Running MCMC in JAGS via `runjags` – Another example

Stat 340, Fall 2021

Do Distracting Colors Influence the Time to Complete a Game?

The Stroop effect demonstrates that automatized behaviors[1] can interfere with other desired behaviors. John Stroop tested the reaction time of college undergraduates in identifying colors. Students took a longer time identifying colors of ink when the ink was used to spell a different color. For example, if the word "red" was printed in blue ink, students took longer to identify the color blue because they automatically read the word "red." Even though students were told only to identify the ink color, the automatized behavior of reading interfered with the task and slowed their reaction time.

Statistics students developed a study to explore the impact of distracters. Specifically, they wanted to determine whether students at their college would perform differently when a distracting color was incorporated into a computerized game. The game challenged people to place an assortment of shaped pegs into the appropriate spaces as quickly as possible. Below are additional details of the study:

- 40 students were randomly selected from the college
- 20 students were randomly assigned to the standard game, the remaining 20 students were assigned to a game with a color distracter
- Subjects saw a picture of the game and had the rules clearly explained to them before they played
- Subjects played the game in the same area with similar background noise
- The research group collected the the time, in seconds, required to complete the game

```r
stroop <- read.csv("http://aloy.rbind.io/data/stroop_game.csv")
```

A noninformative two-sample Bayesian model for this situations can be written as

$$
\begin{aligned}
Y_i &\overset{\text{iid}}{\sim} \mathcal{N}(\mu, \sigma^2),\ i = 1, \ldots, 20 && \text{(standard group)}\\
Y_j &\overset{\text{iid}}{\sim} \mathcal{N}(\mu + \delta, \sigma^2),\ j = 21, \ldots, 40 && \text{(color group)}\\
\mu &\sim \mathcal{N}(0, 100)\\
\delta &\sim \mathcal{N}(0, 100)\\
1/\sigma^2 &\sim \text{Gamma}(0.01, 0.01)
\end{aligned}
$$

where μ, δ, and σ^2 are assumed to be mutually independent.

1. Use the below code to load the data and prepare it for use with JAGS.

```r
# Split data into standard (std) and color (col) groups
y <- stroop$Time
y.std <- y[stroop$Type == "Standard"]
y.col <- y[stroop$Type == "Color"]
n <- length(y.std)
m <- length(y.col)

# Create a list of data for the model
stroop_data <- list(y.std = y.std, y.col = y.col, n = n, m = m)
```

2. Complete the model below for this situation.

```r
stroop_model_string <- "
model{
  # Specify the likelihood for the standard group
  for(i in 1:n) {
y.std[i] ~ dnorm(mu, phi)
  }
```

[1] Many psychologists would call this procedural knowledge instead of automatized behavior.

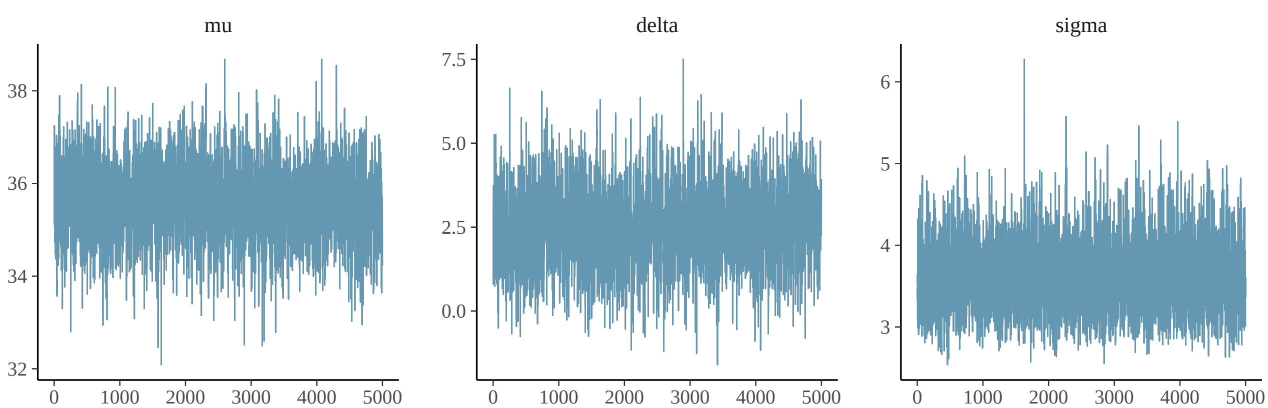
```
  # Specify the likelihood for the color group
  for(j in 1:m) {
y.col[j] ~ dnorm(mu + delta, phi)
  }

  # Specify the prior distributions
  mu ~ dnorm(0, 1 / pow(100, 2))      # pow(100, 2) calculates 100^2
  delta ~ dnorm(0, 1 / pow(100, 2))
  phi ~ dgamma(.01, .01)

  # Calculate sigma from phi
  sigma <- 1 / sqrt(phi)
}"
```

3. Use `run.jags()` to obtain 5000 draws from the joint posterior distribution of μ, δ, and σ. Use a burn-in period of 1000 draws, and an adaptation phase of 1000 draws.

```
library(runjags)
stroop_samples <- run.jags(
  stroop_model_string,
  data = stroop_data,
  monitor = c("mu", "delta", "sigma"),
  n.chains = 1,
  burnin = 1000,
  adapt = 1000,
  sample = 5000,
  silent.jags = TRUE
  )
```

4. Create a trace plot by running the below code. Did the chain(s) for μ, δ, and σ converged? How do you know?

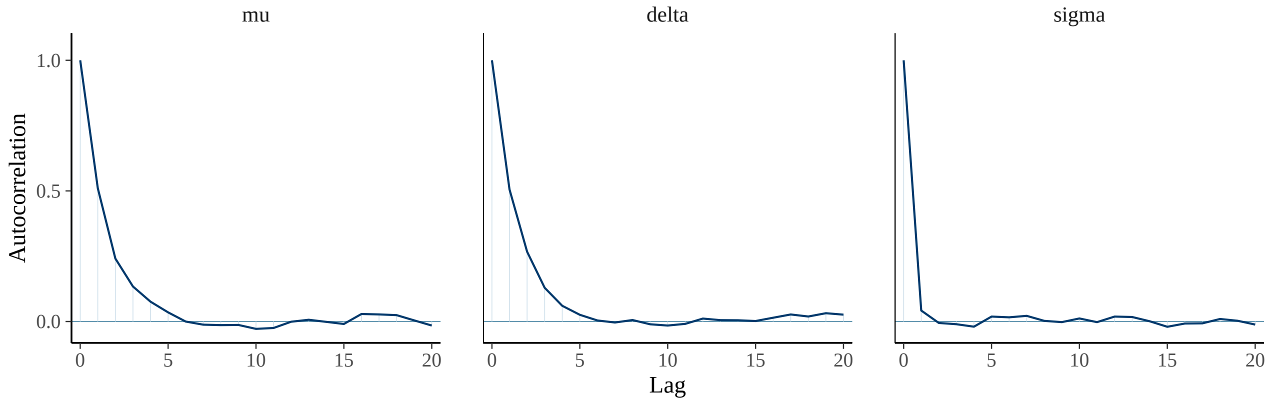
```
library(bayesplot)
mcmc_trace(stroop_samples$mcmc)
```



Yes, the chains have converged. All of the trace plots have a clear center of mass and appear to be "white noise" around this center of mass. (In other words, they look like fuzzy catepillars.)

5. Create a plot of the autocorrelation function of the posterior draws using the below code. Do you have any concerns?

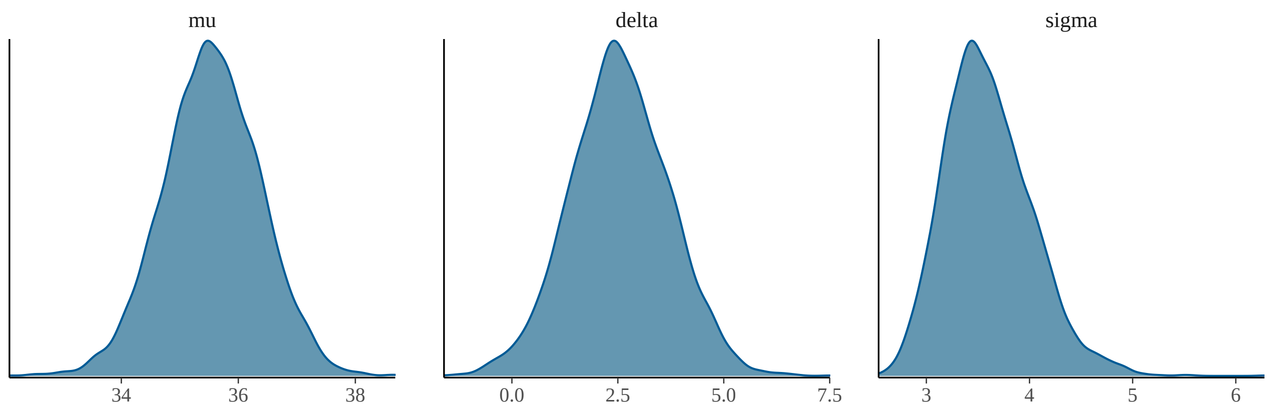
```
mcmc_acf(stroop_samples$mcmc)
```



All of the ACF plots appear to decay quite quickly, indicating there is not strong temporal correlation within a chain. Consequently, we should be exploring the parameter space in a relatively efficient manner.

6. Plot the marginal posterior density for each parameter after discarding the burn-in period by running the following command.

```
mcmc_dens(stroop_samples$mcmc)
```



7. Calculate the posterior mean and SD for each parameter.

   The mean and SD are given in the output of summary().

```
summary(stroop_samples)[,c("Mean", "SD")]
```

```
##            Mean        SD
## mu    35.546805 0.8235487
## delta  2.549174 1.1644637
## sigma  3.602730 0.4208271
```

8. Calculate a 95% credible interval for δ. What does this say about the difference in average completion time between groups?

   You could use the 95% credible interval produced by `summary()`; however, this is a highest posterior density credible interval, which we have not discussed yet. To get an equal-tail credible interval, we can use `quantile()` as before:

```
quantile(unlist(stroop_samples$mcmc[,"delta"]), probs = c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.2484655 4.8145570
```

9. The full conditional posterior distributions can be shown to be:

$$\mu|\text{rest} \sim \mathcal{N}\left(\frac{\phi \sum_{i=1}^{n} Y_i + \phi \sum_{j=n+1}^{n+m}(Y_j - \delta)}{\phi(n+m) + 1/100^2}, \ \frac{1}{\phi(n+m)+1/100^2}\right)$$

$$\delta|\text{rest} \sim \mathcal{N}\left(\frac{\phi \sum_{i=n+1}^{n+m}(Y_i - \mu)}{\phi m + 1/100^2}, \ \frac{1}{\phi m + 1/100^2}\right)$$

$$1/\sigma^2|\text{rest} \sim \text{Gamma}\left(0.01 + \frac{n+m}{2}, \ 0.01 + \frac{\sum_{i=1}^{n}(Y_i-\mu)^2}{2} + \frac{\sum_{j=n+1}^{n+m}(Y_j - \mu - \delta)^2}{2}\right)$$

where $\phi = 1/\sigma^2$. Outline the steps of a Gibbs sampler that can be used to draw samples from the joint posterior distribution.

For $i = 1, \ldots, 5000$ do the following:

i. Draw $\mu^{(i)}$ from $\mathcal{N}\left(\frac{\phi^{(i-1)} \sum_{i=1}^{n} Y_i + \phi^{(i-1)} \sum_{j=n+1}^{n+m}(Y_j - \delta\phi^{(i-1)})}{\phi^{(i-1)}(n+m) + 1/100^2}, \ \frac{1}{\phi^{(i-1)}(n+m)+1/100^2}\right)$.

ii. Draw $\delta^{(i)}$ from $\mathcal{N}\left(\frac{\phi^{(i-1)} \sum_{i=n+1}^{n+m}(Y_i - \mu^{(i)})}{\phi^{(i-1)} m + 1/100^2}, \ \frac{1}{\phi^{(i-1)} m + 1/100^2}\right)$.

iii. Draw $\phi^{(i)}$ from $\text{Gamma}\left(0.01 + \frac{n+m}{2}, \ 0.01 + \frac{\sum_{i=1}^{n}(Y_i-\mu^{(i)})^2}{2} + \frac{\sum_{j=n+1}^{n+m}(Y_j - \mu^{(i)} - \delta^{(i)})^2}{2}\right)$