# ADVANCED BAYESIAN MODELING

## JAGS Analysis: First Attempt

$$\begin{aligned} y_{ijk} &= \text{number of damage incidents to ships in category } (i,j,k) \\ t_{ijk} &= \text{total months of service represented by category } (i,j,k) \end{aligned}$$

$i$ : type of ship      $j$ : era of construction      $k$ : period of operation

## Loglinear Rate Model

$$y_{ijk} \mid \beta, t_{ijk} \quad \sim \quad \text{indep. Poisson}(\lambda_{ijk}) \qquad \lambda_{ijk} \;=\; t_{ijk}\, r_{ijk}$$

$$\log \lambda_{ijk} \;=\; \log(t_{ijk}\, r_{ijk}) \;=\; \log t_{ijk} + \beta^0 + \beta_i^t + \beta_j^e + \beta_k^p$$

$$\beta_i^t \mid \sigma_t^2 \;\sim\; \text{iid}\, \mathrm{N}(0, \sigma_t^2) \qquad \beta_j^e \mid \sigma_e^2 \;\sim\; \text{iid}\, \mathrm{N}(0, \sigma_e^2) \qquad \beta_k^p \mid \sigma_p^2 \;\sim\; \text{iid}\, \mathrm{N}(0, \sigma_p^2)$$

$$\beta^0 \;\sim\; \mathrm{N}(0, 100^2) \qquad \sigma_t, \sigma_e, \sigma_p \;\sim\; \text{iid}\;\; \mathrm{U}(0, 100)$$

In file `ships1.bug`:

```
model {
  for (i in 1:length(incidents)) {
    incidents[i] ~ dpois(lambda[i])
    log(lambda[i]) <- logservice[i] + beta0 + beta.t[type[i]] + beta.e[era[i]] +
                      beta.p[period[i]]

    incidentsrep[i] ~ dpois(lambda[i])
  }
```

(continued ...)

Anticipating posterior predictive checking, we define replicated data `incidentsrep`.

```
  beta0 ~ dnorm(0, 0.0001)
  for (i in 1:max(type)) {
    beta.t[i] ~ dnorm(0, 1/sigma.t^2)
  }
  for (j in 1:max(era)) {
    beta.e[j] ~ dnorm(0, 1/sigma.e^2)
  }
  for (k in 1:max(period)) {
    beta.p[k] ~ dnorm(0, 1/sigma.p^2)
  }

  sigma.t ~ dunif(0, 100)
  sigma.e ~ dunif(0, 100)
  sigma.p ~ dunif(0, 100)
}
```

Set up data and initializations for four chains:

```
> d1 <- list(incidents = shipssub$incidents,
+            logservice = log(shipssub$service),
+            type = unclass(shipssub$type),
+            era = unclass(factor(shipssub$year)),
+            period = unclass(factor(shipssub$period)))

> inits1 <- list(list(beta0=100, sigma.t=90, sigma.e=90, sigma.p=90),
+                list(beta0=-100, sigma.t=0.01, sigma.e=90, sigma.p=90),
+                list(beta0=100, sigma.t=90, sigma.e=0.01, sigma.p=0.01),
+                list(beta0=-100, sigma.t=0.01, sigma.e=0.01, sigma.p=0.01))
```

Recall `unclass` produces the integer codes of a factor.

```
> library(rjags)
...

> m1 <- jags.model("ships1.bug", d1, inits1, n.chains=4, n.adapt=1000)
...

> update(m1, 1000)  # burn-in
  |**************************************************| 100%

> x1 <- coda.samples(m1, c("beta0","sigma.t","sigma.e","sigma.p"), n.iter=2000)
  |**************************************************| 100%
```

Note: For checking convergence, we monitor only the top-level nodes.

```
> gelman.diag(x1, autoburnin=FALSE)
Potential scale reduction factors:

        Point est. Upper C.I.
beta0        90.57     196.05
sigma.e       1.35       1.89
sigma.p       1.02       1.06
sigma.t       1.17       1.46

Multivariate psrf

74
```

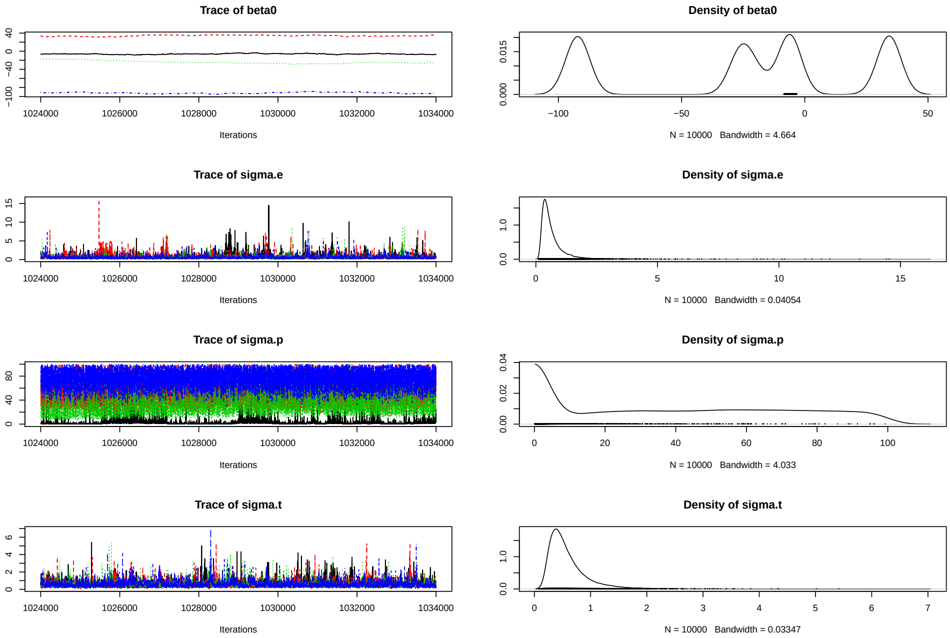
I tried repeatedly running the chains and checking for convergence, doubling the number of iterations each time.

Even after over a million iterations, there was still no convergence.

Trace plots suggest why:

```
> x1 <- coda.samples(m1, c("beta0","sigma.t","sigma.e","sigma.p"), n.iter=10000)
  |**************************************************| 100%

> plot(x1, smooth=FALSE)
```


Trace of beta0
Iterations
Density of beta0
N = 10000 Bandwidth = 4.664
Trace of sigma.e
Iterations
Density of sigma.e
N = 10000 Bandwidth = 0.04054
Trace of sigma.p
Iterations
Density of sigma.p
N = 10000 Bandwidth = 4.033
Trace of sigma.t
Iterations
Density of sigma.t
N = 10000 Bandwidth = 0.03347

The chain shows dramatically slow mixing for $\beta^0$ and $\sigma_p$, and some chains have $\sigma_p$ values near the arbitrary upper bound of 100.

Since the data has only two periods of operation $(k = 1, 2)$, it is possible that $\sigma_p^2$ is poorly defined – it is the variance of only two random effects.

An improved model might treat $\beta_1^p$ and $\beta_2^p$ as fixed effects instead.