1.180	1.542	1.729	1.926	2.291
mu_j[52]	NOBLES	1.571	0.304	0.991	1.365	1.563	1.771	2.169
mu_j[53]	NORMAN	1.186	0.300	0.595	0.985	1.189	1.383	1.768
mu_j[54]	OLMSTED	1.231	0.151	0.936	1.130	1.231	1.334	1.526
mu_j[55]	OTTER TAIL	1.338	0.231	0.883	1.184	1.337	1.494	1.792
mu_j[56]	PENNINGTON	1.027	0.311	0.408	0.828	1.029	1.239	1.636
mu_j[57]	PINE	0.920	0.256	0.428	0.747	0.924	1.094	1.415
mu_j[58]	PIPESTONE	1.488	0.277	0.944	1.303	1.487	1.668	2.040
mu_j[59]	POLK	1.341	0.280	0.790	1.152	1.348	1.530	1.886
mu_j[60]	POPE	1.311	0.317	0.682	1.096	1.313	1.521	1.922
mu_j[61]	RAMSEY	1.120	0.132	0.860	1.030	1.119	1.211	1.379
mu_j[62]	REDWOOD	1.598	0.267	1.086	1.418	1.598	1.773	2.128
mu_j[63]	RENVILLE	1.367	0.302	0.773	1.167	1.366	1.566	1.979
mu_j[64]	RICE	1.652	0.203	1.248	1.518	1.655	1.786	2.050
mu_j[65]	ROCK	1.311	0.325	0.675	1.097	1.310	1.527	1.955
mu_j[66]	ROSEAU	1.264	0.188	0.886	1.139	1.262	1.391	1.629
mu_j[67]	SCOTT	1.517	0.194	1.142	1.388	1.518	1.648	1.900
mu_j[68]	SHERBURNE	1.170	0.228	0.727	1.014	1.168	1.323	1.609
mu_j[69]	SIBLEY	1.286	0.284	0.721	1.098	1.292	1.474	1.843
mu_j[70]	ST LOUIS	0.788	0.073	0.646	0.738	0.787	0.837	0.932
mu_j[71]	STEARNS	1.366	0.144	1.081	1.270	1.366	1.463	1.650
mu_j[72]	STEELE	1.503	0.214	1.077	1.361	1.505	1.642	1.932
mu_j[73]	STEVENS	1.475	0.326	0.852	1.253	1.463	1.693	2.127
mu_j[74]	SWIFT	1.155	0.280	0.603	0.966	1.159	1.341	1.708
mu_j[75]	TODD	1.384	0.300	0.785	1.187	1.386	1.582	1.958
mu_j[76]	TRAVERSE	1.568	0.284	1.018	1.372	1.568	1.760	2.132
mu_j[77]	WABASHA	1.572	0.239	1.109	1.413	1.572	1.731	2.044
mu_j[78]	WADENA	1.143	0.262	0.638	0.966	1.143	1.317	1.669
mu_j[79]	WASECA	0.881	0.285	0.319	0.693	0.887	1.073	1.423
mu_j[80]	WASHINGTON	1.258	0.113	1.039	1.183	1.257	1.332	1.484
mu_j[81]	WATONWAN	1.702	0.309	1.105	1.491	1.702	1.912	2.317
mu_j[82]	WILKIN	1.495	0.358	0.804	1.249	1.489	1.735	2.203
mu_j[83]	WINONA	1.419	0.193	1.044	1.285	1.420	1.551	1.801
mu_j[84]	WRIGHT	1.524	0.194	1.142	1.389	1.523	1.655	1.910

	county	Mean	SD	2.5th quantile	25th quantile	50th quantile	75th quantile	97.5th quantile
mu_j[85]	YELLOW MEDICINE	1.274	0.320	0.642	1.060	1.276	1.490	1.894

This is making the scatterplot of the mean log radon values in each county.

```
#Scatterplot of mean log radon values in each county
ggplot(data = county_summary, aes(x = 1:85, y = Mean)) +
  geom_point(color = "blue") +
  labs(x = "Counties", y = "Mean Log Radon Value",
       title = "Mean Log Radon Values in each County") +
  theme(plot.title = element_text(hjust = 0.5))
```



This is finding the counties with the 3 highest mean log radon values, and the counties with the 3 highest 97.5th quantile for log radon values.

```
#Top 3 by mean value
county_summary %>%
  arrange(desc(Mean)) %>%
  slice(1:3)
```

```
##           county      Mean      SD 2.5th quantile 25th quantile
## mu_j[7]    BLUE EARTH 1.776551 0.1873397      1.404126 1.649534
## mu_j[36]   LAC QUI PARLE 1.741003 0.3385178      1.088749 1.508016
## mu_j[24]    FREEBORN 1.739954 0.2206627      1.314694 1.587599
##           50th quantile 75th quantile 97.5th quantile
## mu_j[7]      1.776316      1.899073      2.152305
```

```
## mu_j[36]      1.741288      1.972459      2.416408
## mu_j[24]      1.741408      1.887289      2.176546
```

*#Top 3 by 97.5th quantile*

```
county_summary %>%
  arrange(desc(`97.5th quantile`)) %>%
  slice(1:3)
```

```
##           county      Mean      SD 2.5th quantile 25th quantile
## mu_j[36] LAC QUI PARLE 1.741003 0.3385178      1.088749      1.508016
## mu_j[81] WATONWAN 1.702427 0.3088504      1.104535      1.491483
## mu_j[51] NICOLLET 1.733616 0.2840474      1.180068      1.542023
##           50th quantile 75th quantile 97.5th quantile
## mu_j[36]      1.741288      1.972459      2.416408
## mu_j[81]      1.702286      1.912137      2.316511
## mu_j[51]      1.728660      1.926305      2.291319
```

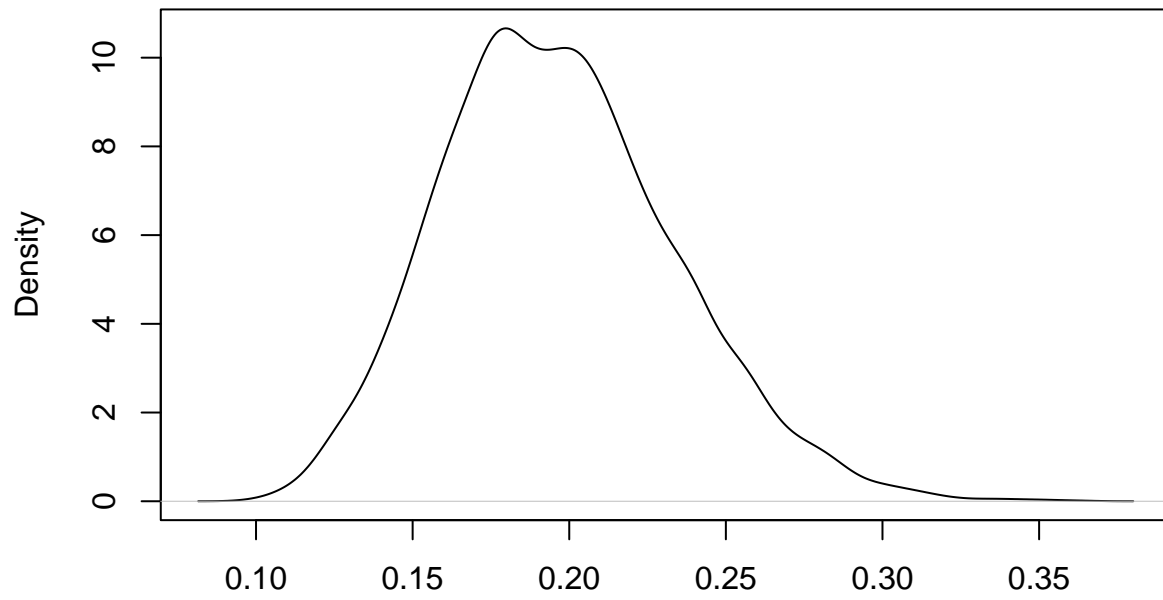
While this wasn't a key detail for analysis, I looked at between versus within county variability out of curiosity. Since much of the posterior probability of  $R$  is located below the value 0.5, this confirms that the variation between each county is smaller than the variation of the radon levels within each county. Since the 97.5th percentile is only 0.277, this means that there is very little evidence for between-county variability.

```
tau_draws <- as.mcmc(posterior, vars = "tau")
sigma_draws <- as.mcmc(posterior, vars = "sigma")
R <- tau_draws ^ 2 / (tau_draws ^ 2 + sigma_draws ^ 2)
summary(R)
```

```
##
## Iterations = 6001:11000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 5000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean      SD      Naive SE Time-series SE
##    0.1966147    0.0372945    0.0005274    0.0009660
##
## 2. Quantiles for each variable:
##
##    2.5%    25%    50%    75%    97.5%
## 0.1306 0.1702 0.1940 0.2201 0.2770
```

```
plot(density(R), labs(main = "Density Plot of R"))
```

**Density Plot of R**



N = 5000 Bandwidth = 0.006095