Homework #8

Dustin Leatherman

November 9, 2020

In this assignment, you will perform random slopes logistic regression in JAGS using the Gambia data described in gambia.csv.

Let Y_i be the binary response for individual i testing positive for malaria, $\nu_i \in \{1, ..., 65\}$ denote the village of the individual i and $X_i = 1$ if individual i regularly sleeps under a bed-net and $X_i = 0$ otherwise. Fit the model

$$logit[P(Y_i = 1)] = \alpha_{\nu_i} + X_i * \beta_{\nu_i}$$

where α and β_j are the intercept and slope for village j. The priors (independent over village and with each other) are $\alpha \sim N(\mu_\alpha, \sigma^2_\alpha)$ and $\beta_j \sim N(\mu_b, \sigma^2_b)$. Choose uninformative priors for the hyper-parameters $\mu_\alpha, \mu_b, \sigma^2_\alpha, \sigma^2_b$ In your report, address the following questions: 1. Scientifically, why might the effect of bed-net vary by village? 2. Did the MCMC algorithm converge? 3. Do you see evidence that the slopes and/or intercepts vary by village? 4. Which village has the largest intercept? Slope? Does this agree with the data in these villages?

1

Scientifically, why might the effect of bed-net vary by village?

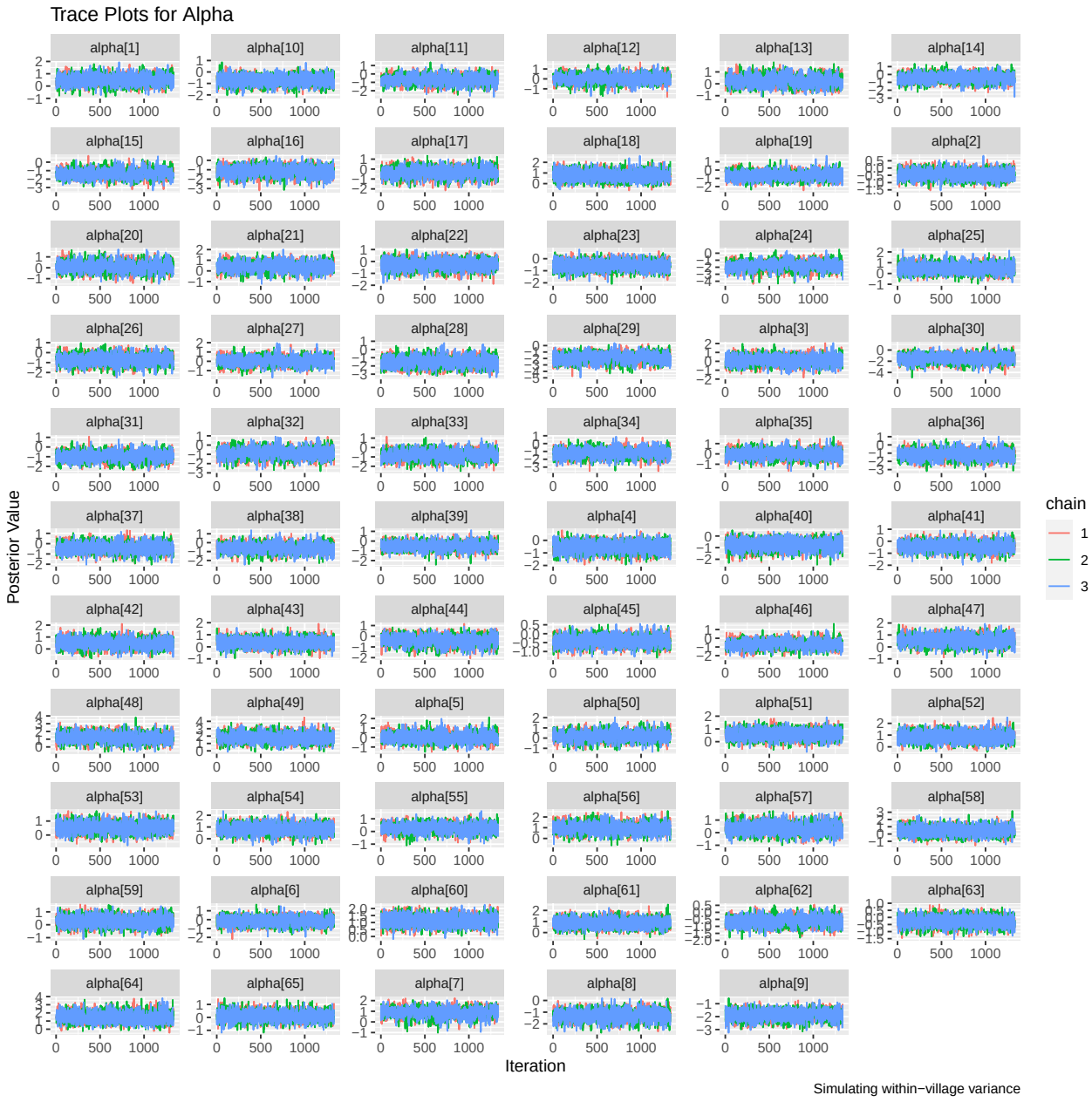
The villages may have be in various degrees of forest cover which impacts the number of mosquitos present. If there are more mosquitos present, there is a higher probability that one slips through the bed net as a person enters or exits.

```
n <- dim(gambia)[1]

Y <- gambia$Y
X <- gambia$X
v <- gambia$v
distinct.v <- unique(v)

data <- list(Y = Y, X = X, v = v, n = n, distinct.v = distinct.v)
params <- c("beta", "alpha")

# Settings (automatically calculates the number of iterations needed based on inputs)
nBurn <- 15000
nChains <- 3
nSave <- 4000
nThin <- 3
nIter <- ceiling((nSave*nThin)/nChains)

unionJagsOutput <- function(jags_data) {
  data <- NULL
```

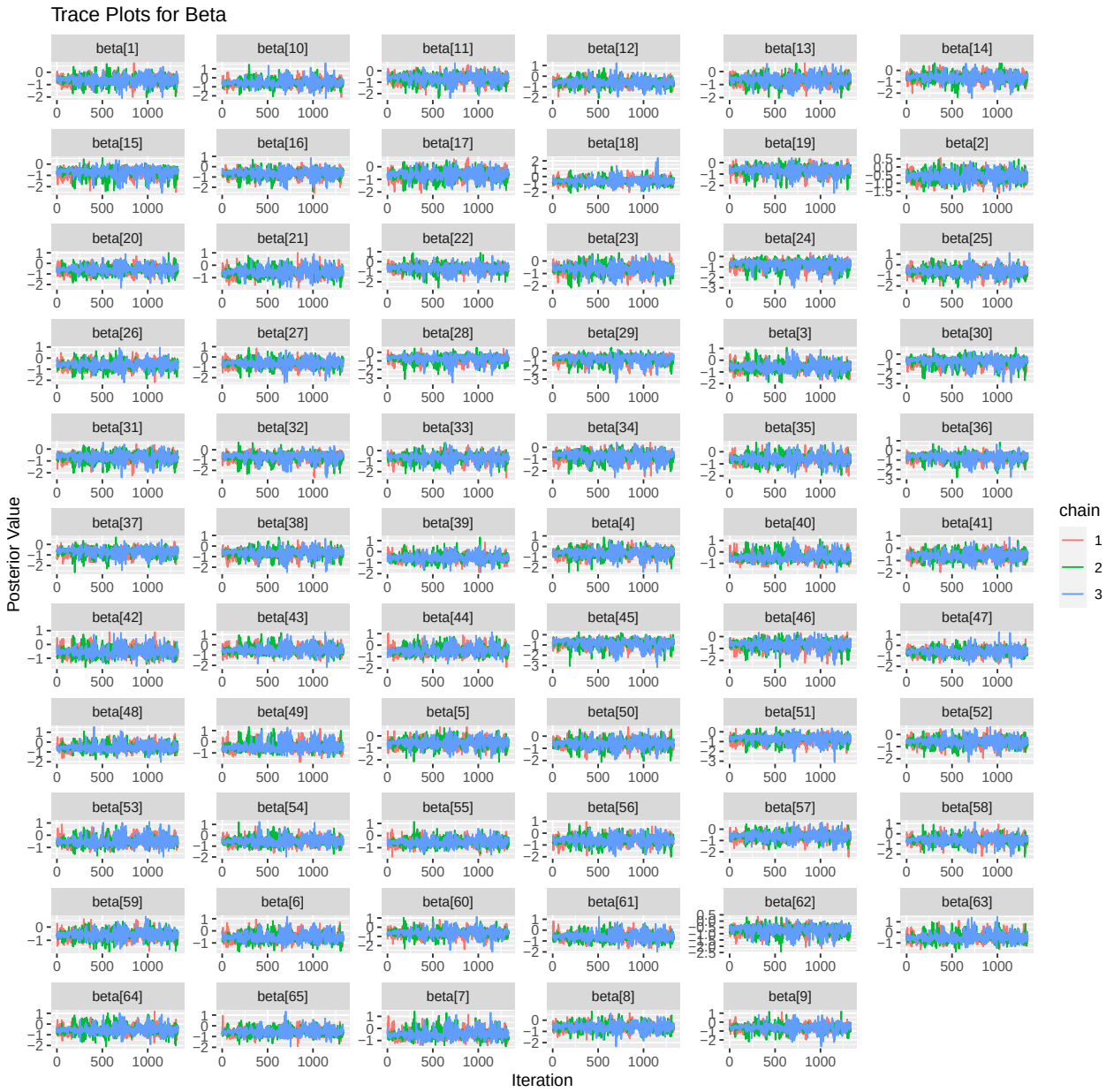
```
  for(chain in 1:length(jags_data)) {
    new.data <-
      jags_data[[chain]] %>%
        as_tibble() %>%
        mutate(chain = factor(chain), row_num = row_number()) %>%
        pivot_longer(-c(chain, row_num), names_to = "col", values_to = "value")

    if(is.null(data)) {
      data <- new.data
    } else {
      data <- union_all(data, new.data)
    }
  }
  return (data)
}
```

```
model_string <- textConnection("model{
  # Likelihood
  for(i in 1:n){
    Y[i] ~ dbern(p[i])
    logit(p[i]) <- alpha[v[i]] + X[i] * beta[v[i]]
  }

  # Priors
  for(j in distinct.v){
    alpha[j] ~ dnorm(mu[1], tau[1])
    beta[j] ~ dnorm(mu[2], tau[2])
  }

  for(z in 1:2) {
    mu[z] ~ dnorm(0, 0.001)
    tau[z] ~ dgamma(0.01, 0.01)
  }
}")

model <- jags.model(model_string,data=data,n.chains=nChains, quiet = TRUE)
update(model,burn=nBurn,progress.bar="none")
model1.out <- coda.samples(model,variable.names=params,thin=nThin,n.iter=nIter, progress.bar = "none")
model1.data <- unionJagsOutput(model1.out)
```

```
model1.data %>%
  filter(str_detect(col, "alpha")) %>%
    ggplot(aes(x = row_num, y = value, color = chain)) +
      geom_line() +
      facet_wrap(~col, ncol = 6, scales = "free") +
      labs(x = "Iteration", y = "Posterior Value", caption = "Simulating within-village variance", titl
```



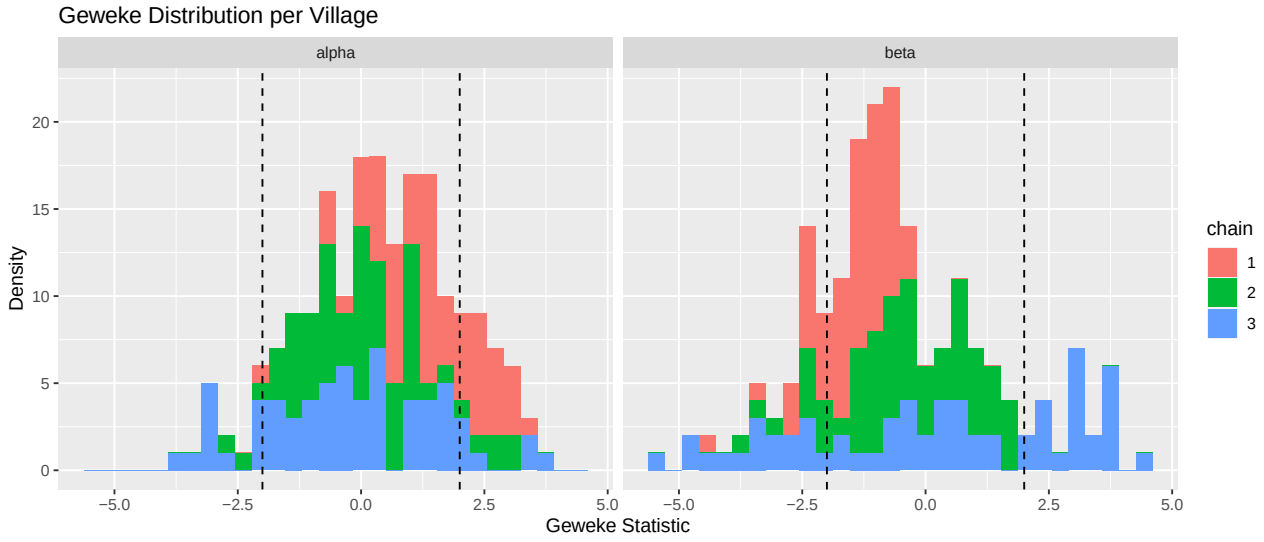

```
model1.data %>%
  filter(str_detect(col, "beta")) %>%
    ggplot(aes(x = row_num, y = value, color = chain)) +
      geom_line() +
      facet_wrap(~col, ncol = 6, scales = "free") +
      labs(x = "Iteration", y = "Posterior Value", caption = "Simulating the effect of Mosquito Nets on
```




```
g.out <- geweke.diag(model1.out)

geweke.data <- NULL
for(chain in 1:nChains){
  new.data <- g.out[[chain]]$z %>%
    as.data.frame() %>%
    rownames_to_column(var = "col") %>%
    rename(value = ".") %>%
    mutate(
      chain = factor(chain)
    )

  #colnames(new.data) <- c("col", "chain")
  if(is.null(geweke.data)) {
```

```
    geweke.data <- new.data
  } else {
    geweke.data <- union_all(geweke.data, new.data)
  }
}

geweke.data <-
  geweke.data %>%
    mutate(
      col = str_replace_all(col, "(\\[|,)", "_") %>% str_remove("\\]"),
      sim_row = str_split(col, "_", simplify = TRUE)[,2],
      group = str_split(col, "_", simplify = TRUE)[,1]
    )

geweke.data %>%
  ggplot(aes(x = value, fill = chain)) +
    geom_histogram() +
    facet_wrap(~group) +
    geom_vline(xintercept = -2, linetype = "dashed") +
    geom_vline(xintercept = 2, linetype = "dashed") +
    scale_color_discrete(name = "Chain") +
    labs(x = "Geweke Statistic", y = "Density", title = "Geweke Distribution per Village", caption = "H
```

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



Histograms highlighting the convergence of the chains using the Geweke Statistic. It is ideal to be between the dotted lines.

```
gelman.out <- gelman.diag(model1.out)
gelman.out$psrf %>%
  as.data.frame() %>%
  rownames_to_column(var = "col") %>%
  as_tibble() %>%
  summarise_at(c("Point est.", "Upper C.I."), list(quantile)) %>%
  mutate(
    quantile = c("min", "0.25", "0.5", "0.75", "max")
  ) %>%
  select(quantile, "Point est.", "Upper C.I.") %>%
      kable(
```

```
    caption = "Gelman-Rubin Statistic Quantiles to measure convergence of chains"
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

Table 1: Gelman-Rubin Statistic Quantiles to measure convergence of chains

quantile	Point est.	Upper C.I.
min	0.9997384	0.9998723
0.25	1.0022689	1.0059763
0.5	1.0071980	1.0144127
0.75	1.0168183	1.0334601
max	1.0397086	1.0908818

2

Did the MCMC algorithm converge?

The Trace Plots, Geweke Distributions, and Gelman-Rubin Statistics indicate that the model has converged.

3

Do you see evidence that slopes and/or intercepts vary by village?

```
post.means <-
  model1.data %>%
    group_by(col) %>%
    summarise(post.mean = mean(value)) %>%
    mutate(
        col = str_replace_all(col, "(\\[|,)", "_") %>% str_remove("\\]"),
        sim_row = str_split(col, "_", simplify = TRUE)[,2] %>% as.integer(),
        group = str_split(col, "_", simplify = TRUE)[,1]
      ) %>%
    arrange(group, sim_row)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
post.means %>%
  filter(group == "alpha") %>%
  summarize(value = quantile(post.mean, probs = c(0, 0.05, 0.25, 0.5, 0.75, 0.95, 1))) %>%
  mutate(quantile = c("min", "5%", "25%", "median", "75%", "95%", "max")) %>%
  select(quantile, value) %>%
  kable(
    caption = "Quantiles of the Posterior Means for Intercepts"
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

```
post.means %>%
  filter(group == "beta") %>%
  summarize(value = quantile(post.mean, probs = c(0, 0.05, 0.25, 0.5, 0.75, 0.95, 1))) %>%
  mutate(quantile = c("min", "5%", "25%", "median", "75%", "95%", "max")) %>%
  select(quantile, value) %>%
  kable(
```

Table 2: Quantiles of the Posterior Means for Intercepts

quantile	value
min	-1.8611094
5%	-1.4335310
25%	-0.6818909
median	-0.2614942
75%	0.4538062
95%	1.0741154
max	1.7598681

```
    caption = "Quantiles of the Posterior Means for Slopes"
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

Table 3: Quantiles of the Posterior Means for Slopes

quantile	value
min	-0.8434368
5%	-0.7616219
25%	-0.6708556
median	-0.6093004
75%	-0.5647803
95%	-0.4717323
max	-0.4205457

Slopes don't appear to vary much between village but intercepts range from -1.87 to 1.77, meaning that without taking into account bed nets, there is a predisposition to Malaria based on the village. This may indicate that there is information not being taken into account in this model.

4

Which village has the largest intercept? Slope? Does this agree with the data in these villages?

```
gambia %>%
  group_by(v) %>%
  summarise(infected = sum(Y), n = n(), rate = infected / n) %>%
  arrange(desc(rate)) %>%
  head(n = 10) %>%
  kable(
    caption = "Infected Rates per village"
  ) %>%
  kable_styling(full_width = T, bootstrap_options = "striped", latex_options = "hold_position")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

Village 49 has the largest intercept. The data shows that village 49 has the highest estimated effected rate via the MLE.

Table 4: Infected Rates per village

v	infected	n	rate
49	14	15	0.9333333
64	12	13	0.9230769
48	9	11	0.8181818
60	29	37	0.7837838
18	15	21	0.7142857
61	20	28	0.7142857
54	19	29	0.6551724
56	19	29	0.6551724
25	13	20	0.6500000
51	19	30	0.6333333