Homework #7

Dustin Leatherman

November 1, 2020

Let Y_i be the precipitation for observation i and X_{ij} equal one if OTU j is present in sample i. First, extract the 50 OTU with the largest absolute correlation between X_{ij} and Y_i. Then fit a Bayesian linear regression model with precipitation as the response and with these 50 covariates (and an intercept term) using three priors:

1. Uninformative Normal Priors: $\beta_j \sim N(0, 100^2)$
2. Hierarchical Normal Priors: $\beta_j|\tau \sim N(0, \tau^2),\ \tau^2 \sim InvGamma(0.01, 0.01)$
3. Bayesian LASSO: $\beta_j|\tau^2 \sim DE(0, \tau^2),\ \tau^2 \sim InvGamma(0.01, 0.01)$

Compare convergence and the posterior distribution of the regression coefficients under these three priors. In particular, are the same OTU's significant in all three fits?

```r
load("~/StatisticsMasters/bayesian/homes.rdata")

# Join OTU dataset with Homes metadata
sorted_data <-
  # Indicator if OTU is present
  as.data.frame(ifelse(OTU > 0, 1, 0)) %>%
  rownames_to_column(var = "ID") %>%
  # data cleansing
  mutate(
    ID = as.integer(str_remove(ID, ".I"))
  ) %>%
  # join with metadata and sort by ids
  inner_join(homes, by = "ID") %>%
  arrange(ID)


X <- sorted_data %>% select(starts_with("OTU")) %>% as.matrix()
Y <- sorted_data %>% select(MeanAnnualPrecipitation) %>% as.matrix()

# calculate top 50 OTUs by absolute correlation with rainfall
top_otu <-
  as.data.frame(cor(X, Y)) %>%
    rownames_to_column(var = "OTU") %>%
    rename(corr = MeanAnnualPrecipitation) %>%
    arrange(desc(abs(corr))) %>%
    head(n = 50) %>%
    select("OTU")
```

```
## Warning in cor(X, Y): the standard deviation is zero
```

```r
# Put the data in JAGS format
X <-
```

```r
  sorted_data %>%
    #mutate(intercept = 0) %>%
    #select(intercept, top_otu$OTU) %>%
    select(top_otu$OTU) %>%
    as.matrix()


Y <- sorted_data %>% select(MeanAnnualPrecipitation)
Y <- Y$MeanAnnualPrecipitation

# utility function to union the output of JAGS
unionJagsOutput <- function(jags_data, names) {
  colnames(jags_data[[1]]) <- names
  colnames(jags_data[[2]]) <- names
  return (
    jags_data[[1]] %>%
      as_tibble() %>%
      mutate(chain = "1", row_num = row_number()) %>%
      pivot_longer(starts_with("OTU"), names_to = "var", values_to = "value") %>%
      union_all(
        jags_data[[2]] %>%
        as_tibble() %>%
        mutate(chain = "2", row_num = row_number()) %>%
        pivot_longer(starts_with("OTU"), names_to = "var", values_to = "value")
      )
  )
}

convergenceDiags <- function(jags_data, names) {
  p1 <-
    unionJagsOutput(jags_data, names) %>%
      ggplot(aes(x = row_num, y = value, color = chain)) +
        geom_line() +
        facet_wrap(~var, ncol = 5) +
        labs(x = "Iteration", y = "Posterior Value") +
        ggtitle("Trace of OTU Covariates")

  # Geweke Statistic
  p3 <- geweke.diag(jags_data)

  list(p1 = p1, p3 = p3)
}
```

Fitting the Models

Let's fit our different Models. First, some setup though.

```r
n <- length(Y)
p <- ncol(X)

data <- list(Y=Y,X=X,n=n,p=p)
names <- colnames(X)
```

```r
params <- c("beta")

# Settings (automatically calculates the number of iterations needed based on inputs)

nBurn <- 10000
nChains <- 2
nSave <- 4000
nThin <- 10
nIter <- ceiling((nSave*nThin)/nChains)
```

Uninformative Normal Prior

$\beta_j \sim N(0, 100^2)$

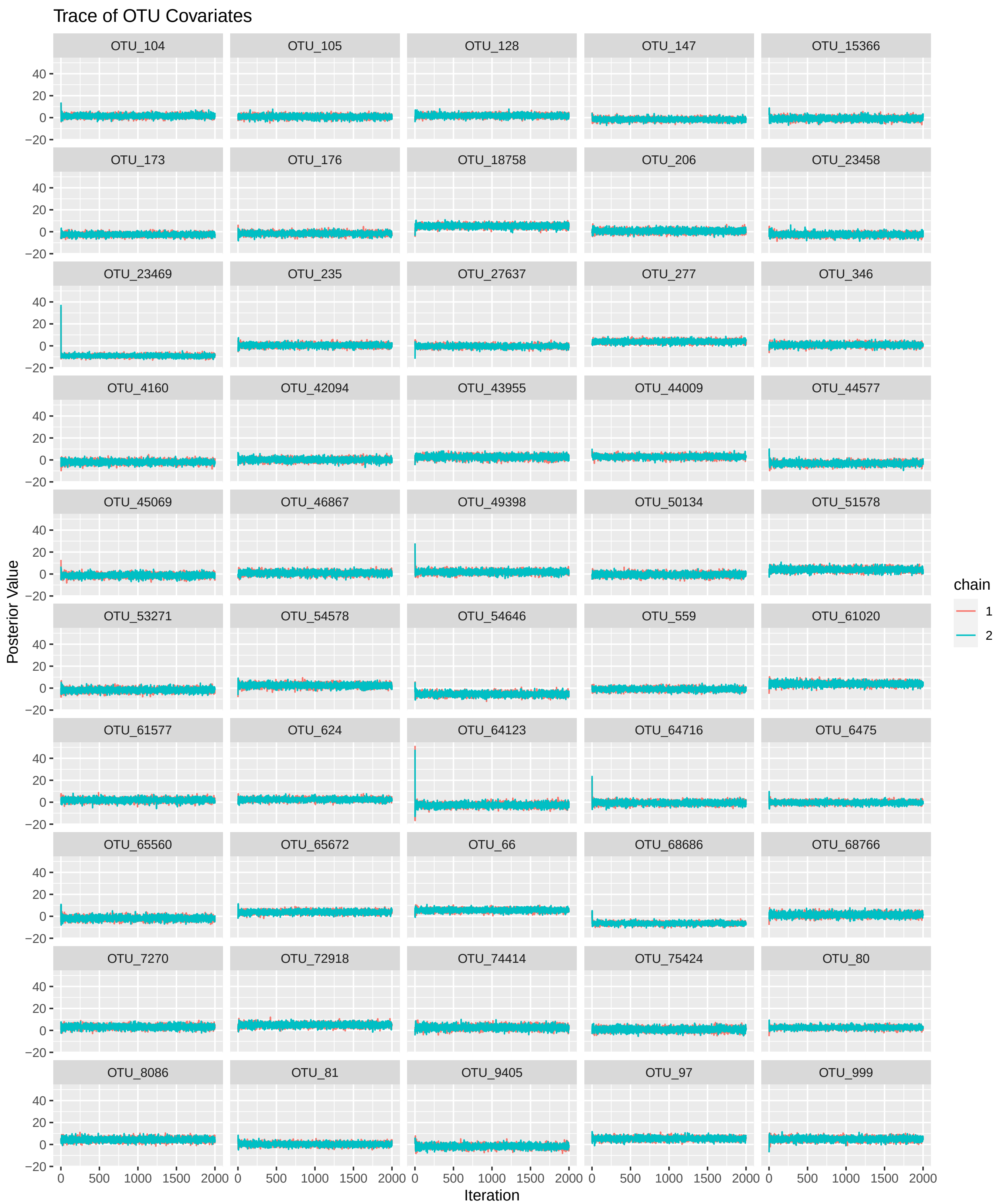
```r
model_string1 <- textConnection("model{
  # Likelihood
  for(i in 1:n){
    Y[i] ~ dnorm(alpha + inprod(X[i,], beta[]), taue)
  }
  # Priors
  for(j in 1:p){
    beta[j] ~ dnorm(0,0.01)
  }
  alpha ~ dnorm(0,0.01)
  taue ~ dgamma(0.1,0.1)
}")

model <- jags.model(model_string1,data=data,n.chains=nChains,quiet=TRUE)
update(model,burn=nBurn,progress.bar="none")
samples1 <- coda.samples(model,variable.names=params,thin=nThin,n.iter=nIter, progress.bar="none")
```

Convergence

```r
convergenceDiags(samples1, names)
```

```
## $p1
```




```
##
## $p3
## $p3[[1]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
```

```
##
##    beta[1]   beta[2]   beta[3]   beta[4]   beta[5]   beta[6]   beta[7]   beta[8]
## -1.896553 -1.213712 -0.790650 -0.968541 -0.087435 -0.005671  0.405580  0.717480
##    beta[9]  beta[10]  beta[11]  beta[12]  beta[13]  beta[14]  beta[15]  beta[16]
##  0.765891 -1.151549 -0.317467  2.994067 -0.281611  0.692229  0.074587 -0.797158
##   beta[17]  beta[18]  beta[19]  beta[20]  beta[21]  beta[22]  beta[23]  beta[24]
##  0.369532 -0.047323  1.766071 -0.175980  0.611209  1.157035 -0.006498 -0.669323
##   beta[25]  beta[26]  beta[27]  beta[28]  beta[29]  beta[30]  beta[31]  beta[32]
## -0.885826 -1.334443  1.234013  0.291985 -0.535353 -1.421727  2.224038  0.053253
##   beta[33]  beta[34]  beta[35]  beta[36]  beta[37]  beta[38]  beta[39]  beta[40]
## -0.481938 -0.896334  0.723477  0.532980  1.110954 -1.648406  0.423094  1.424063
##   beta[41]  beta[42]  beta[43]  beta[44]  beta[45]  beta[46]  beta[47]  beta[48]
##  0.752894  0.299436 -0.746968 -0.356495  0.009525  0.892394 -0.851871  0.821191
##   beta[49]  beta[50]
##  0.751628  0.467520
##
##
## $p3[[2]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##    beta[1]   beta[2]   beta[3]   beta[4]   beta[5]   beta[6]   beta[7]   beta[8]
##  0.175118 -0.296487 -1.809555  0.723186  0.161235 -1.694785  1.563892  0.770586
##    beta[9]  beta[10]  beta[11]  beta[12]  beta[13]  beta[14]  beta[15]  beta[16]
##  0.520916 -0.608995  0.096903  0.181433 -0.380354  0.386511  0.599666 -0.647586
##   beta[17]  beta[18]  beta[19]  beta[20]  beta[21]  beta[22]  beta[23]  beta[24]
## -0.003173 -0.424802  0.343090  0.846076  0.224883 -2.324151 -0.168237  0.409611
##   beta[25]  beta[26]  beta[27]  beta[28]  beta[29]  beta[30]  beta[31]  beta[32]
##  0.334971 -0.541726  1.030561 -1.824041 -0.413188  0.935040  0.217605 -0.197126
##   beta[33]  beta[34]  beta[35]  beta[36]  beta[37]  beta[38]  beta[39]  beta[40]
##  0.861667 -1.041956  0.964948  1.431119 -1.524896 -0.717013  0.766135  1.313138
##   beta[41]  beta[42]  beta[43]  beta[44]  beta[45]  beta[46]  beta[47]  beta[48]
##  0.568614  0.761285  0.591903 -0.164032  0.354550 -1.456377  0.632699  0.296381
##   beta[49]  beta[50]
##  0.765585  0.977798
```

```r
# gelmen-rubin statistics
gelman.diag(samples1)$psrf %>%
      kable(
        caption = "Gelman-Rubin Statistics for convergence"
      ) %>%
      kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

According to the Gelman-Rubin diagnostics, all the coefficients converge. The Geweke statistics show convergence and the Trace plots support this conclusion.

Hierarchical Normal Prior

$\beta_j|\tau \sim N(0, \tau^2),\ \tau^2 \sim InvGamma(0.01, 0.01)$

```r
model_string2 <- textConnection("model{
  # Likelihood
  for(i in 1:n){
```

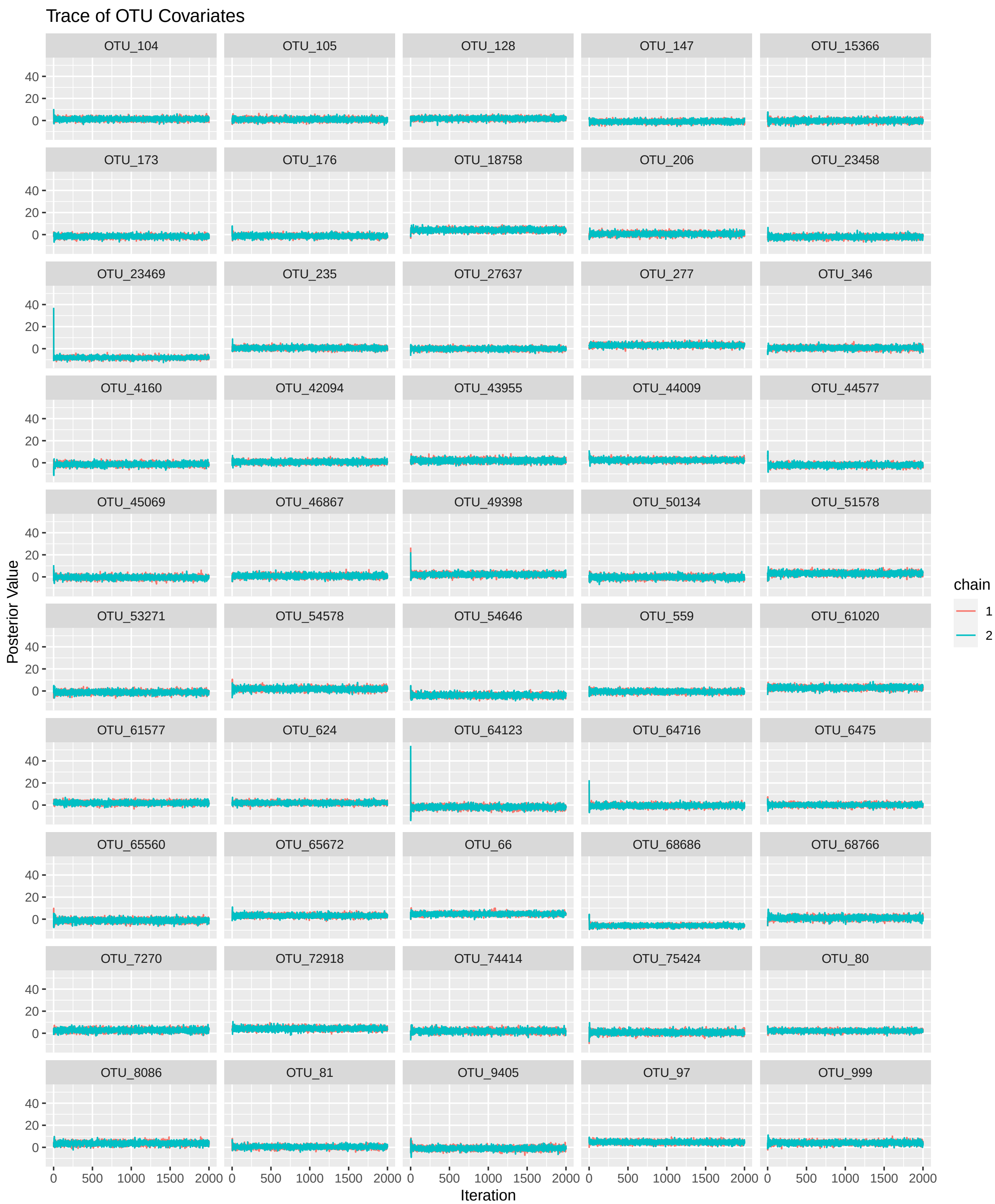
```
    Y[i] ~ dnorm(alpha + inprod(X[i,], beta[]), taue)
  }
  # Priors
  for(j in 1:p){
    beta[j] ~ dnorm(0, taub)
  }
  alpha ~ dnorm(0,0.01)
  taue ~ dgamma(0.01,0.01)
  taub ~ dgamma(0.01,0.01)
}")

model <- jags.model(model_string2,data=data,n.chains=nChains,quiet=TRUE)
update(model,burn=nBurn,progress.bar="none")
samples2 <- coda.samples(model,variable.names=params,thin=nThin,n.iter=nIter, progress.bar="none")
```

Convergence

```
convergenceDiags(samples2, names)
```

```
## $p1
```




```
##
## $p3
## $p3[[1]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
```

```
##
##  beta[1]  beta[2]  beta[3]  beta[4]  beta[5]  beta[6]  beta[7]  beta[8]
## -0.66965 -0.94309  0.90865 -0.52303  1.07685  0.02619 -0.21639  0.73369
##  beta[9] beta[10] beta[11] beta[12] beta[13] beta[14] beta[15] beta[16]
## -1.53264 -0.27072 -1.94402 -0.02818  0.23027 -1.13353  1.08638 -0.67612
## beta[17] beta[18] beta[19] beta[20] beta[21] beta[22] beta[23] beta[24]
##  1.12102 -1.23320  1.28846  0.17880  0.81877 -0.48056  0.98002 -0.79962
## beta[25] beta[26] beta[27] beta[28] beta[29] beta[30] beta[31] beta[32]
##  0.02872 -0.67483  0.28743  1.23849 -0.01756 -0.40658  1.68827 -1.22907
## beta[33] beta[34] beta[35] beta[36] beta[37] beta[38] beta[39] beta[40]
## -0.83942 -1.18204 -0.07101 -0.38051  0.77693  0.98282 -0.79220  0.57478
## beta[41] beta[42] beta[43] beta[44] beta[45] beta[46] beta[47] beta[48]
##  1.36865  0.59148 -0.20002  1.50186 -0.77148 -0.19332  0.52790  0.40067
## beta[49] beta[50]
##  0.57557  0.82229
##
##
## $p3[[2]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##  beta[1]  beta[2]  beta[3]  beta[4]  beta[5]  beta[6]  beta[7]  beta[8]
## -1.24956 -1.35209 -0.58743 -0.20020  1.65518  0.46866  0.21372  0.92450
##  beta[9] beta[10] beta[11] beta[12] beta[13] beta[14] beta[15] beta[16]
##  0.43701 -0.98510  0.66596  0.53886  0.36724 -0.56442  0.31414 -0.76612
## beta[17] beta[18] beta[19] beta[20] beta[21] beta[22] beta[23] beta[24]
##  1.51630  0.20599  0.11890 -1.71055  1.47626  1.67426 -0.35537 -0.02050
## beta[25] beta[26] beta[27] beta[28] beta[29] beta[30] beta[31] beta[32]
## -0.57090 -0.46876 -0.99425  2.24709 -1.56601  1.50490  0.63404 -0.92066
## beta[33] beta[34] beta[35] beta[36] beta[37] beta[38] beta[39] beta[40]
## -0.08511  0.80985  0.14482 -0.51318  0.83272 -1.43496 -0.70725 -0.54324
## beta[41] beta[42] beta[43] beta[44] beta[45] beta[46] beta[47] beta[48]
##  1.45102  0.32094 -0.99881 -1.21424  0.72975  0.90255 -1.21450 -0.33531
## beta[49] beta[50]
##  1.13153  1.52123
```

```r
gelman.diag(samples2)$psrf %>%
      kable(
        caption = "Gelman-Rubin Statistics for convergence"
      ) %>%
      kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

According to the Gelman-Rubin diagnostics, all the coefficients converge. The Geweke statistics show convergence and the Trace plots support this conclusion.

Bayesian LASSO

$\beta_j|\tau^2 \sim DE(0,\tau^2),\ \tau^2 \sim InvGamma(0.01, 0.01)$

```r
model_string3 <- textConnection("model{
  # Likelihood
  for(i in 1:n){
    Y[i] ~ dnorm(alpha + inprod(X[i,], beta[]), taue)
```

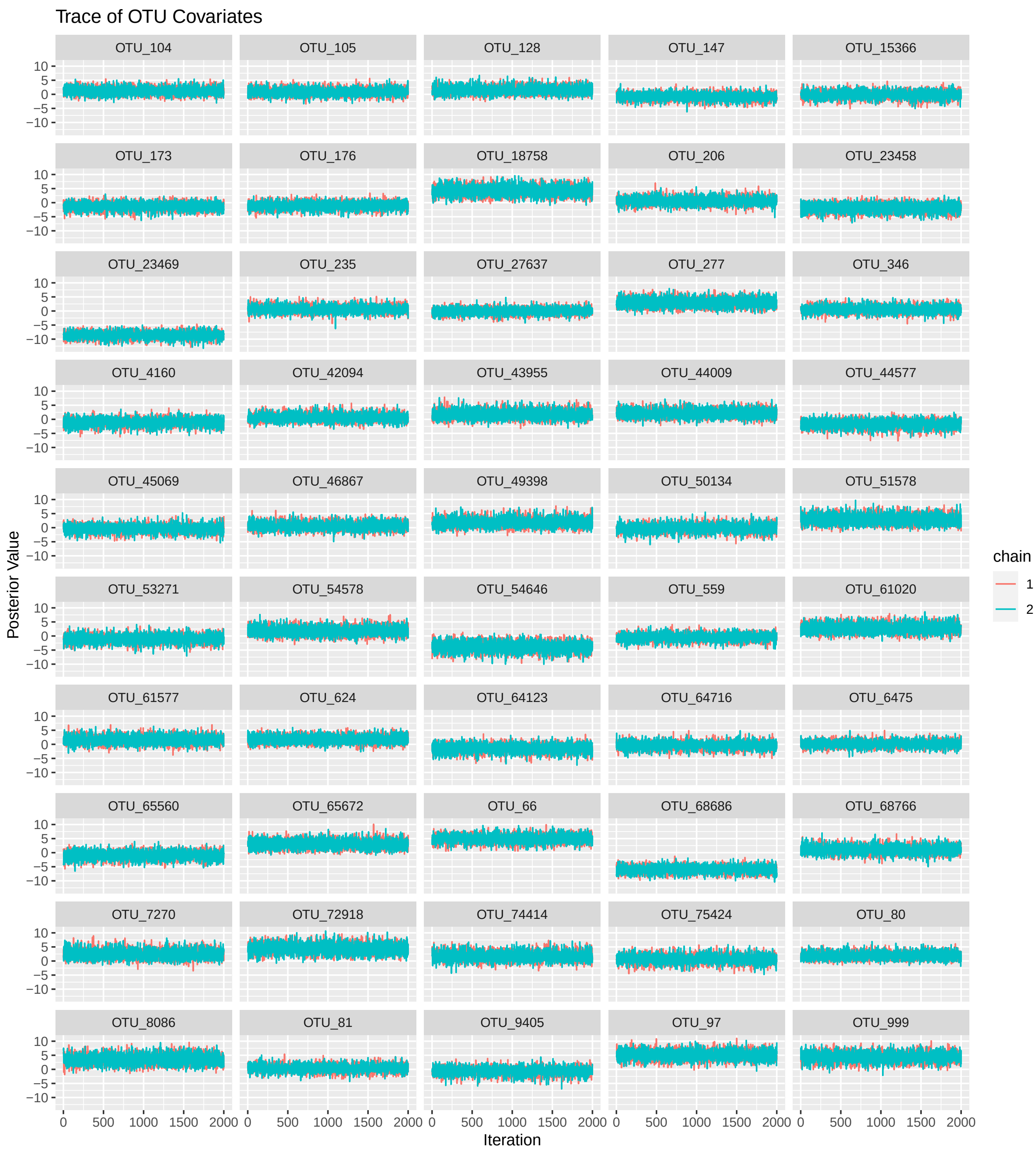
```
  }
  # Priors
  for(j in 1:p){
    beta[j] ~ ddexp(0, taub)
  }
  alpha ~ dnorm(0,0.01)
  taue ~ dgamma(0.01,0.01)
  taub ~ dgamma(0.01,0.01)
}")

model <- jags.model(model_string3,data=data,n.chains=nChains,quiet=TRUE)
update(model,burn=nBurn,progress.bar="none")
samples3 <- coda.samples(model,variable.names=params,thin=nThin,n.iter=nIter, progress.bar="none")
```

Convergence

```
convergenceDiags(samples3, names)
```

```
## $p1
```




```
##
## $p3
## $p3[[1]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##  beta[1]  beta[2]  beta[3]  beta[4]  beta[5]  beta[6]  beta[7]  beta[8]
##  0.91710  1.35274 -0.33915 -1.58368  0.15422  0.30901 -0.42904 -0.15590
##  beta[9] beta[10] beta[11] beta[12] beta[13] beta[14] beta[15] beta[16]
```

```
##  0.50199 -1.13194 -1.57502  0.28662 -1.32465  0.16800  1.07174  0.44354
## beta[17] beta[18] beta[19] beta[20] beta[21] beta[22] beta[23] beta[24]
##  0.01009  0.39073 -1.57967  0.21305 -1.11677  0.52895 -0.02012  0.77014
## beta[25] beta[26] beta[27] beta[28] beta[29] beta[30] beta[31] beta[32]
##  0.36720 -1.61691  0.67864  0.20742 -0.13784  0.26706  0.50668  1.88910
## beta[33] beta[34] beta[35] beta[36] beta[37] beta[38] beta[39] beta[40]
##  1.12091  0.15696 -0.11362 -0.02135  0.36963 -1.47267  0.75380  2.37252
## beta[41] beta[42] beta[43] beta[44] beta[45] beta[46] beta[47] beta[48]
## -2.81605  0.75032 -0.98050  1.57712 -0.67569  0.08521 -1.30521  1.29823
## beta[49] beta[50]
## -1.13099 -1.43491
##
##
## $p3[[2]]
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##   beta[1]   beta[2]   beta[3]   beta[4]   beta[5]   beta[6]   beta[7]   beta[8]
## -0.307293  0.528643 -0.693345 -1.225787  1.806405 -1.118417  0.065333  1.312381
##   beta[9]  beta[10]  beta[11]  beta[12]  beta[13]  beta[14]  beta[15]  beta[16]
##  1.452802 -1.023124  1.344375 -0.217658  0.666612 -1.350960  0.184612 -0.083585
##  beta[17]  beta[18]  beta[19]  beta[20]  beta[21]  beta[22]  beta[23]  beta[24]
##  0.665358  1.023250 -1.212197 -1.158831 -2.577788  1.038092  0.678531 -0.528886
##  beta[25]  beta[26]  beta[27]  beta[28]  beta[29]  beta[30]  beta[31]  beta[32]
## -0.933813 -0.741929  0.227717  0.061612 -0.849049  0.507283  0.924021  0.111097
##  beta[33]  beta[34]  beta[35]  beta[36]  beta[37]  beta[38]  beta[39]  beta[40]
## -0.859758  0.609015 -0.057646  1.744474  0.090916  0.138012  0.976446  0.166251
##  beta[41]  beta[42]  beta[43]  beta[44]  beta[45]  beta[46]  beta[47]  beta[48]
## -0.370009 -1.665621  1.679021  0.364750  1.866124 -1.037856 -2.118496 -0.007364
##  beta[49]  beta[50]
##  0.304895  1.507102
```

```r
gelman.diag(samples3)$psrf %>%
      kable(
        caption = "Gelman-Rubin Statistics for convergence"
      ) %>%
      kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

According to the Gelman-Rubin diagnostics, all the coefficients converge. There are a few covariates that don't show signs of convergence in the Geweke Statistics; however the Trace plots show that there does appear to be convergence.

Plot Comparison

```r
getDensity <- function(jags_data, names) {
  return (
    unionJagsOutput(jags_data, names) %>%
      group_by(var) %>%
      summarise(den_x = density(value)$x, den_y = density(value)$y)
  )
}
```

```r
d1 <- getDensity(samples1, names) %>% mutate(prior = "uninformative")
```

```
## `summarise()` regrouping output by 'var' (override with `.groups` argument)
```

```r
d2 <- getDensity(samples2, names) %>% mutate(prior = "Hierarchical Gaussian")
```

```
## `summarise()` regrouping output by 'var' (override with `.groups` argument)
```

```r
d3 <- getDensity(samples3, names) %>% mutate(prior = "BLASSO")
```

```
## `summarise()` regrouping output by 'var' (override with `.groups` argument)
```

```r
d_all <- union_all(d1, d2) %>% union_all(d3)

d_all %>%
  ggplot(aes(x = den_x, y = den_y, color = prior)) +
  geom_line() +
  geom_vline(xintercept = 0) +
  facet_wrap(~var, ncol = 5) +
  labs(x = expression(beta), y = "Posterior Density") +
  ggtitle("Comparing Posterior Densities across Priors")
```


Comparing Posterior Densities across Priors
OTU_104
OTU_105
OTU_128
OTU_147
OTU_15366
OTU_173
OTU_176
OTU_18758
OTU_206
OTU_23458
OTU_23469
OTU_235
OTU_27637
OTU_277
OTU_346
OTU_4160
OTU_42094
OTU_43955
OTU_44009
OTU_44577
OTU_45069
OTU_46867
OTU_49398
OTU_50134
OTU_51578
OTU_53271
OTU_54578
OTU_54646
OTU_559
OTU_61020
OTU_61577
OTU_624
OTU_64123
OTU_64716
OTU_6475
OTU_65560
OTU_65672
OTU_66
OTU_68686
OTU_68766
OTU_7270
OTU_72918
OTU_74414
OTU_75424
OTU_80
OTU_8086
OTU_81
OTU_9405
OTU_97
OTU_999
Posterior Density
β
prior
BLASSO
Hierarchical Gaussian
uninformative

There does not appear to be a big difference in the posterior distributions with different priors. Since the density lines have similar peaks, it does appear that these 50 OTUs are significant in all three fits.

Table 1: Gelman-Rubin Statistics for convergence

	Point est.	Upper C.I.
beta[1]	0.9998898	0.9999035
beta[2]	0.9995174	0.9995746
beta[3]	1.0015941	1.0047975
beta[4]	1.0002068	1.0014766
beta[5]	1.0041319	1.0201044
beta[6]	1.0001827	1.0027766
beta[7]	0.9998290	1.0006340
beta[8]	0.9997649	1.0001801
beta[9]	1.0009007	1.0048072
beta[10]	1.0011096	1.0011128
beta[11]	0.9996496	0.9997166
beta[12]	1.0021696	1.0126733
beta[13]	1.0013651	1.0079439
beta[14]	1.0013746	1.0047879
beta[15]	1.0013702	1.0082472
beta[16]	1.0004396	1.0040436
beta[17]	1.0018577	1.0091084
beta[18]	1.0015460	1.0032661
beta[19]	1.0006072	1.0020264
beta[20]	0.9998971	1.0006382
beta[21]	1.0021234	1.0042546
beta[22]	1.0008221	1.0037484
beta[23]	1.0012062	1.0023671
beta[24]	1.0000866	1.0015885
beta[25]	1.0005783	1.0028697
beta[26]	0.9998669	1.0008602
beta[27]	0.9998072	0.9998812
beta[28]	0.9997646	0.9997963
beta[29]	1.0011571	1.0027501
beta[30]	1.0015317	1.0083468
beta[31]	1.0008459	1.0030394
beta[32]	0.9996095	1.0000181
beta[33]	1.0008809	1.0058429
beta[34]	0.9997390	0.9997432
beta[35]	0.9996342	0.9996896
beta[36]	1.0010717	1.0010718
beta[37]	1.0020561	1.0115271
beta[38]	1.0025408	1.0138152
beta[39]	1.0007059	1.0050998
beta[40]	1.0002601	1.0006205
beta[41]	1.0014503	1.0014916
beta[42]	1.0012381	1.0078389
beta[43]	1.0056601	1.0190949
beta[44]	1.0002560	1.0008558
beta[45]	1.0003778	1.0033896
beta[46]	0.9996846	1.0000627
beta[47]	1.0001655	1.0024471
beta[48]	0.9995722	0.9997321
beta[49]	1.0013658	1.0032060
beta[50]	1.0004985	1.0016856

Table 2: Gelman-Rubin Statistics for convergence

	Point est.	Upper C.I.
beta[1]	1.0002939	1.0034722
beta[2]	0.9997115	0.9998522
beta[3]	1.0014599	1.0087745
beta[4]	0.9997114	1.0002225
beta[5]	1.0026353	1.0123030
beta[6]	1.0022969	1.0120532
beta[7]	1.0009612	1.0067850
beta[8]	0.9997323	1.0006214
beta[9]	1.0045841	1.0187501
beta[10]	1.0005394	1.0005794
beta[11]	0.9995631	0.9997888
beta[12]	1.0037335	1.0196079
beta[13]	1.0000874	1.0018506
beta[14]	1.0037017	1.0199265
beta[15]	0.9996014	0.9996054
beta[16]	1.0022449	1.0031019
beta[17]	1.0012419	1.0025523
beta[18]	1.0002237	1.0002267
beta[19]	0.9995268	0.9996345
beta[20]	1.0000272	1.0000511
beta[21]	0.9998184	1.0007431
beta[22]	1.0024521	1.0129253
beta[23]	1.0001287	1.0005990
beta[24]	1.0001965	1.0024613
beta[25]	0.9996194	0.9996321
beta[26]	1.0001340	1.0020879
beta[27]	1.0012172	1.0080518
beta[28]	1.0027133	1.0129422
beta[29]	1.0006939	1.0027822
beta[30]	1.0030644	1.0160363
beta[31]	1.0010859	1.0032557
beta[32]	1.0006903	1.0039346
beta[33]	1.0011113	1.0016476
beta[34]	1.0010944	1.0034625
beta[35]	1.0000604	1.0001676
beta[36]	1.0035959	1.0044636
beta[37]	1.0012050	1.0033896
beta[38]	1.0007778	1.0015849
beta[39]	1.0002442	1.0031910
beta[40]	0.9996104	0.9996174
beta[41]	1.0032251	1.0127006
beta[42]	1.0006970	1.0045176
beta[43]	1.0003703	1.0020619
beta[44]	1.0015913	1.0065665
beta[45]	1.0004183	1.0040525
beta[46]	0.9996025	0.9997657
beta[47]	0.9995088	0.9995125
beta[48]	0.9996200	0.9996281
beta[49]	0.9996637	1.0001630
beta[50]	1.0001644	1.0028260

Table 3: Gelman-Rubin Statistics for convergence

	Point est.	Upper C.I.
beta[1]	1.0011390	1.0067760
beta[2]	1.0005043	1.0015345
beta[3]	1.0000594	1.0009836
beta[4]	0.9996640	0.9998131
beta[5]	0.9996688	0.9996689
beta[6]	1.0004854	1.0042891
beta[7]	1.0008989	1.0042929
beta[8]	0.9999275	1.0014279
beta[9]	1.0009779	1.0052391
beta[10]	0.9998478	1.0011251
beta[11]	1.0003438	1.0035868
beta[12]	0.9996612	1.0001605
beta[13]	1.0017458	1.0101338
beta[14]	1.0016487	1.0098900
beta[15]	1.0004608	1.0012426
beta[16]	0.9999232	1.0014199
beta[17]	1.0021813	1.0052303
beta[18]	1.0007954	1.0010367
beta[19]	0.9998173	0.9999427
beta[20]	1.0022178	1.0091540
beta[21]	1.0003848	1.0037324
beta[22]	1.0039876	1.0214503
beta[23]	1.0003589	1.0032636
beta[24]	1.0003291	1.0031533
beta[25]	1.0009061	1.0021126
beta[26]	0.9999043	1.0014277
beta[27]	0.9997144	1.0004479
beta[28]	1.0001305	1.0010908
beta[29]	1.0005179	1.0044060
beta[30]	1.0016870	1.0087791
beta[31]	1.0004518	1.0013284
beta[32]	0.9997001	0.9999314
beta[33]	1.0005309	1.0017370
beta[34]	1.0016111	1.0084702
beta[35]	1.0032699	1.0088491
beta[36]	0.9999092	1.0002556
beta[37]	1.0013795	1.0029459
beta[38]	1.0003767	1.0013927
beta[39]	1.0006949	1.0051626
beta[40]	1.0027714	1.0129753
beta[41]	1.0023044	1.0130794
beta[42]	1.0015353	1.0092630
beta[43]	1.0015712	1.0044908
beta[44]	0.9995818	0.9997061
beta[45]	1.0000231	1.0019353
beta[46]	1.0063096	1.0255429
beta[47]	1.0002741	1.0029486
beta[48]	1.0020472	1.0031674
beta[49]	1.0052263	1.0213349
beta[50]	1.0027089	1.0152344