# MCMC Diagnostics - IFLS data

*Sarah Teichman*

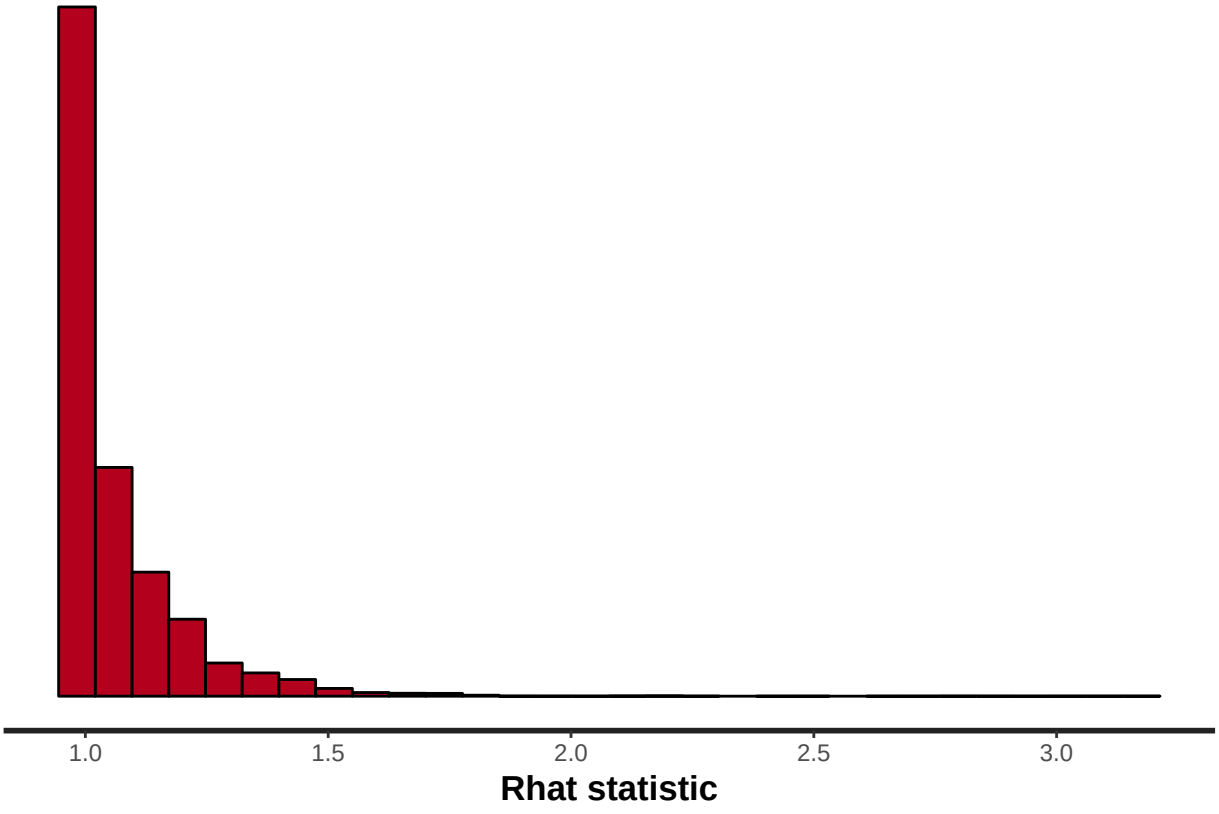
*07/03/2020*

```
K <- 7
Ti <- 3
N <- 1973
```

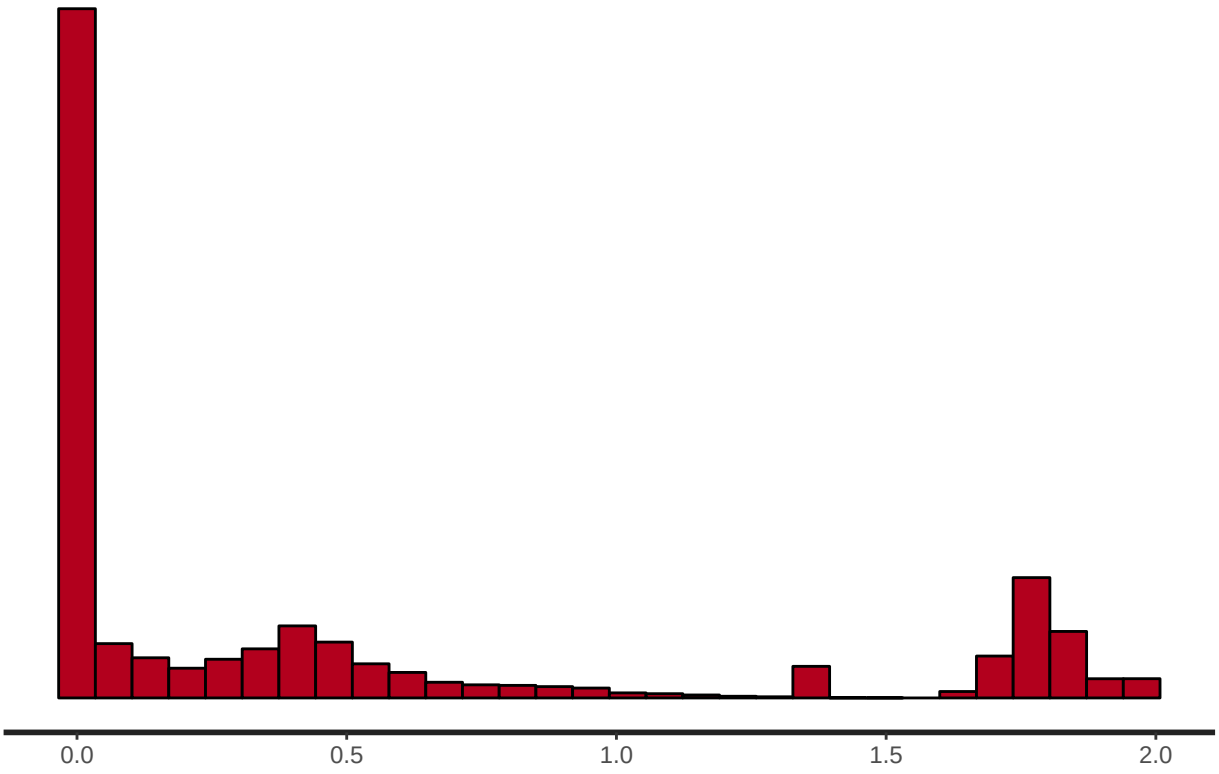
### General MCMC diagnostic plots

Overall model diagnostics from rstan package.

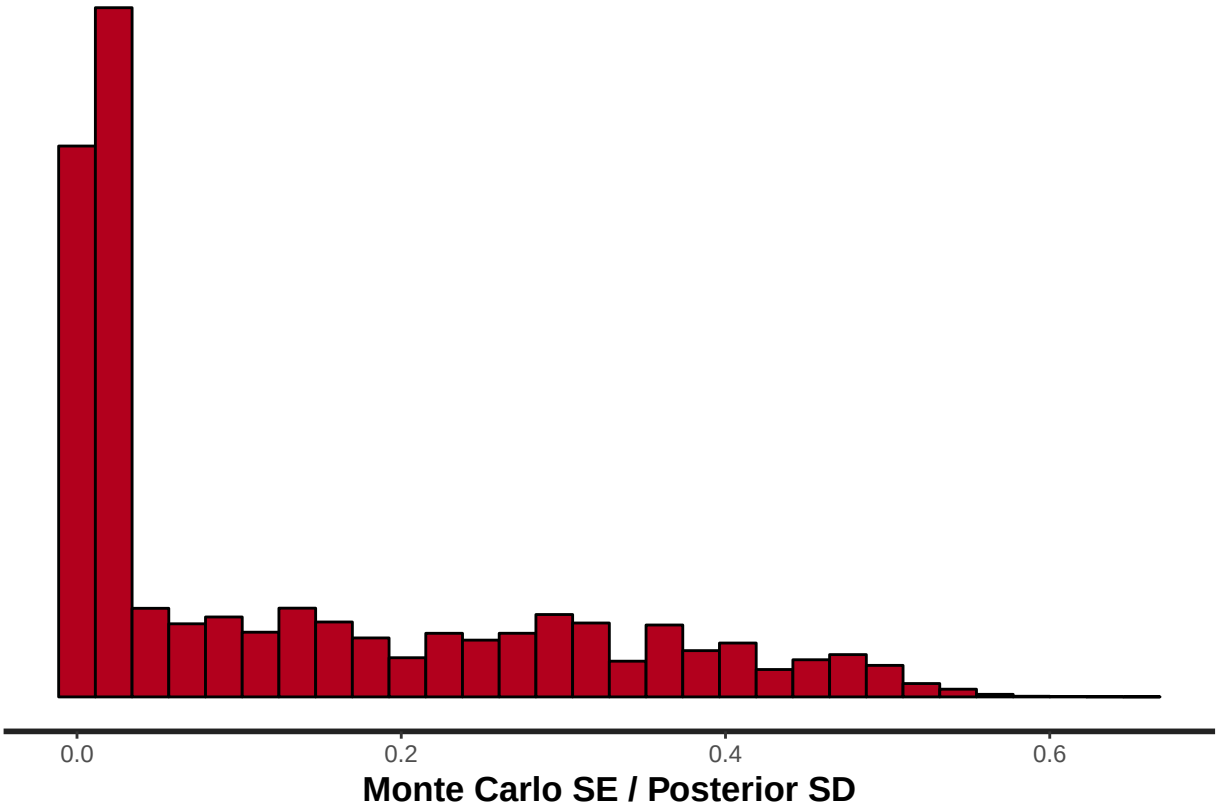
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



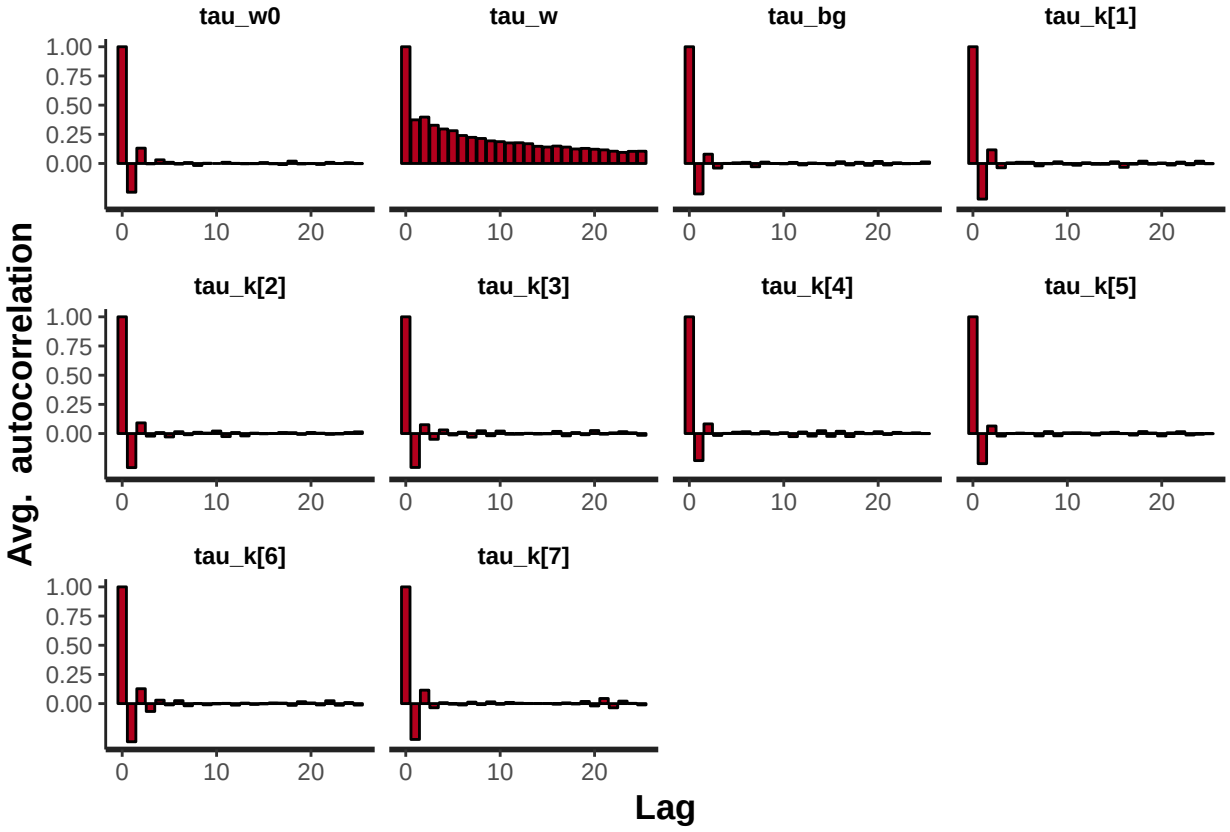
### Individual Parameter Diagnostics

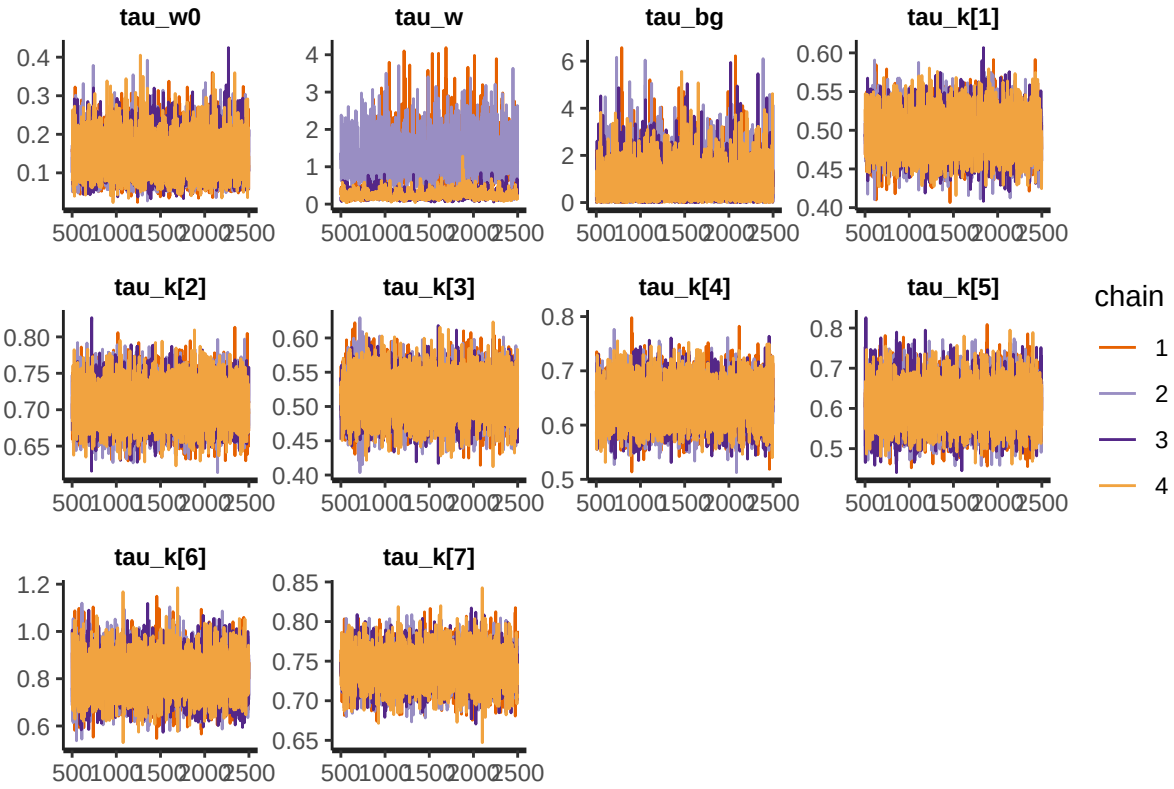
Individual parameter plots. Autocorrelation and trace plots for individual parameters, and histograms of posterior medians for group parameters.

```
get_single_plots <- function(fit, param) {
  print(fit_summ[param,c(1,2,3,5,6,7,9,10)])
  print(stan_ac(fit, pars = param))
  print(rstan::traceplot(fit, pars = param))
}
get_aggreg_plots <- function(fit, param, trim = F, trim_amount) {
  ind <- grep(paste0("^",param), rownames(as.data.frame(summary(fit)$summary)))
  medians <- data.frame(avg = as.data.frame(summary(fit)$summary)$`50%`[ind])
  title <- paste0("Posterior Medians of ",param)
  print(ggplot(medians, aes(x = avg)) + geom_histogram(bins = 30) + ggtitle(title) +
    xlab("Medians") + ylab("Count"))
  print("       ")
  if (trim == T) {
    lim <- quantile(abs(medians$avg), probs = trim_amount)
    meds_trim <- medians %>% filter(abs(medians$avg) < lim)
    print(ggplot(meds_trim, aes(x = avg)) + geom_histogram(bins = 60) +
            ggtitle(paste0(title, " Without Extreme ",100*(1-trim_amount),"%")))
  }
  means <- data.frame(avg = as.data.frame(summary(fit)$summary)$`mean`[ind])
  title <- paste0("Posterior Means of ",param)
  print(ggplot(means, aes(x = avg)) + geom_histogram(bins = 30) + ggtitle(title) +
    xlab("Means") + ylab("Count"))
  print("      ")
  sds <- data.frame(avg = as.data.frame(summary(fit)$summary)$`sd`[ind])
  title <- paste0("Posterior Standard Deviations of ",param)
  print(ggplot(sds, aes(x = avg)) + geom_histogram(bins = 30) + ggtitle(title) +
    xlab("Standard Deviations") + ylab("Count"))
}
plot_fit <- function(fit) {
  get_single_plots(fit, tau_params)
  get_single_plots(fit, beta)
  get_aggreg_plots(fit, "w")
  get_aggreg_plots(fit, "z")
  get_aggreg_plots(fit, "p")
}
plot_fit(fit)
```

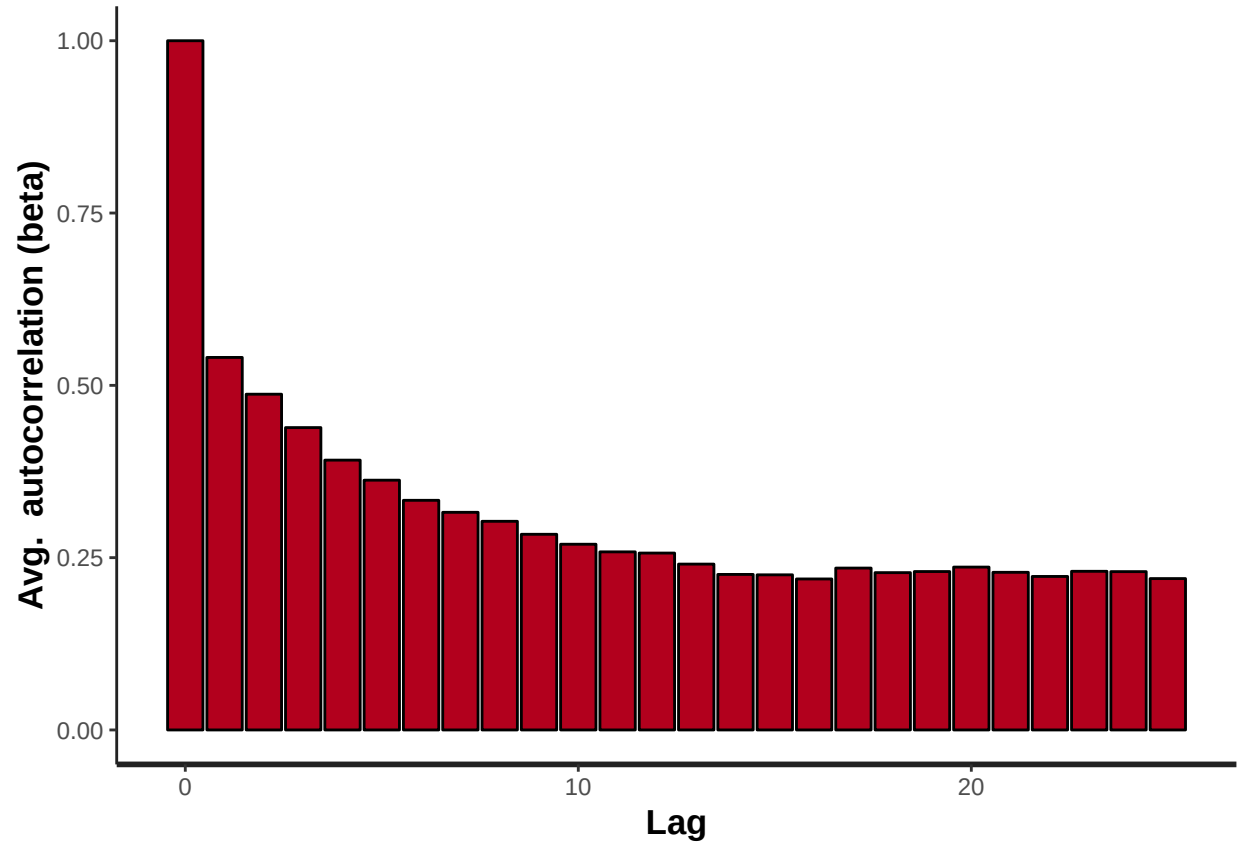
```
##               mean      se_mean         sd       25%       50%       75%
## tau_w0   0.1390434 0.0005353379 0.05166637 0.1015706 0.1319186 0.1701778
## tau_w    0.7269688 0.3518771300 0.63761208 0.2165992 0.4119741 1.1571370
## tau_bg   0.7771439 0.0066620190 0.77330520 0.2280969 0.5432984 1.0666378
## tau_k[1] 0.4955211 0.0002304050 0.02674802 0.4779712 0.4946748 0.5127265
## tau_k[2] 0.7078066 0.0002350892 0.02823286 0.6885168 0.7070915 0.7268156
## tau_k[3] 0.5121901 0.0002426474 0.03048774 0.4909811 0.5115131 0.5324657
## tau_k[4] 0.6409356 0.0003551755 0.03671412 0.6155407 0.6398498 0.6656795
## tau_k[5] 0.6078210 0.0004535134 0.05340558 0.5708170 0.6062448 0.6425526
## tau_k[6] 0.8116088 0.0006653318 0.08360877 0.7544889 0.8087230 0.8648769
## tau_k[7] 0.7421658 0.0001823733 0.02195914 0.7273321 0.7419537 0.7567059
##                 n_eff      Rhat
```

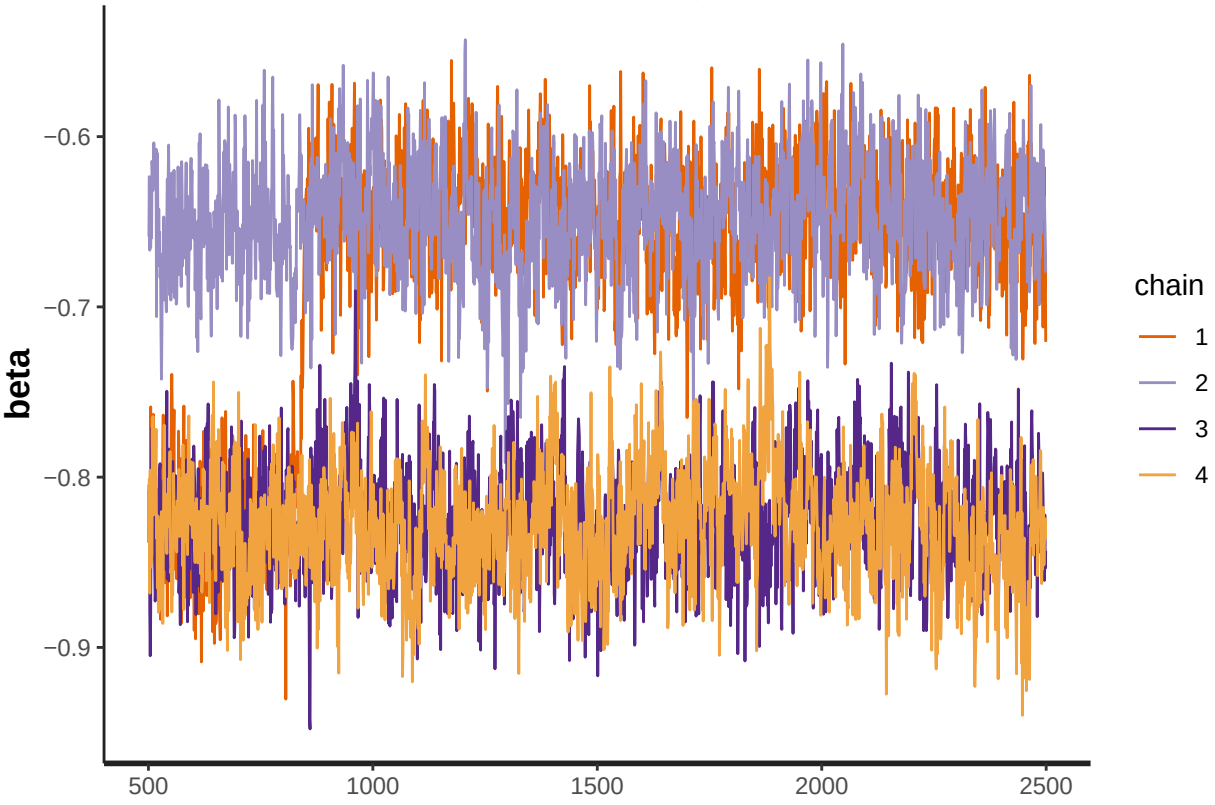
```
## tau_w0    9314.505871 1.0003891
## tau_w        3.283454 1.6367063
## tau_bg   13473.800903 1.0007620
## tau_k[1] 13477.201458 0.9998075
## tau_k[2] 14422.623092 0.9998169
## tau_k[3] 15786.984613 0.9998202
## tau_k[4] 10685.139265 0.9999653
## tau_k[5] 13867.332658 0.9996276
## tau_k[6] 15791.633857 0.9997157
## tau_k[7] 14497.989906 0.9998169
```

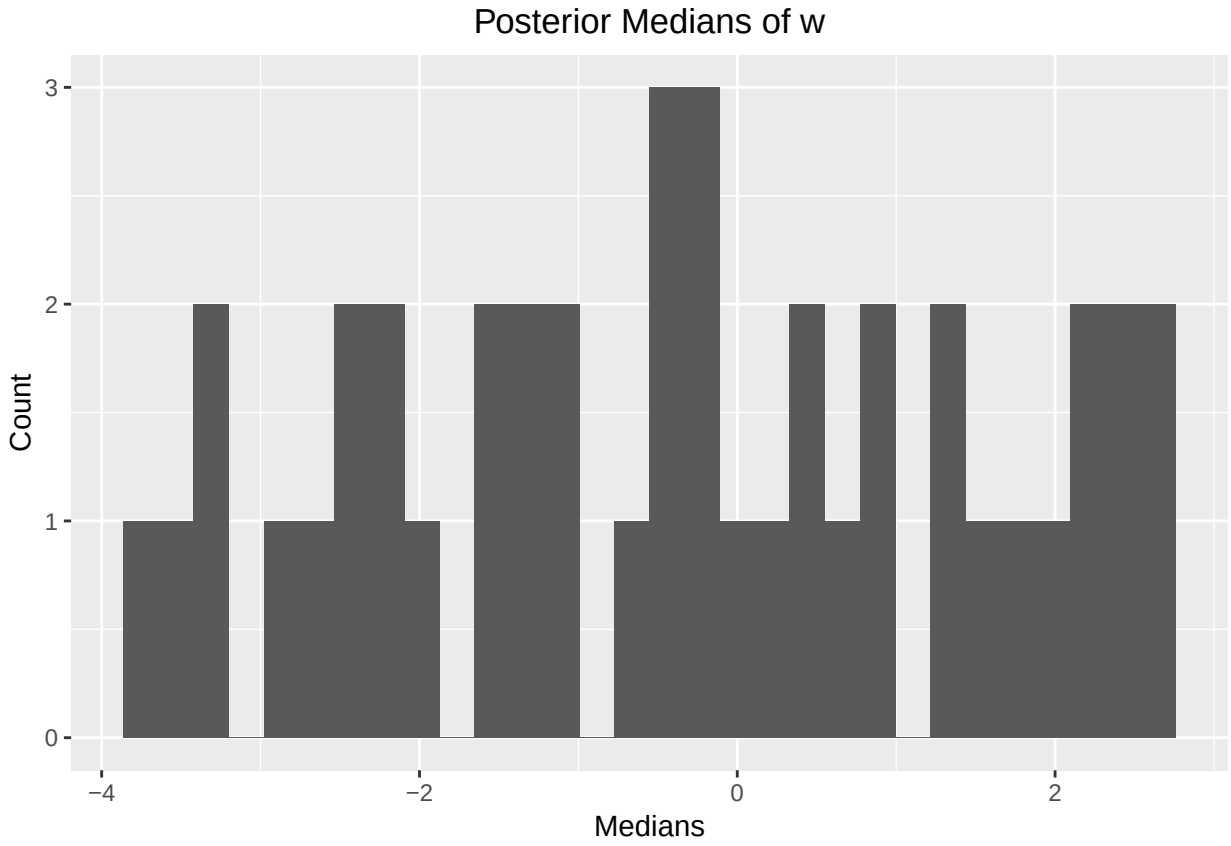




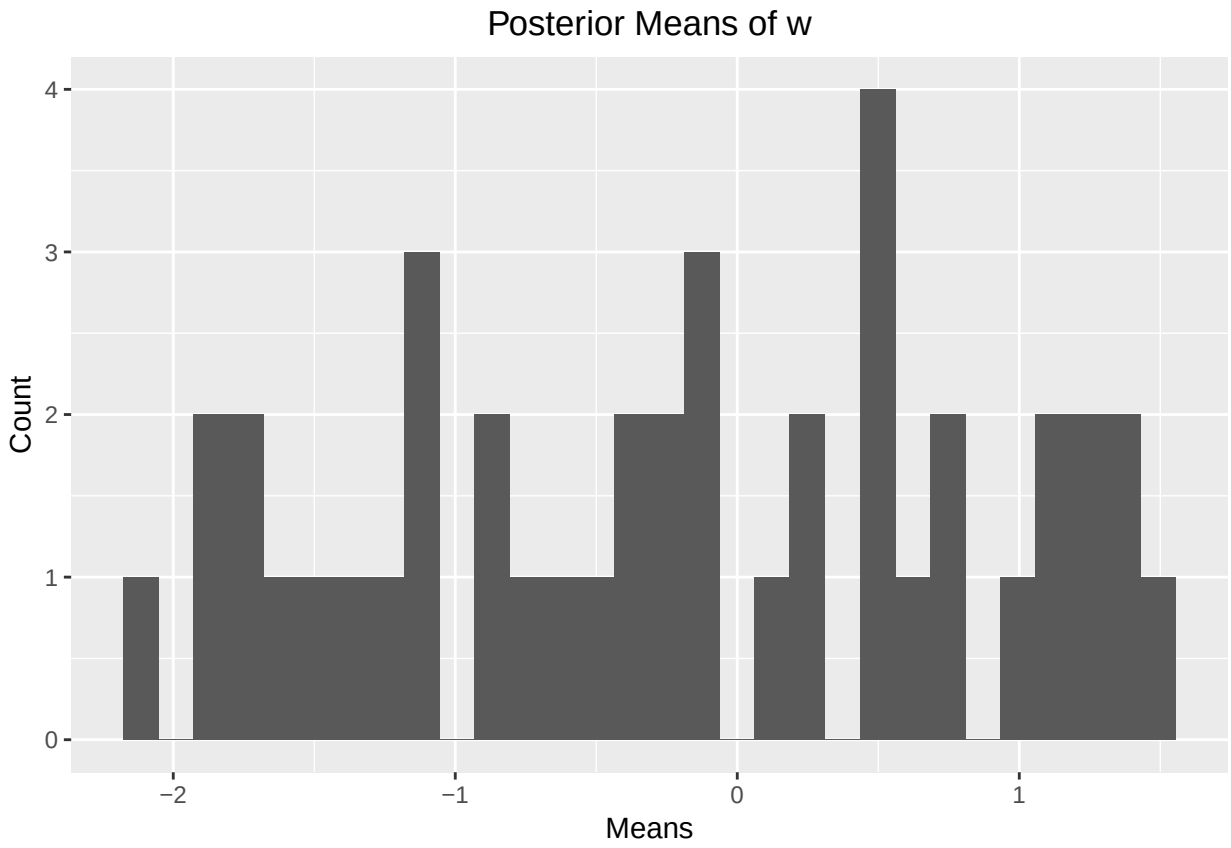

```
##             mean    se_mean         sd        25%       50%        75%
## beta -0.7443171 0.05929281 0.09504036 -0.8301068 -0.779555 -0.6499302
##          n_eff     Rhat
## beta 2.569284 2.251311
```


Avg. autocorrelation (beta)
1.00
0.75
0.50
0.25
0.00
0
10
20
Lag

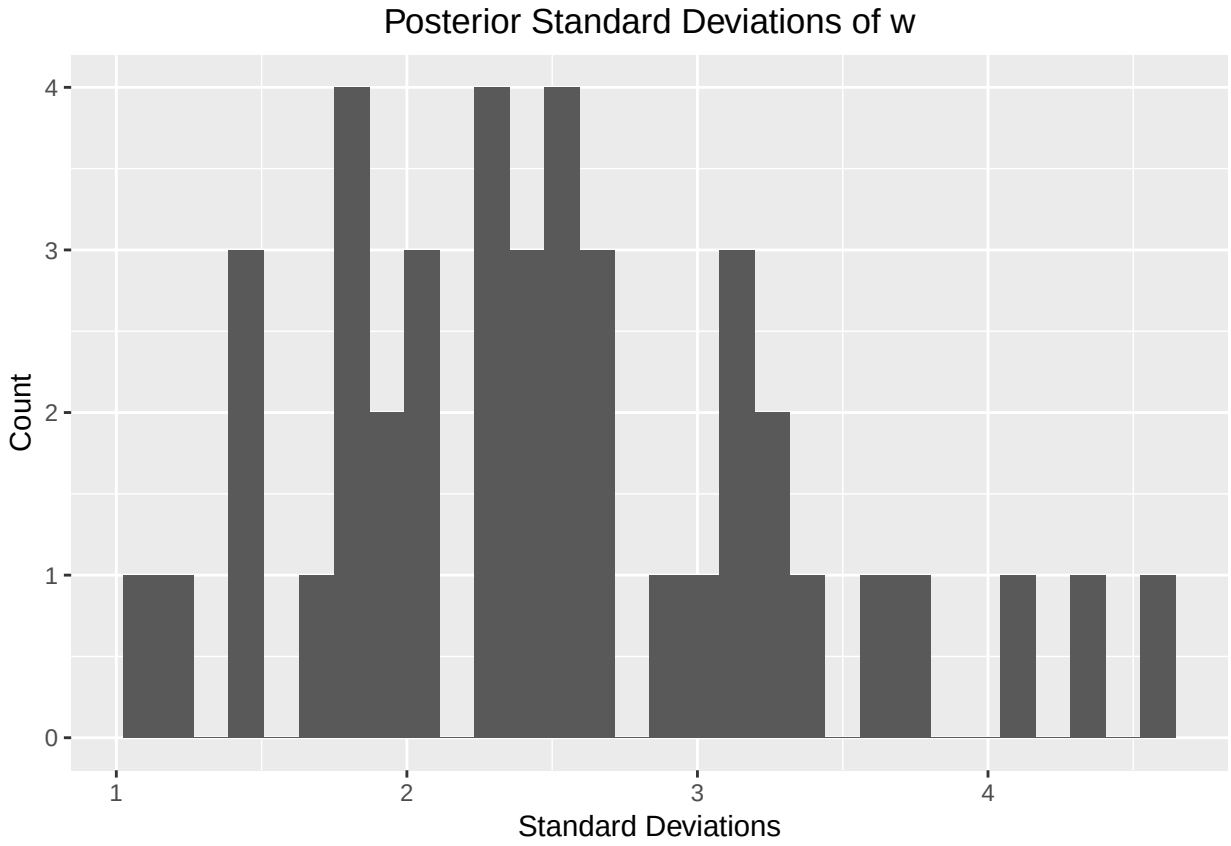

beta
−0.6
−0.7
−0.8
−0.9
500
1000
1500
2000
2500
chain
1
2
3
4

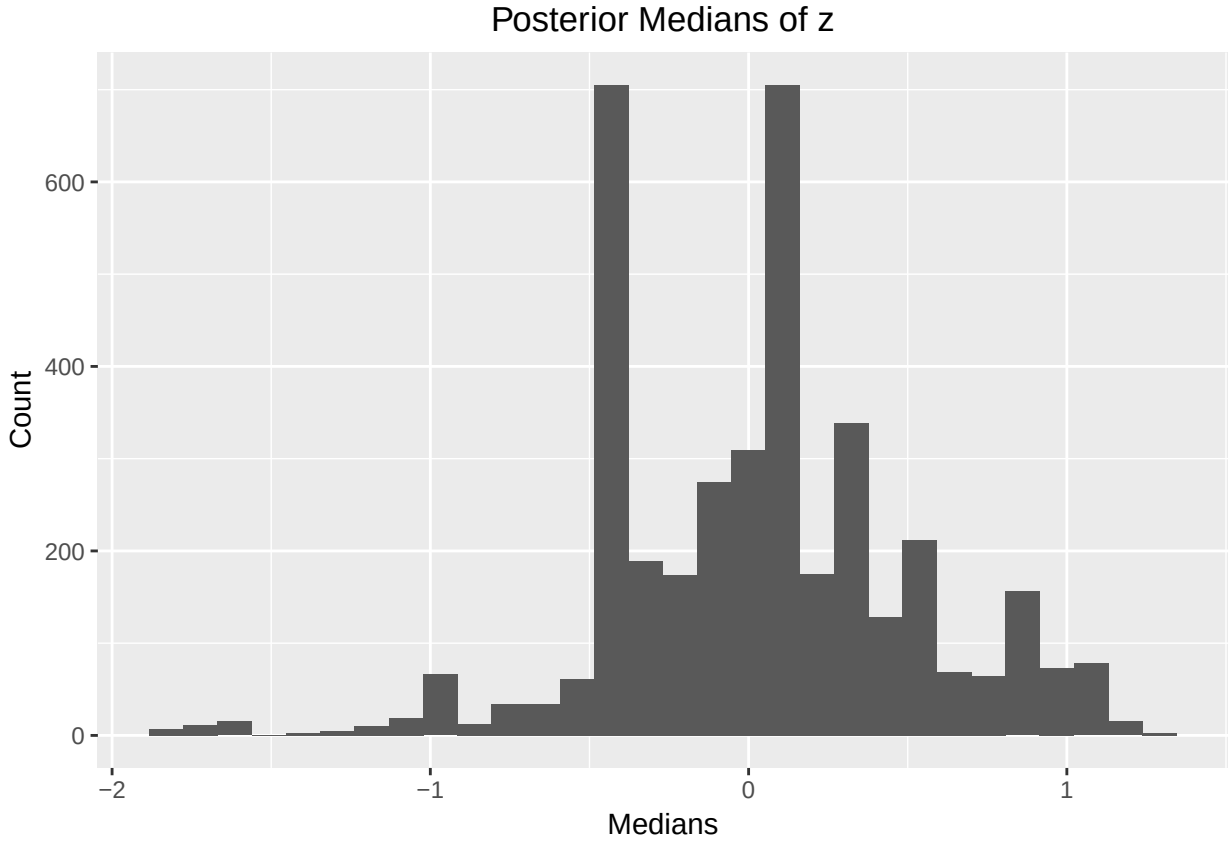



```
## [1] "       "
```

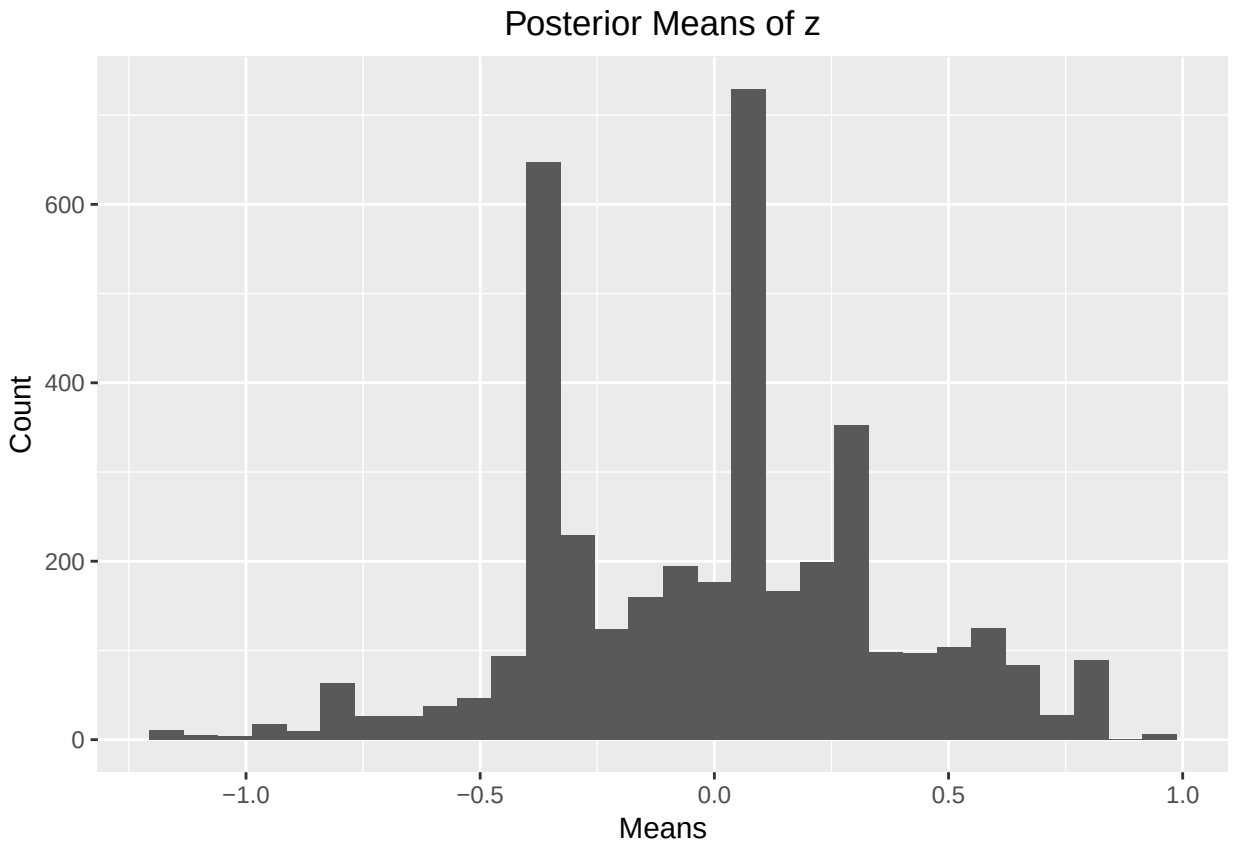



```
## [1] "      "
```

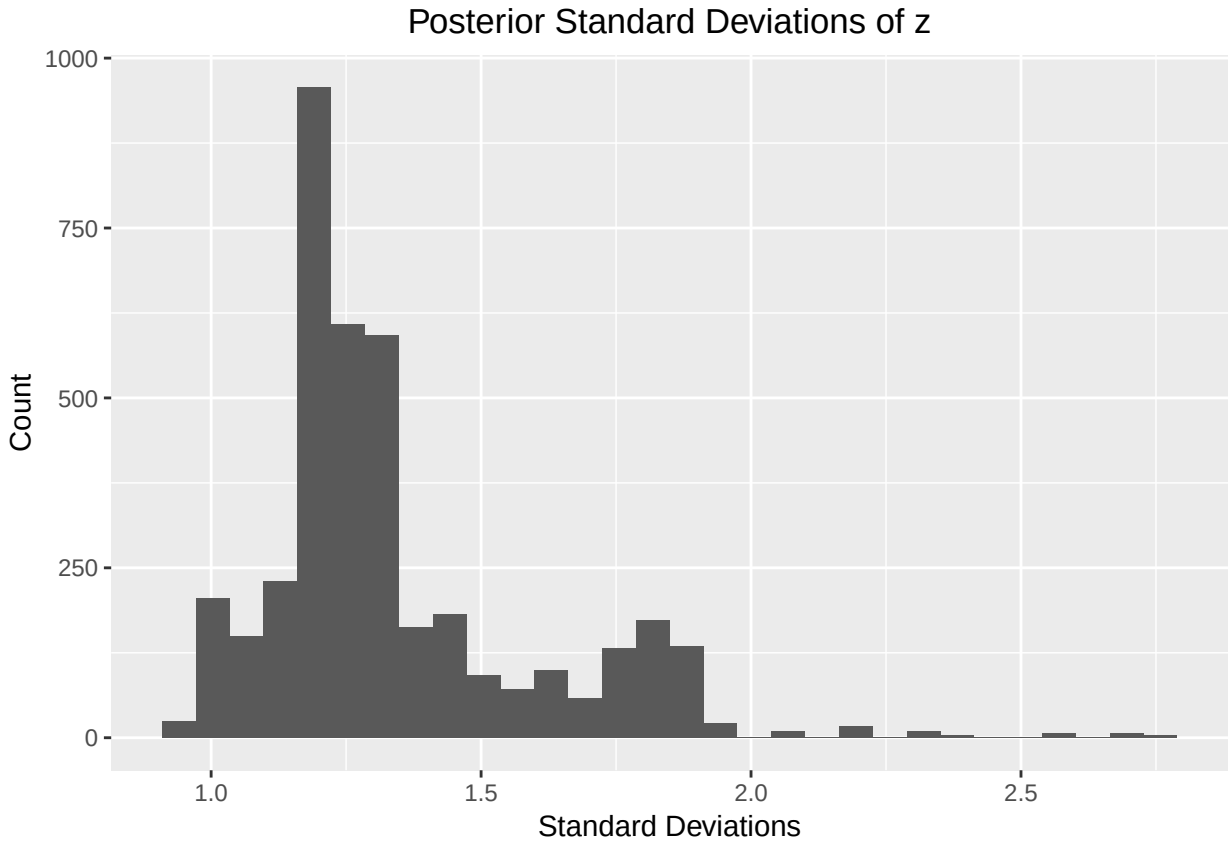

Posterior Standard Deviations of w
Count
0
1
2
3
4
1
2
3
4
Standard Deviations

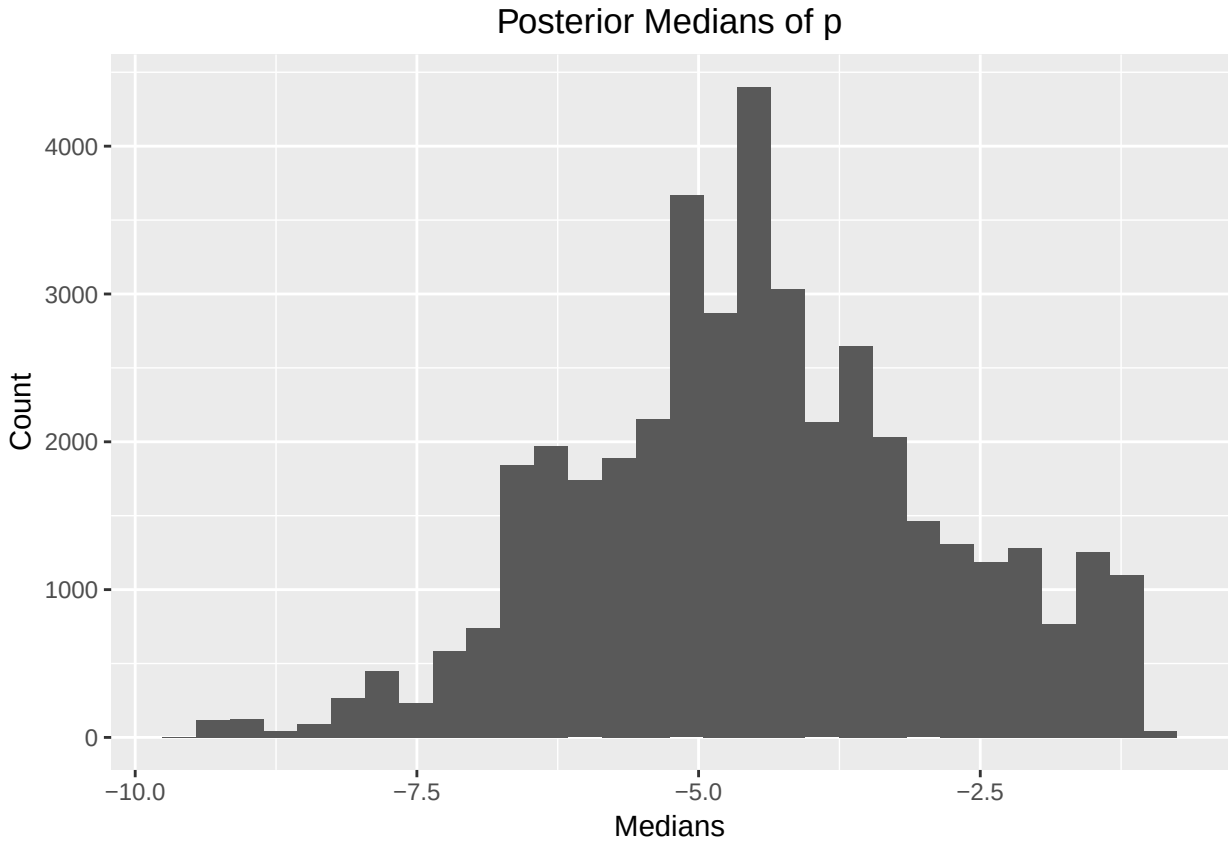

Posterior Medians of z
Count
0
200
400
600
−2
−1
0
1
Medians

```
## [1] "       "
```

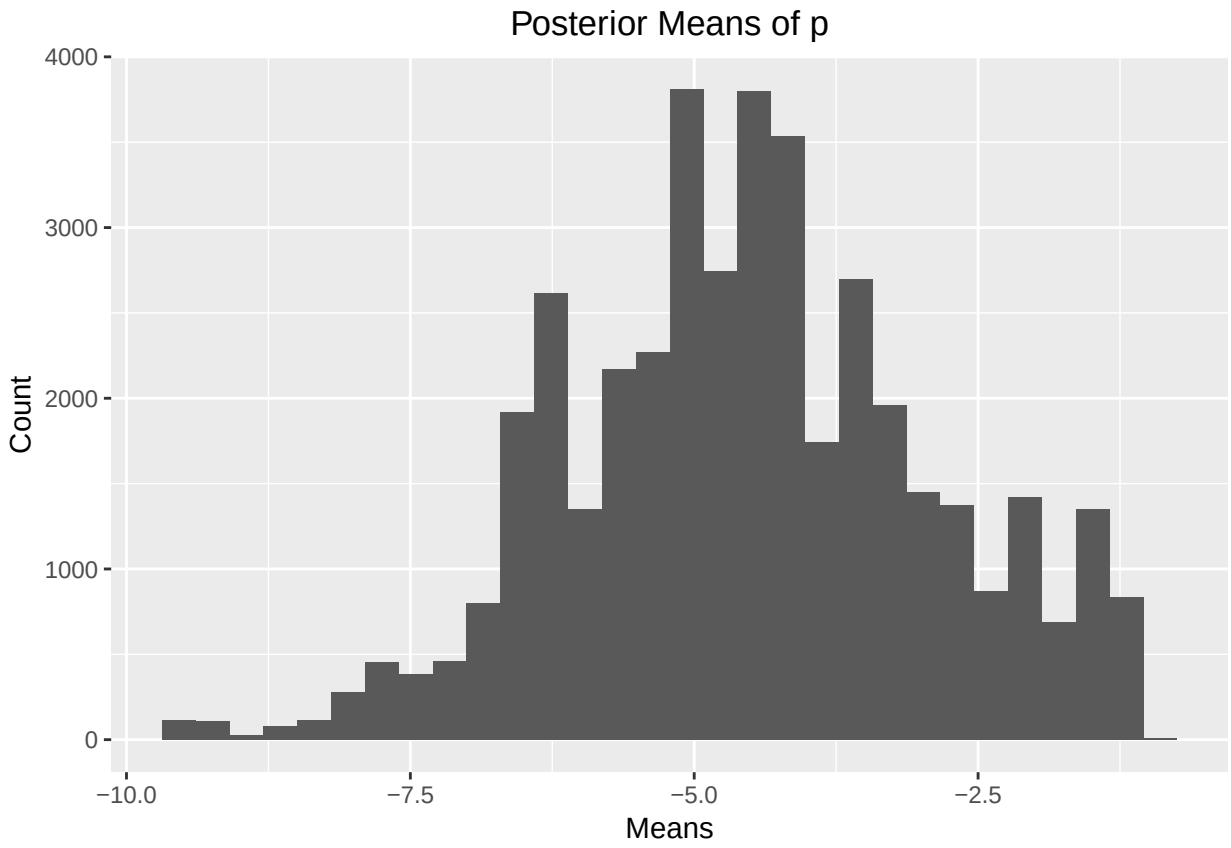



```
## [1] "       "
```

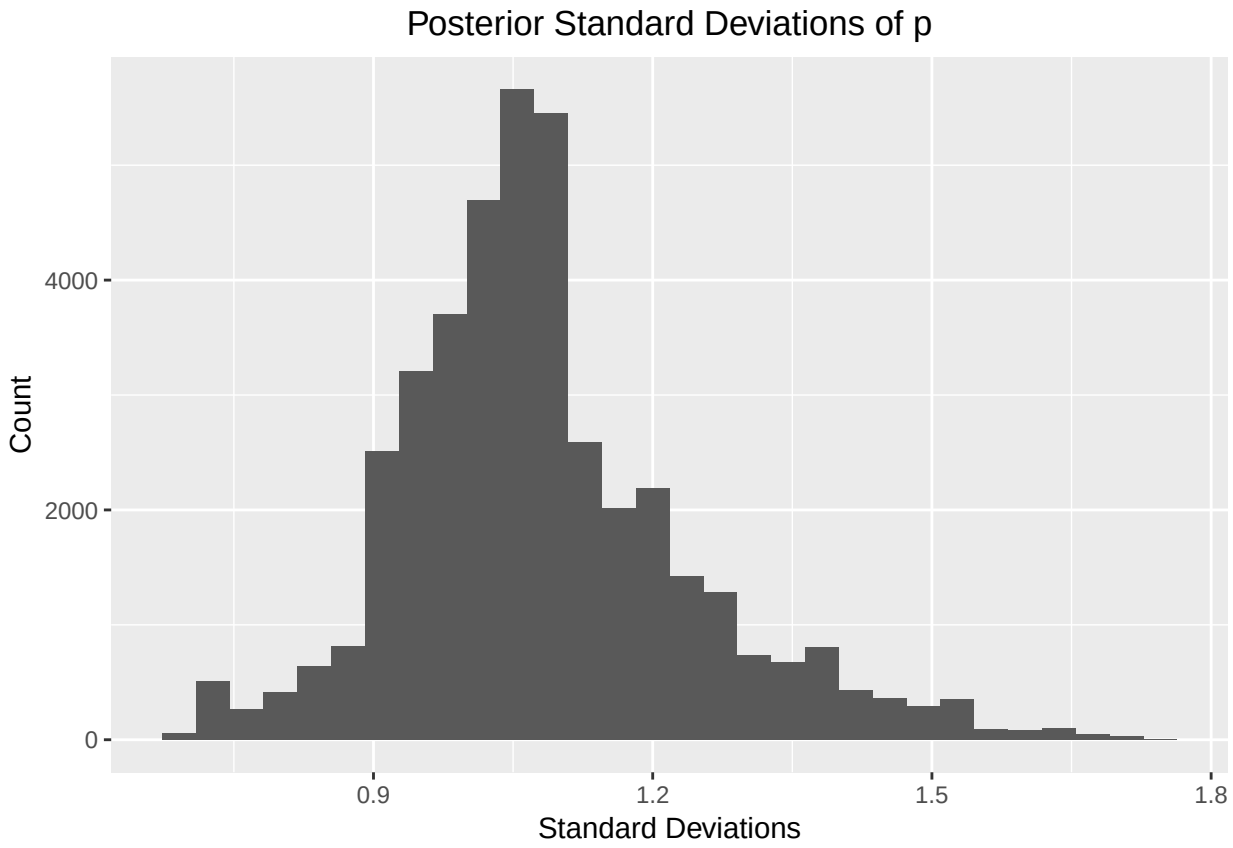

Posterior Standard Deviations of z
Count
1000
750
500
250
0
1.0
1.5
2.0
2.5
Standard Deviations


Posterior Medians of p
Count
4000
3000
2000
1000
0
−10.0
−7.5
−5.0
−2.5
Medians

```
## [1] "       "
```


Posterior Means of p
Count
4000
3000
2000
1000
0
−10.0
−7.5
−5.0
−2.5
Means

```
## [1] "       "
```


Posterior Standard Deviations of p
Count
4000
2000
0
0.9
1.2
1.5
1.8
Standard Deviations