

# Summarizing posteriors through tables in RMarkdown

*Johannes Karreth*

*July 4, 2016*

## Simulate data and fit a model

```
n.sim <- 100; set.seed(123)
x1 <- rnorm(n.sim, mean = 5, sd = 2)
x2 <- rbinom(n.sim, size = 1, prob = 0.3)
e <- rnorm(n.sim, mean = 0, sd = 1)
b1 <- 1.2
b2 <- -3.1
a <- 1.5
y <- a + b1 * x1 + b2 * x2 + e

sim.dat <- data.frame(y, x1, x2)

bayes.mod <- function() {

  for(i in 1:N){
    y[i] ~ dnorm(mu[i], tau)
    mu[i] <- alpha + beta1 * x1[i] + beta2 * x2[i]
  }

  alpha ~ dnorm(0, .01)
  beta1 ~ dunif(-100, 100)
  beta2 ~ dunif(-100, 100)
  tau ~ dgamma(.01, .01)

}

sim.dat.jags <- as.list(sim.dat)

sim.dat.jags$N <- nrow(sim.dat)

bayes.mod.params <- c("alpha", "beta1", "beta2")

inits1 <- list("alpha" = 0, "beta1" = 0, "beta2" = 0)
inits2 <- list("alpha" = 1, "beta1" = 1, "beta2" = 1)
inits3 <- list("alpha" = -1, "beta1" = -1, "beta2" = -1)
bayes.mod.inits <- list(inits1, inits2, inits3)

library(R2jags)

## Loading required package: rjags

## Loading required package: coda

## Linked to JAGS 4.2.0
```

```
## Loaded modules: basemod,bugs
```

```
##
```

```
## Attaching package: 'R2jags'
```

```
## The following object is masked from 'package:coda':
```

```
##
```

```
##      traceplot
```

```
set.seed(123)
```

```
bayes.mod.fit <- jags(data = sim.dat.jags, inits = bayes.mod.inits,  
  parameters.to.save = bayes.mod.params, n.chains = 3, n.iter = 9000,  
  n.burnin = 1000,  
  model.file = bayes.mod)
```

```
## module glm loaded
```

```
## Compiling model graph
```

```
##   Resolving undeclared variables
```

```
##   Allocating nodes
```

```
## Graph information:
```

```
##   Observed stochastic nodes: 100
```

```
##   Unobserved stochastic nodes: 4
```

```
##   Total graph size: 516
```

```
##
```

```
## Initializing model
```

## Create a summary table

```
devtools::source_url("https://raw.githubusercontent.com/jkarreth/JKmisc/master/mcmctab.R")
```

```
## SHA-1 hash of file is aaf011f39a5b02aa830972cda46ebd378fc45d9f
```

```
regtable <- mcmctab(as.mcmc(bayes.mod.fit))[-c(4), ] # remove the fourth row
```

## Option 1: the pander package

```
library(pander)  
pander(regtable)
```

	Mean	SD	Lower	Upper	Pr
<b>alpha</b>	1.849	0.294	1.259	2.427	1
<b>beta1</b>	1.134	0.052	1.032	1.237	1
<b>beta2</b>	-3.086	0.212	-3.494	-2.667	1

## Option 2: the knitr package

```
library(knitr)
kable(regtable)
```

	Mean	SD	Lower	Upper	Pr
alpha	1.849	0.294	1.259	2.427	1
beta1	1.134	0.052	1.032	1.237	1
beta2	-3.086	0.212	-3.494	-2.667	1