Dugong – HW 9

Garrett Carr

Due March 29, 2022

These data concern growth of dugongs (sometimes called sea cows, an aquatic mammal found primarily in the Indo-West Pacific). The data file is called 'dugong.dat' and contains two columns, the age of the animal in years, and the length of the animal in meters for 27 individuals. If you plot the data, you will see that growth is faster for young animals and then seems to stop in older animals. This type of growth is called nonlinear growth, and the simplest curve to describe such growth is:

$$y_i = a - bg^{x_i}, \tag{1}$$

where y_i represents the length of the animal and x_i represents the age of the animal. As you can see, there are three parameters to estimate, a, b, and g. 'a' represents the asymptote or value at which growth stops, 'b' is constrained to be positive, and 'g' is constrained to be between 0 and 1.

You have two tasks:

1. Determine parameter estimates for the growth curve.
2. Compare the results from JAGS to those from Stan.

After making sure chains have converged appropriately, you will compare the results by examining both equal tail and HPD interval estimates of the parameters (and functions thereof) that matter.

You should hand in a no more than a one page summary of your results that should include at least one plot. Make sure to address all appropriate model diagnostics and convergence, but note that you might not be able to fit code output in your one page summary. All code should come in as an appendix to your one page summary.

Summary

I first ran the model in JAGS, and obtained the following results:

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 4
##    Total graph size: 128
##
## Initializing model
```

```
## Inference for Bugs model at "model1.txt", fit using jags,
##  4 chains, each with 50000 iterations (first 2000 discarded)
```

```
##  n.sims = 192000 iterations saved
##          mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
## a          2.658   0.077   2.529   2.606   2.651   2.701   2.831 1.001 36000
## b          0.982   0.080   0.829   0.929   0.980   1.032   1.146 1.001 31000
## g          0.863   0.034   0.786   0.845   0.867   0.886   0.919 1.001 11000
## s2error    0.011   0.004   0.006   0.008   0.010   0.013   0.020 1.001 75000
## deviance -48.609   3.474 -52.935 -51.161 -49.405 -46.897 -39.792 1.001 30000
##
## For each parameter, n.eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor (at convergence, Rhat=1).
##
## DIC info (using the rule, pD = var(deviance)/2)
## pD = 6.0 and DIC = -42.6
## DIC is an estimate of expected predictive error (lower deviance is better).
```

```
##                  lower        upper
## a          2.518030089   2.81303670
## b          0.823503674   1.13881610
## deviance -53.417501091 -41.79498584
## g          0.795065576   0.92426316
## s2error    0.005212471   0.01835893
## attr(,"Probability")
## [1] 0.95
```

```
##     2.5%    97.5%
## 2.529287 2.831464
```

```
##      2.5%     97.5%
## 0.8294786 1.1459334
```

```
##      2.5%     97.5%
## 0.7864701 0.9186779
```

```
##        2.5%       97.5%
## 0.005984892 0.020167184
```

It seems like there is a high level of autocorrelation for the JAGS output, so using a different sampler would help solve the issue.

```
## Running MCMC with 4 parallel chains...
##
## Chain 4 Iteration:     1 / 14000 [  0%]  (Warmup)
## Chain 4 Iteration:   100 / 14000 [  0%]  (Warmup)
## Chain 3 Iteration:     1 / 14000 [  0%]  (Warmup)
## Chain 3 Iteration:   100 / 14000 [  0%]  (Warmup)
## Chain 3 Iteration:   200 / 14000 [  1%]  (Warmup)
## Chain 3 Iteration:   300 / 14000 [  2%]  (Warmup)
## Chain 3 Iteration:   400 / 14000 [  2%]  (Warmup)
## Chain 3 Iteration:   500 / 14000 [  3%]  (Warmup)
## Chain 3 Iteration:   600 / 14000 [  4%]  (Warmup)
## Chain 3 Iteration:   700 / 14000 [  5%]  (Warmup)
## Chain 3 Iteration:   800 / 14000 [  5%]  (Warmup)
```

```
## Chain 3 Iteration:   900 / 14000 [  6%]  (Warmup)
## Chain 3 Iteration:  1000 / 14000 [  7%]  (Warmup)
## Chain 4 Iteration:   200 / 14000 [  1%]  (Warmup)
## Chain 4 Iteration:   300 / 14000 [  2%]  (Warmup)
## Chain 4 Iteration:   400 / 14000 [  2%]  (Warmup)
## Chain 4 Iteration:   500 / 14000 [  3%]  (Warmup)
## Chain 4 Iteration:   600 / 14000 [  4%]  (Warmup)
## Chain 4 Iteration:   700 / 14000 [  5%]  (Warmup)
## Chain 4 Iteration:   800 / 14000 [  5%]  (Warmup)
## Chain 4 Iteration:   900 / 14000 [  6%]  (Warmup)
## Chain 4 Iteration:  1000 / 14000 [  7%]  (Warmup)
## Chain 4 Iteration:  1100 / 14000 [  7%]  (Warmup)
## Chain 4 Iteration:  1200 / 14000 [  8%]  (Warmup)
## Chain 4 Iteration:  1300 / 14000 [  9%]  (Warmup)
## Chain 4 Iteration:  1400 / 14000 [ 10%]  (Warmup)
## Chain 4 Iteration:  1500 / 14000 [ 10%]  (Warmup)
## Chain 1 Iteration:     1 / 14000 [  0%]  (Warmup)
## Chain 1 Iteration:   100 / 14000 [  0%]  (Warmup)
## Chain 1 Iteration:   200 / 14000 [  1%]  (Warmup)
## Chain 1 Iteration:   300 / 14000 [  2%]  (Warmup)
## Chain 1 Iteration:   400 / 14000 [  2%]  (Warmup)
## Chain 1 Iteration:   500 / 14000 [  3%]  (Warmup)
## Chain 1 Iteration:   600 / 14000 [  4%]  (Warmup)
## Chain 1 Iteration:   700 / 14000 [  5%]  (Warmup)
## Chain 1 Iteration:   800 / 14000 [  5%]  (Warmup)
## Chain 1 Iteration:   900 / 14000 [  6%]  (Warmup)
## Chain 1 Iteration:  1000 / 14000 [  7%]  (Warmup)
## Chain 1 Iteration:  1100 / 14000 [  7%]  (Warmup)
## Chain 1 Iteration:  1200 / 14000 [  8%]  (Warmup)
## Chain 1 Iteration:  1300 / 14000 [  9%]  (Warmup)
## Chain 1 Iteration:  1400 / 14000 [ 10%]  (Warmup)
## Chain 1 Iteration:  1500 / 14000 [ 10%]  (Warmup)
## Chain 1 Iteration:  1600 / 14000 [ 11%]  (Warmup)
## Chain 1 Iteration:  1700 / 14000 [ 12%]  (Warmup)
## Chain 1 Iteration:  1800 / 14000 [ 12%]  (Warmup)
## Chain 1 Iteration:  1900 / 14000 [ 13%]  (Warmup)
## Chain 1 Iteration:  2000 / 14000 [ 14%]  (Warmup)
## Chain 1 Iteration:  2001 / 14000 [ 14%]  (Sampling)
## Chain 2 Iteration:     1 / 14000 [  0%]  (Warmup)
## Chain 2 Iteration:   100 / 14000 [  0%]  (Warmup)
## Chain 2 Iteration:   200 / 14000 [  1%]  (Warmup)
## Chain 2 Iteration:   300 / 14000 [  2%]  (Warmup)
## Chain 2 Iteration:   400 / 14000 [  2%]  (Warmup)
## Chain 2 Iteration:   500 / 14000 [  3%]  (Warmup)
## Chain 2 Iteration:   600 / 14000 [  4%]  (Warmup)
## Chain 2 Iteration:   700 / 14000 [  5%]  (Warmup)
## Chain 2 Iteration:   800 / 14000 [  5%]  (Warmup)
## Chain 2 Iteration:   900 / 14000 [  6%]  (Warmup)
## Chain 2 Iteration:  1000 / 14000 [  7%]  (Warmup)
## Chain 2 Iteration:  1100 / 14000 [  7%]  (Warmup)
## Chain 2 Iteration:  1200 / 14000 [  8%]  (Warmup)
## Chain 2 Iteration:  1300 / 14000 [  9%]  (Warmup)
## Chain 2 Iteration:  1400 / 14000 [ 10%]  (Warmup)
## Chain 2 Iteration:  1500 / 14000 [ 10%]  (Warmup)
```

```
## Chain 2 Iteration:  1600 / 14000 [ 11%]  (Warmup)
## Chain 2 Iteration:  1700 / 14000 [ 12%]  (Warmup)
## Chain 2 Iteration:  1800 / 14000 [ 12%]  (Warmup)
## Chain 2 Iteration:  1900 / 14000 [ 13%]  (Warmup)
## Chain 2 Iteration:  2000 / 14000 [ 14%]  (Warmup)
## Chain 2 Iteration:  2001 / 14000 [ 14%]  (Sampling)
## Chain 3 Iteration:  1100 / 14000 [  7%]  (Warmup)
## Chain 3 Iteration:  1200 / 14000 [  8%]  (Warmup)
## Chain 3 Iteration:  1300 / 14000 [  9%]  (Warmup)
## Chain 3 Iteration:  1400 / 14000 [ 10%]  (Warmup)
## Chain 3 Iteration:  1500 / 14000 [ 10%]  (Warmup)
## Chain 3 Iteration:  1600 / 14000 [ 11%]  (Warmup)
## Chain 3 Iteration:  1700 / 14000 [ 12%]  (Warmup)
## Chain 3 Iteration:  1800 / 14000 [ 12%]  (Warmup)
## Chain 3 Iteration:  1900 / 14000 [ 13%]  (Warmup)
## Chain 3 Iteration:  2000 / 14000 [ 14%]  (Warmup)
## Chain 3 Iteration:  2001 / 14000 [ 14%]  (Sampling)
## Chain 4 Iteration:  1600 / 14000 [ 11%]  (Warmup)
## Chain 4 Iteration:  1700 / 14000 [ 12%]  (Warmup)
## Chain 4 Iteration:  1800 / 14000 [ 12%]  (Warmup)
## Chain 4 Iteration:  1900 / 14000 [ 13%]  (Warmup)
## Chain 4 Iteration:  2000 / 14000 [ 14%]  (Warmup)
## Chain 4 Iteration:  2001 / 14000 [ 14%]  (Sampling)
## Chain 4 Iteration:  2100 / 14000 [ 15%]  (Sampling)
## Chain 4 Iteration:  2200 / 14000 [ 15%]  (Sampling)
## Chain 1 Iteration:  2100 / 14000 [ 15%]  (Sampling)
## Chain 1 Iteration:  2200 / 14000 [ 15%]  (Sampling)
## Chain 1 Iteration:  2300 / 14000 [ 16%]  (Sampling)
## Chain 1 Iteration:  2400 / 14000 [ 17%]  (Sampling)
## Chain 1 Iteration:  2500 / 14000 [ 17%]  (Sampling)
## Chain 1 Iteration:  2600 / 14000 [ 18%]  (Sampling)
## Chain 1 Iteration:  2700 / 14000 [ 19%]  (Sampling)
## Chain 1 Iteration:  2800 / 14000 [ 20%]  (Sampling)
## Chain 2 Iteration:  2100 / 14000 [ 15%]  (Sampling)
## Chain 2 Iteration:  2200 / 14000 [ 15%]  (Sampling)
## Chain 2 Iteration:  2300 / 14000 [ 16%]  (Sampling)
## Chain 2 Iteration:  2400 / 14000 [ 17%]  (Sampling)
## Chain 2 Iteration:  2500 / 14000 [ 17%]  (Sampling)
## Chain 2 Iteration:  2600 / 14000 [ 18%]  (Sampling)
## Chain 2 Iteration:  2700 / 14000 [ 19%]  (Sampling)
## Chain 2 Iteration:  2800 / 14000 [ 20%]  (Sampling)
## Chain 2 Iteration:  2900 / 14000 [ 20%]  (Sampling)
## Chain 3 Iteration:  2100 / 14000 [ 15%]  (Sampling)
## Chain 3 Iteration:  2200 / 14000 [ 15%]  (Sampling)
## Chain 3 Iteration:  2300 / 14000 [ 16%]  (Sampling)
## Chain 3 Iteration:  2400 / 14000 [ 17%]  (Sampling)
## Chain 3 Iteration:  2500 / 14000 [ 17%]  (Sampling)
## Chain 3 Iteration:  2600 / 14000 [ 18%]  (Sampling)
## Chain 4 Iteration:  2300 / 14000 [ 16%]  (Sampling)
## Chain 4 Iteration:  2400 / 14000 [ 17%]  (Sampling)
## Chain 4 Iteration:  2500 / 14000 [ 17%]  (Sampling)
## Chain 4 Iteration:  2600 / 14000 [ 18%]  (Sampling)
## Chain 4 Iteration:  2700 / 14000 [ 19%]  (Sampling)
## Chain 4 Iteration:  2800 / 14000 [ 20%]  (Sampling)
```

```
## Chain 4 Iteration:  2900 / 14000 [ 20%]  (Sampling)
## Chain 4 Iteration:  3000 / 14000 [ 21%]  (Sampling)
## Chain 4 Iteration:  3100 / 14000 [ 22%]  (Sampling)
## Chain 4 Iteration:  3200 / 14000 [ 22%]  (Sampling)
## Chain 4 Iteration:  3300 / 14000 [ 23%]  (Sampling)
## Chain 1 Iteration:  2900 / 14000 [ 20%]  (Sampling)
## Chain 1 Iteration:  3000 / 14000 [ 21%]  (Sampling)
## Chain 1 Iteration:  3100 / 14000 [ 22%]  (Sampling)
## Chain 1 Iteration:  3200 / 14000 [ 22%]  (Sampling)
## Chain 1 Iteration:  3300 / 14000 [ 23%]  (Sampling)
## Chain 2 Iteration:  3000 / 14000 [ 21%]  (Sampling)
## Chain 2 Iteration:  3100 / 14000 [ 22%]  (Sampling)
## Chain 2 Iteration:  3200 / 14000 [ 22%]  (Sampling)
## Chain 3 Iteration:  2700 / 14000 [ 19%]  (Sampling)
## Chain 3 Iteration:  2800 / 14000 [ 20%]  (Sampling)
## Chain 3 Iteration:  2900 / 14000 [ 20%]  (Sampling)
## Chain 3 Iteration:  3000 / 14000 [ 21%]  (Sampling)
## Chain 4 Iteration:  3400 / 14000 [ 24%]  (Sampling)
## Chain 4 Iteration:  3500 / 14000 [ 25%]  (Sampling)
## Chain 4 Iteration:  3600 / 14000 [ 25%]  (Sampling)
## Chain 4 Iteration:  3700 / 14000 [ 26%]  (Sampling)
## Chain 1 Iteration:  3400 / 14000 [ 24%]  (Sampling)
## Chain 1 Iteration:  3500 / 14000 [ 25%]  (Sampling)
## Chain 2 Iteration:  3300 / 14000 [ 23%]  (Sampling)
## Chain 2 Iteration:  3400 / 14000 [ 24%]  (Sampling)
## Chain 2 Iteration:  3500 / 14000 [ 25%]  (Sampling)
## Chain 3 Iteration:  3100 / 14000 [ 22%]  (Sampling)
## Chain 3 Iteration:  3200 / 14000 [ 22%]  (Sampling)
## Chain 3 Iteration:  3300 / 14000 [ 23%]  (Sampling)
## Chain 4 Iteration:  3800 / 14000 [ 27%]  (Sampling)
## Chain 4 Iteration:  3900 / 14000 [ 27%]  (Sampling)
## Chain 4 Iteration:  4000 / 14000 [ 28%]  (Sampling)
## Chain 1 Iteration:  3600 / 14000 [ 25%]  (Sampling)
## Chain 1 Iteration:  3700 / 14000 [ 26%]  (Sampling)
## Chain 1 Iteration:  3800 / 14000 [ 27%]  (Sampling)
## Chain 2 Iteration:  3600 / 14000 [ 25%]  (Sampling)
## Chain 3 Iteration:  3400 / 14000 [ 24%]  (Sampling)
## Chain 4 Iteration:  4100 / 14000 [ 29%]  (Sampling)
## Chain 4 Iteration:  4200 / 14000 [ 30%]  (Sampling)
## Chain 4 Iteration:  4300 / 14000 [ 30%]  (Sampling)
## Chain 1 Iteration:  3900 / 14000 [ 27%]  (Sampling)
## Chain 1 Iteration:  4000 / 14000 [ 28%]  (Sampling)
## Chain 1 Iteration:  4100 / 14000 [ 29%]  (Sampling)
## Chain 1 Iteration:  4200 / 14000 [ 30%]  (Sampling)
## Chain 2 Iteration:  3700 / 14000 [ 26%]  (Sampling)
## Chain 2 Iteration:  3800 / 14000 [ 27%]  (Sampling)
## Chain 2 Iteration:  3900 / 14000 [ 27%]  (Sampling)
## Chain 2 Iteration:  4000 / 14000 [ 28%]  (Sampling)
## Chain 3 Iteration:  3500 / 14000 [ 25%]  (Sampling)
## Chain 3 Iteration:  3600 / 14000 [ 25%]  (Sampling)
## Chain 3 Iteration:  3700 / 14000 [ 26%]  (Sampling)
## Chain 4 Iteration:  4400 / 14000 [ 31%]  (Sampling)
## Chain 4 Iteration:  4500 / 14000 [ 32%]  (Sampling)
## Chain 4 Iteration:  4600 / 14000 [ 32%]  (Sampling)
```

```
## Chain 4 Iteration:  4700 / 14000 [ 33%]  (Sampling)
## Chain 1 Iteration:  4300 / 14000 [ 30%]  (Sampling)
## Chain 1 Iteration:  4400 / 14000 [ 31%]  (Sampling)
## Chain 1 Iteration:  4500 / 14000 [ 32%]  (Sampling)
## Chain 2 Iteration:  4100 / 14000 [ 29%]  (Sampling)
## Chain 2 Iteration:  4200 / 14000 [ 30%]  (Sampling)
## Chain 2 Iteration:  4300 / 14000 [ 30%]  (Sampling)
## Chain 3 Iteration:  3800 / 14000 [ 27%]  (Sampling)
## Chain 3 Iteration:  3900 / 14000 [ 27%]  (Sampling)
## Chain 3 Iteration:  4000 / 14000 [ 28%]  (Sampling)
## Chain 3 Iteration:  4100 / 14000 [ 29%]  (Sampling)
## Chain 4 Iteration:  4800 / 14000 [ 34%]  (Sampling)
## Chain 4 Iteration:  4900 / 14000 [ 35%]  (Sampling)
## Chain 4 Iteration:  5000 / 14000 [ 35%]  (Sampling)
## Chain 1 Iteration:  4600 / 14000 [ 32%]  (Sampling)
## Chain 1 Iteration:  4700 / 14000 [ 33%]  (Sampling)
## Chain 1 Iteration:  4800 / 14000 [ 34%]  (Sampling)
## Chain 2 Iteration:  4400 / 14000 [ 31%]  (Sampling)
## Chain 2 Iteration:  4500 / 14000 [ 32%]  (Sampling)
## Chain 2 Iteration:  4600 / 14000 [ 32%]  (Sampling)
## Chain 3 Iteration:  4200 / 14000 [ 30%]  (Sampling)
## Chain 3 Iteration:  4300 / 14000 [ 30%]  (Sampling)
## Chain 4 Iteration:  5100 / 14000 [ 36%]  (Sampling)
## Chain 4 Iteration:  5200 / 14000 [ 37%]  (Sampling)
## Chain 4 Iteration:  5300 / 14000 [ 37%]  (Sampling)
## Chain 1 Iteration:  4900 / 14000 [ 35%]  (Sampling)
## Chain 1 Iteration:  5000 / 14000 [ 35%]  (Sampling)
## Chain 1 Iteration:  5100 / 14000 [ 36%]  (Sampling)
## Chain 2 Iteration:  4700 / 14000 [ 33%]  (Sampling)
## Chain 2 Iteration:  4800 / 14000 [ 34%]  (Sampling)
## Chain 2 Iteration:  4900 / 14000 [ 35%]  (Sampling)
## Chain 3 Iteration:  4400 / 14000 [ 31%]  (Sampling)
## Chain 3 Iteration:  4500 / 14000 [ 32%]  (Sampling)
## Chain 4 Iteration:  5400 / 14000 [ 38%]  (Sampling)
## Chain 4 Iteration:  5500 / 14000 [ 39%]  (Sampling)
## Chain 4 Iteration:  5600 / 14000 [ 40%]  (Sampling)
## Chain 1 Iteration:  5200 / 14000 [ 37%]  (Sampling)
## Chain 1 Iteration:  5300 / 14000 [ 37%]  (Sampling)
## Chain 2 Iteration:  5000 / 14000 [ 35%]  (Sampling)
## Chain 3 Iteration:  4600 / 14000 [ 32%]  (Sampling)
## Chain 3 Iteration:  4700 / 14000 [ 33%]  (Sampling)
## Chain 4 Iteration:  5700 / 14000 [ 40%]  (Sampling)
## Chain 4 Iteration:  5800 / 14000 [ 41%]  (Sampling)
## Chain 1 Iteration:  5400 / 14000 [ 38%]  (Sampling)
## Chain 1 Iteration:  5500 / 14000 [ 39%]  (Sampling)
## Chain 2 Iteration:  5100 / 14000 [ 36%]  (Sampling)
## Chain 2 Iteration:  5200 / 14000 [ 37%]  (Sampling)
## Chain 2 Iteration:  5300 / 14000 [ 37%]  (Sampling)
## Chain 3 Iteration:  4800 / 14000 [ 34%]  (Sampling)
## Chain 3 Iteration:  4900 / 14000 [ 35%]  (Sampling)
## Chain 4 Iteration:  5900 / 14000 [ 42%]  (Sampling)
## Chain 4 Iteration:  6000 / 14000 [ 42%]  (Sampling)
## Chain 4 Iteration:  6100 / 14000 [ 43%]  (Sampling)
## Chain 1 Iteration:  5600 / 14000 [ 40%]  (Sampling)
```

```
## Chain 1 Iteration:  5700 / 14000 [ 40%]  (Sampling)
## Chain 1 Iteration:  5800 / 14000 [ 41%]  (Sampling)
## Chain 2 Iteration:  5400 / 14000 [ 38%]  (Sampling)
## Chain 2 Iteration:  5500 / 14000 [ 39%]  (Sampling)
## Chain 3 Iteration:  5000 / 14000 [ 35%]  (Sampling)
## Chain 3 Iteration:  5100 / 14000 [ 36%]  (Sampling)
## Chain 3 Iteration:  5200 / 14000 [ 37%]  (Sampling)
## Chain 4 Iteration:  6200 / 14000 [ 44%]  (Sampling)
## Chain 4 Iteration:  6300 / 14000 [ 45%]  (Sampling)
## Chain 4 Iteration:  6400 / 14000 [ 45%]  (Sampling)
## Chain 4 Iteration:  6500 / 14000 [ 46%]  (Sampling)
## Chain 1 Iteration:  5900 / 14000 [ 42%]  (Sampling)
## Chain 1 Iteration:  6000 / 14000 [ 42%]  (Sampling)
## Chain 1 Iteration:  6100 / 14000 [ 43%]  (Sampling)
## Chain 2 Iteration:  5600 / 14000 [ 40%]  (Sampling)
## Chain 2 Iteration:  5700 / 14000 [ 40%]  (Sampling)
## Chain 2 Iteration:  5800 / 14000 [ 41%]  (Sampling)
## Chain 3 Iteration:  5300 / 14000 [ 37%]  (Sampling)
## Chain 3 Iteration:  5400 / 14000 [ 38%]  (Sampling)
## Chain 3 Iteration:  5500 / 14000 [ 39%]  (Sampling)
## Chain 4 Iteration:  6600 / 14000 [ 47%]  (Sampling)
## Chain 4 Iteration:  6700 / 14000 [ 47%]  (Sampling)
## Chain 4 Iteration:  6800 / 14000 [ 48%]  (Sampling)
## Chain 1 Iteration:  6200 / 14000 [ 44%]  (Sampling)
## Chain 1 Iteration:  6300 / 14000 [ 45%]  (Sampling)
## Chain 1 Iteration:  6400 / 14000 [ 45%]  (Sampling)
## Chain 2 Iteration:  5900 / 14000 [ 42%]  (Sampling)
## Chain 2 Iteration:  6000 / 14000 [ 42%]  (Sampling)
## Chain 2 Iteration:  6100 / 14000 [ 43%]  (Sampling)
## Chain 3 Iteration:  5600 / 14000 [ 40%]  (Sampling)
## Chain 3 Iteration:  5700 / 14000 [ 40%]  (Sampling)
## Chain 4 Iteration:  6900 / 14000 [ 49%]  (Sampling)
## Chain 4 Iteration:  7000 / 14000 [ 50%]  (Sampling)
## Chain 4 Iteration:  7100 / 14000 [ 50%]  (Sampling)
## Chain 1 Iteration:  6500 / 14000 [ 46%]  (Sampling)
## Chain 1 Iteration:  6600 / 14000 [ 47%]  (Sampling)
## Chain 1 Iteration:  6700 / 14000 [ 47%]  (Sampling)
## Chain 2 Iteration:  6200 / 14000 [ 44%]  (Sampling)
## Chain 2 Iteration:  6300 / 14000 [ 45%]  (Sampling)
## Chain 2 Iteration:  6400 / 14000 [ 45%]  (Sampling)
## Chain 3 Iteration:  5800 / 14000 [ 41%]  (Sampling)
## Chain 3 Iteration:  5900 / 14000 [ 42%]  (Sampling)
## Chain 3 Iteration:  6000 / 14000 [ 42%]  (Sampling)
## Chain 4 Iteration:  7200 / 14000 [ 51%]  (Sampling)
## Chain 4 Iteration:  7300 / 14000 [ 52%]  (Sampling)
## Chain 4 Iteration:  7400 / 14000 [ 52%]  (Sampling)
## Chain 1 Iteration:  6800 / 14000 [ 48%]  (Sampling)
## Chain 1 Iteration:  6900 / 14000 [ 49%]  (Sampling)
## Chain 1 Iteration:  7000 / 14000 [ 50%]  (Sampling)
## Chain 1 Iteration:  7100 / 14000 [ 50%]  (Sampling)
## Chain 2 Iteration:  6500 / 14000 [ 46%]  (Sampling)
## Chain 2 Iteration:  6600 / 14000 [ 47%]  (Sampling)
## Chain 2 Iteration:  6700 / 14000 [ 47%]  (Sampling)
## Chain 3 Iteration:  6100 / 14000 [ 43%]  (Sampling)
```

```
## Chain 3 Iteration:  6200 / 14000 [ 44%]  (Sampling)
## Chain 4 Iteration:  7500 / 14000 [ 53%]  (Sampling)
## Chain 4 Iteration:  7600 / 14000 [ 54%]  (Sampling)
## Chain 4 Iteration:  7700 / 14000 [ 55%]  (Sampling)
## Chain 4 Iteration:  7800 / 14000 [ 55%]  (Sampling)
## Chain 1 Iteration:  7200 / 14000 [ 51%]  (Sampling)
## Chain 1 Iteration:  7300 / 14000 [ 52%]  (Sampling)
## Chain 1 Iteration:  7400 / 14000 [ 52%]  (Sampling)
## Chain 2 Iteration:  6800 / 14000 [ 48%]  (Sampling)
## Chain 2 Iteration:  6900 / 14000 [ 49%]  (Sampling)
## Chain 2 Iteration:  7000 / 14000 [ 50%]  (Sampling)
## Chain 2 Iteration:  7100 / 14000 [ 50%]  (Sampling)
## Chain 3 Iteration:  6300 / 14000 [ 45%]  (Sampling)
## Chain 3 Iteration:  6400 / 14000 [ 45%]  (Sampling)
## Chain 3 Iteration:  6500 / 14000 [ 46%]  (Sampling)
## Chain 3 Iteration:  6600 / 14000 [ 47%]  (Sampling)
## Chain 4 Iteration:  7900 / 14000 [ 56%]  (Sampling)
## Chain 4 Iteration:  8000 / 14000 [ 57%]  (Sampling)
## Chain 4 Iteration:  8100 / 14000 [ 57%]  (Sampling)
## Chain 1 Iteration:  7500 / 14000 [ 53%]  (Sampling)
## Chain 1 Iteration:  7600 / 14000 [ 54%]  (Sampling)
## Chain 1 Iteration:  7700 / 14000 [ 55%]  (Sampling)
## Chain 2 Iteration:  7200 / 14000 [ 51%]  (Sampling)
## Chain 2 Iteration:  7300 / 14000 [ 52%]  (Sampling)
## Chain 2 Iteration:  7400 / 14000 [ 52%]  (Sampling)
## Chain 3 Iteration:  6700 / 14000 [ 47%]  (Sampling)
## Chain 3 Iteration:  6800 / 14000 [ 48%]  (Sampling)
## Chain 4 Iteration:  8200 / 14000 [ 58%]  (Sampling)
## Chain 4 Iteration:  8300 / 14000 [ 59%]  (Sampling)
## Chain 4 Iteration:  8400 / 14000 [ 60%]  (Sampling)
## Chain 4 Iteration:  8500 / 14000 [ 60%]  (Sampling)
## Chain 1 Iteration:  7800 / 14000 [ 55%]  (Sampling)
## Chain 1 Iteration:  7900 / 14000 [ 56%]  (Sampling)
## Chain 1 Iteration:  8000 / 14000 [ 57%]  (Sampling)
## Chain 2 Iteration:  7500 / 14000 [ 53%]  (Sampling)
## Chain 2 Iteration:  7600 / 14000 [ 54%]  (Sampling)
## Chain 3 Iteration:  6900 / 14000 [ 49%]  (Sampling)
## Chain 3 Iteration:  7000 / 14000 [ 50%]  (Sampling)
## Chain 4 Iteration:  8600 / 14000 [ 61%]  (Sampling)
## Chain 4 Iteration:  8700 / 14000 [ 62%]  (Sampling)
## Chain 4 Iteration:  8800 / 14000 [ 62%]  (Sampling)
## Chain 1 Iteration:  8100 / 14000 [ 57%]  (Sampling)
## Chain 1 Iteration:  8200 / 14000 [ 58%]  (Sampling)
## Chain 1 Iteration:  8300 / 14000 [ 59%]  (Sampling)
## Chain 2 Iteration:  7700 / 14000 [ 55%]  (Sampling)
## Chain 2 Iteration:  7800 / 14000 [ 55%]  (Sampling)
## Chain 3 Iteration:  7100 / 14000 [ 50%]  (Sampling)
## Chain 3 Iteration:  7200 / 14000 [ 51%]  (Sampling)
## Chain 4 Iteration:  8900 / 14000 [ 63%]  (Sampling)
## Chain 4 Iteration:  9000 / 14000 [ 64%]  (Sampling)
## Chain 4 Iteration:  9100 / 14000 [ 65%]  (Sampling)
## Chain 1 Iteration:  8400 / 14000 [ 60%]  (Sampling)
## Chain 1 Iteration:  8500 / 14000 [ 60%]  (Sampling)
## Chain 2 Iteration:  7900 / 14000 [ 56%]  (Sampling)
```

```
## Chain 2 Iteration:  8000 / 14000 [ 57%]  (Sampling)
## Chain 3 Iteration:  7300 / 14000 [ 52%]  (Sampling)
## Chain 3 Iteration:  7400 / 14000 [ 52%]  (Sampling)
## Chain 4 Iteration:  9200 / 14000 [ 65%]  (Sampling)
## Chain 4 Iteration:  9300 / 14000 [ 66%]  (Sampling)
## Chain 1 Iteration:  8600 / 14000 [ 61%]  (Sampling)
## Chain 1 Iteration:  8700 / 14000 [ 62%]  (Sampling)
## Chain 1 Iteration:  8800 / 14000 [ 62%]  (Sampling)
## Chain 2 Iteration:  8100 / 14000 [ 57%]  (Sampling)
## Chain 2 Iteration:  8200 / 14000 [ 58%]  (Sampling)
## Chain 2 Iteration:  8300 / 14000 [ 59%]  (Sampling)
## Chain 3 Iteration:  7500 / 14000 [ 53%]  (Sampling)
## Chain 3 Iteration:  7600 / 14000 [ 54%]  (Sampling)
## Chain 4 Iteration:  9400 / 14000 [ 67%]  (Sampling)
## Chain 4 Iteration:  9500 / 14000 [ 67%]  (Sampling)
## Chain 4 Iteration:  9600 / 14000 [ 68%]  (Sampling)
## Chain 1 Iteration:  8900 / 14000 [ 63%]  (Sampling)
## Chain 1 Iteration:  9000 / 14000 [ 64%]  (Sampling)
## Chain 1 Iteration:  9100 / 14000 [ 65%]  (Sampling)
## Chain 2 Iteration:  8400 / 14000 [ 60%]  (Sampling)
## Chain 2 Iteration:  8500 / 14000 [ 60%]  (Sampling)
## Chain 2 Iteration:  8600 / 14000 [ 61%]  (Sampling)
## Chain 3 Iteration:  7700 / 14000 [ 55%]  (Sampling)
## Chain 3 Iteration:  7800 / 14000 [ 55%]  (Sampling)
## Chain 4 Iteration:  9700 / 14000 [ 69%]  (Sampling)
## Chain 4 Iteration:  9800 / 14000 [ 70%]  (Sampling)
## Chain 4 Iteration:  9900 / 14000 [ 70%]  (Sampling)
## Chain 1 Iteration:  9200 / 14000 [ 65%]  (Sampling)
## Chain 1 Iteration:  9300 / 14000 [ 66%]  (Sampling)
## Chain 1 Iteration:  9400 / 14000 [ 67%]  (Sampling)
## Chain 2 Iteration:  8700 / 14000 [ 62%]  (Sampling)
## Chain 2 Iteration:  8800 / 14000 [ 62%]  (Sampling)
## Chain 3 Iteration:  7900 / 14000 [ 56%]  (Sampling)
## Chain 3 Iteration:  8000 / 14000 [ 57%]  (Sampling)
## Chain 3 Iteration:  8100 / 14000 [ 57%]  (Sampling)
## Chain 4 Iteration: 10000 / 14000 [ 71%]  (Sampling)
## Chain 4 Iteration: 10100 / 14000 [ 72%]  (Sampling)
## Chain 4 Iteration: 10200 / 14000 [ 72%]  (Sampling)
## Chain 1 Iteration:  9500 / 14000 [ 67%]  (Sampling)
## Chain 1 Iteration:  9600 / 14000 [ 68%]  (Sampling)
## Chain 2 Iteration:  8900 / 14000 [ 63%]  (Sampling)
## Chain 2 Iteration:  9000 / 14000 [ 64%]  (Sampling)
## Chain 3 Iteration:  8200 / 14000 [ 58%]  (Sampling)
## Chain 3 Iteration:  8300 / 14000 [ 59%]  (Sampling)
## Chain 3 Iteration:  8400 / 14000 [ 60%]  (Sampling)
## Chain 4 Iteration: 10300 / 14000 [ 73%]  (Sampling)
## Chain 4 Iteration: 10400 / 14000 [ 74%]  (Sampling)
## Chain 4 Iteration: 10500 / 14000 [ 75%]  (Sampling)
## Chain 1 Iteration:  9700 / 14000 [ 69%]  (Sampling)
## Chain 1 Iteration:  9800 / 14000 [ 70%]  (Sampling)
## Chain 1 Iteration:  9900 / 14000 [ 70%]  (Sampling)
## Chain 2 Iteration:  9100 / 14000 [ 65%]  (Sampling)
## Chain 2 Iteration:  9200 / 14000 [ 65%]  (Sampling)
## Chain 2 Iteration:  9300 / 14000 [ 66%]  (Sampling)
```

```
## Chain 3 Iteration:  8500 / 14000 [ 60%]  (Sampling)
## Chain 3 Iteration:  8600 / 14000 [ 61%]  (Sampling)
## Chain 4 Iteration: 10600 / 14000 [ 75%]  (Sampling)
## Chain 4 Iteration: 10700 / 14000 [ 76%]  (Sampling)
## Chain 4 Iteration: 10800 / 14000 [ 77%]  (Sampling)
## Chain 1 Iteration: 10000 / 14000 [ 71%]  (Sampling)
## Chain 1 Iteration: 10100 / 14000 [ 72%]  (Sampling)
## Chain 1 Iteration: 10200 / 14000 [ 72%]  (Sampling)
## Chain 2 Iteration:  9400 / 14000 [ 67%]  (Sampling)
## Chain 2 Iteration:  9500 / 14000 [ 67%]  (Sampling)
## Chain 2 Iteration:  9600 / 14000 [ 68%]  (Sampling)
## Chain 3 Iteration:  8700 / 14000 [ 62%]  (Sampling)
## Chain 3 Iteration:  8800 / 14000 [ 62%]  (Sampling)
## Chain 4 Iteration: 10900 / 14000 [ 77%]  (Sampling)
## Chain 4 Iteration: 11000 / 14000 [ 78%]  (Sampling)
## Chain 4 Iteration: 11100 / 14000 [ 79%]  (Sampling)
## Chain 4 Iteration: 11200 / 14000 [ 80%]  (Sampling)
## Chain 1 Iteration: 10300 / 14000 [ 73%]  (Sampling)
## Chain 1 Iteration: 10400 / 14000 [ 74%]  (Sampling)
## Chain 1 Iteration: 10500 / 14000 [ 75%]  (Sampling)
## Chain 2 Iteration:  9700 / 14000 [ 69%]  (Sampling)
## Chain 2 Iteration:  9800 / 14000 [ 70%]  (Sampling)
## Chain 2 Iteration:  9900 / 14000 [ 70%]  (Sampling)
## Chain 3 Iteration:  8900 / 14000 [ 63%]  (Sampling)
## Chain 3 Iteration:  9000 / 14000 [ 64%]  (Sampling)
## Chain 3 Iteration:  9100 / 14000 [ 65%]  (Sampling)
## Chain 4 Iteration: 11300 / 14000 [ 80%]  (Sampling)
## Chain 4 Iteration: 11400 / 14000 [ 81%]  (Sampling)
## Chain 4 Iteration: 11500 / 14000 [ 82%]  (Sampling)
## Chain 1 Iteration: 10600 / 14000 [ 75%]  (Sampling)
## Chain 1 Iteration: 10700 / 14000 [ 76%]  (Sampling)
## Chain 1 Iteration: 10800 / 14000 [ 77%]  (Sampling)
## Chain 1 Iteration: 10900 / 14000 [ 77%]  (Sampling)
## Chain 2 Iteration: 10000 / 14000 [ 71%]  (Sampling)
## Chain 2 Iteration: 10100 / 14000 [ 72%]  (Sampling)
## Chain 2 Iteration: 10200 / 14000 [ 72%]  (Sampling)
## Chain 3 Iteration:  9200 / 14000 [ 65%]  (Sampling)
## Chain 3 Iteration:  9300 / 14000 [ 66%]  (Sampling)
## Chain 3 Iteration:  9400 / 14000 [ 67%]  (Sampling)
## Chain 4 Iteration: 11600 / 14000 [ 82%]  (Sampling)
## Chain 4 Iteration: 11700 / 14000 [ 83%]  (Sampling)
## Chain 4 Iteration: 11800 / 14000 [ 84%]  (Sampling)
## Chain 4 Iteration: 11900 / 14000 [ 85%]  (Sampling)
## Chain 1 Iteration: 11000 / 14000 [ 78%]  (Sampling)
## Chain 1 Iteration: 11100 / 14000 [ 79%]  (Sampling)
## Chain 1 Iteration: 11200 / 14000 [ 80%]  (Sampling)
## Chain 1 Iteration: 11300 / 14000 [ 80%]  (Sampling)
## Chain 2 Iteration: 10300 / 14000 [ 73%]  (Sampling)
## Chain 2 Iteration: 10400 / 14000 [ 74%]  (Sampling)
## Chain 2 Iteration: 10500 / 14000 [ 75%]  (Sampling)
## Chain 3 Iteration:  9500 / 14000 [ 67%]  (Sampling)
## Chain 3 Iteration:  9600 / 14000 [ 68%]  (Sampling)
## Chain 4 Iteration: 12000 / 14000 [ 85%]  (Sampling)
## Chain 4 Iteration: 12100 / 14000 [ 86%]  (Sampling)
```

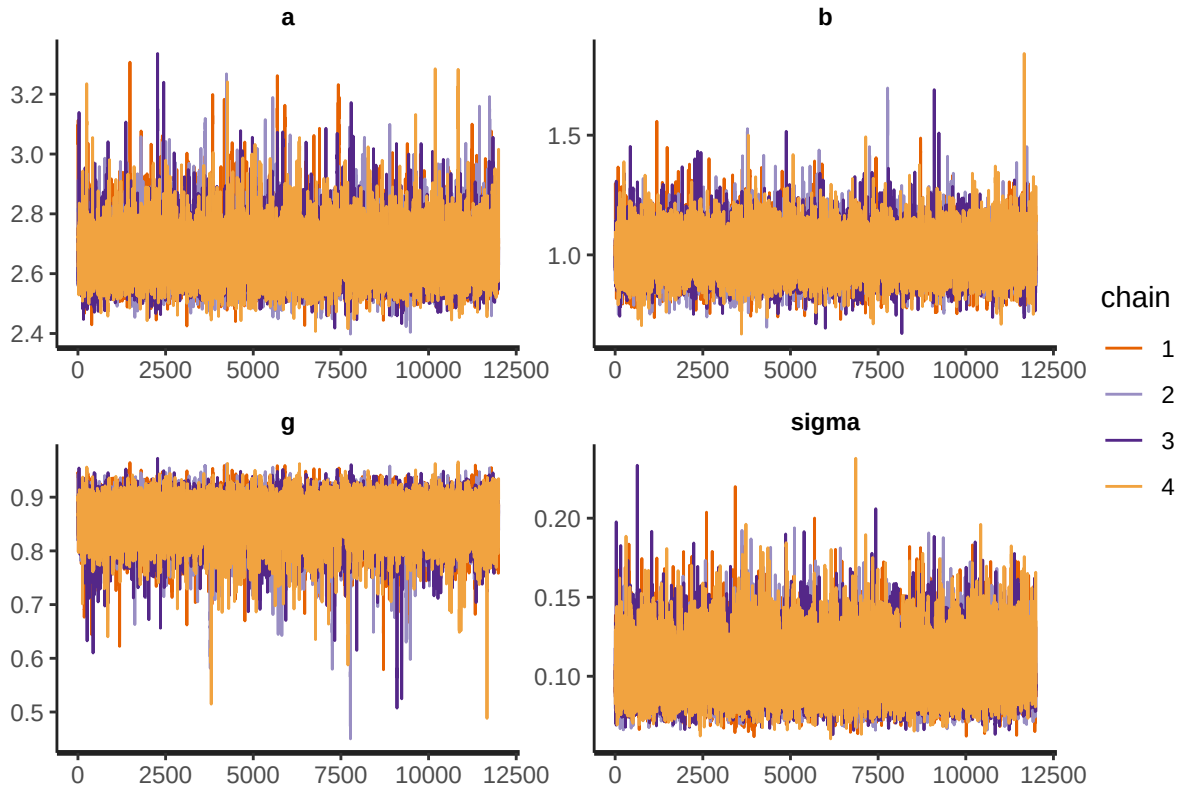
```
## Chain 4 Iteration: 12200 / 14000 [ 87%]  (Sampling)
## Chain 1 Iteration: 11400 / 14000 [ 81%]  (Sampling)
## Chain 1 Iteration: 11500 / 14000 [ 82%]  (Sampling)
## Chain 1 Iteration: 11600 / 14000 [ 82%]  (Sampling)
## Chain 2 Iteration: 10600 / 14000 [ 75%]  (Sampling)
## Chain 2 Iteration: 10700 / 14000 [ 76%]  (Sampling)
## Chain 2 Iteration: 10800 / 14000 [ 77%]  (Sampling)
## Chain 3 Iteration:  9700 / 14000 [ 69%]  (Sampling)
## Chain 3 Iteration:  9800 / 14000 [ 70%]  (Sampling)
## Chain 3 Iteration:  9900 / 14000 [ 70%]  (Sampling)
## Chain 4 Iteration: 12300 / 14000 [ 87%]  (Sampling)
## Chain 4 Iteration: 12400 / 14000 [ 88%]  (Sampling)
## Chain 4 Iteration: 12500 / 14000 [ 89%]  (Sampling)
## Chain 4 Iteration: 12600 / 14000 [ 90%]  (Sampling)
## Chain 1 Iteration: 11700 / 14000 [ 83%]  (Sampling)
## Chain 1 Iteration: 11800 / 14000 [ 84%]  (Sampling)
## Chain 1 Iteration: 11900 / 14000 [ 85%]  (Sampling)
## Chain 1 Iteration: 12000 / 14000 [ 85%]  (Sampling)
## Chain 2 Iteration: 10900 / 14000 [ 77%]  (Sampling)
## Chain 2 Iteration: 11000 / 14000 [ 78%]  (Sampling)
## Chain 2 Iteration: 11100 / 14000 [ 79%]  (Sampling)
## Chain 2 Iteration: 11200 / 14000 [ 80%]  (Sampling)
## Chain 3 Iteration: 10000 / 14000 [ 71%]  (Sampling)
## Chain 3 Iteration: 10100 / 14000 [ 72%]  (Sampling)
## Chain 3 Iteration: 10200 / 14000 [ 72%]  (Sampling)
## Chain 3 Iteration: 10300 / 14000 [ 73%]  (Sampling)
## Chain 4 Iteration: 12700 / 14000 [ 90%]  (Sampling)
## Chain 4 Iteration: 12800 / 14000 [ 91%]  (Sampling)
## Chain 4 Iteration: 12900 / 14000 [ 92%]  (Sampling)
## Chain 4 Iteration: 13000 / 14000 [ 92%]  (Sampling)
## Chain 1 Iteration: 12100 / 14000 [ 86%]  (Sampling)
## Chain 1 Iteration: 12200 / 14000 [ 87%]  (Sampling)
## Chain 1 Iteration: 12300 / 14000 [ 87%]  (Sampling)
## Chain 1 Iteration: 12400 / 14000 [ 88%]  (Sampling)
## Chain 2 Iteration: 11300 / 14000 [ 80%]  (Sampling)
## Chain 2 Iteration: 11400 / 14000 [ 81%]  (Sampling)
## Chain 2 Iteration: 11500 / 14000 [ 82%]  (Sampling)
## Chain 2 Iteration: 11600 / 14000 [ 82%]  (Sampling)
## Chain 3 Iteration: 10400 / 14000 [ 74%]  (Sampling)
## Chain 3 Iteration: 10500 / 14000 [ 75%]  (Sampling)
## Chain 3 Iteration: 10600 / 14000 [ 75%]  (Sampling)
## Chain 3 Iteration: 10700 / 14000 [ 76%]  (Sampling)
## Chain 4 Iteration: 13100 / 14000 [ 93%]  (Sampling)
## Chain 4 Iteration: 13200 / 14000 [ 94%]  (Sampling)
## Chain 4 Iteration: 13300 / 14000 [ 95%]  (Sampling)
## Chain 4 Iteration: 13400 / 14000 [ 95%]  (Sampling)
## Chain 4 Iteration: 13500 / 14000 [ 96%]  (Sampling)
## Chain 1 Iteration: 12500 / 14000 [ 89%]  (Sampling)
## Chain 1 Iteration: 12600 / 14000 [ 90%]  (Sampling)
## Chain 1 Iteration: 12700 / 14000 [ 90%]  (Sampling)
## Chain 1 Iteration: 12800 / 14000 [ 91%]  (Sampling)
## Chain 2 Iteration: 11700 / 14000 [ 83%]  (Sampling)
## Chain 2 Iteration: 11800 / 14000 [ 84%]  (Sampling)
## Chain 2 Iteration: 11900 / 14000 [ 85%]  (Sampling)
```

```
## Chain 3 Iteration: 10800 / 14000 [ 77%]  (Sampling)
## Chain 3 Iteration: 10900 / 14000 [ 77%]  (Sampling)
## Chain 3 Iteration: 11000 / 14000 [ 78%]  (Sampling)
## Chain 4 Iteration: 13600 / 14000 [ 97%]  (Sampling)
## Chain 4 Iteration: 13700 / 14000 [ 97%]  (Sampling)
## Chain 4 Iteration: 13800 / 14000 [ 98%]  (Sampling)
## Chain 1 Iteration: 12900 / 14000 [ 92%]  (Sampling)
## Chain 1 Iteration: 13000 / 14000 [ 92%]  (Sampling)
## Chain 1 Iteration: 13100 / 14000 [ 93%]  (Sampling)
## Chain 1 Iteration: 13200 / 14000 [ 94%]  (Sampling)
## Chain 2 Iteration: 12000 / 14000 [ 85%]  (Sampling)
## Chain 2 Iteration: 12100 / 14000 [ 86%]  (Sampling)
## Chain 2 Iteration: 12200 / 14000 [ 87%]  (Sampling)
## Chain 2 Iteration: 12300 / 14000 [ 87%]  (Sampling)
## Chain 3 Iteration: 11100 / 14000 [ 79%]  (Sampling)
## Chain 3 Iteration: 11200 / 14000 [ 80%]  (Sampling)
## Chain 3 Iteration: 11300 / 14000 [ 80%]  (Sampling)
## Chain 4 Iteration: 13900 / 14000 [ 99%]  (Sampling)
## Chain 4 Iteration: 14000 / 14000 [100%]  (Sampling)
## Chain 4 finished in 5.7 seconds.
## Chain 1 Iteration: 13300 / 14000 [ 95%]  (Sampling)
## Chain 1 Iteration: 13400 / 14000 [ 95%]  (Sampling)
## Chain 2 Iteration: 12400 / 14000 [ 88%]  (Sampling)
## Chain 2 Iteration: 12500 / 14000 [ 89%]  (Sampling)
## Chain 3 Iteration: 11400 / 14000 [ 81%]  (Sampling)
## Chain 3 Iteration: 11500 / 14000 [ 82%]  (Sampling)
## Chain 1 Iteration: 13500 / 14000 [ 96%]  (Sampling)
## Chain 1 Iteration: 13600 / 14000 [ 97%]  (Sampling)
## Chain 1 Iteration: 13700 / 14000 [ 97%]  (Sampling)
## Chain 1 Iteration: 13800 / 14000 [ 98%]  (Sampling)
## Chain 2 Iteration: 12600 / 14000 [ 90%]  (Sampling)
## Chain 2 Iteration: 12700 / 14000 [ 90%]  (Sampling)
## Chain 2 Iteration: 12800 / 14000 [ 91%]  (Sampling)
## Chain 3 Iteration: 11600 / 14000 [ 82%]  (Sampling)
## Chain 3 Iteration: 11700 / 14000 [ 83%]  (Sampling)
## Chain 3 Iteration: 11800 / 14000 [ 84%]  (Sampling)
## Chain 1 Iteration: 13900 / 14000 [ 99%]  (Sampling)
## Chain 1 Iteration: 14000 / 14000 [100%]  (Sampling)
## Chain 2 Iteration: 12900 / 14000 [ 92%]  (Sampling)
## Chain 2 Iteration: 13000 / 14000 [ 92%]  (Sampling)
## Chain 2 Iteration: 13100 / 14000 [ 93%]  (Sampling)
## Chain 3 Iteration: 11900 / 14000 [ 85%]  (Sampling)
## Chain 3 Iteration: 12000 / 14000 [ 85%]  (Sampling)
## Chain 3 Iteration: 12100 / 14000 [ 86%]  (Sampling)
## Chain 1 finished in 5.9 seconds.
## Chain 2 Iteration: 13200 / 14000 [ 94%]  (Sampling)
## Chain 2 Iteration: 13300 / 14000 [ 95%]  (Sampling)
## Chain 2 Iteration: 13400 / 14000 [ 95%]  (Sampling)
## Chain 3 Iteration: 12200 / 14000 [ 87%]  (Sampling)
## Chain 3 Iteration: 12300 / 14000 [ 87%]  (Sampling)
## Chain 3 Iteration: 12400 / 14000 [ 88%]  (Sampling)
## Chain 2 Iteration: 13500 / 14000 [ 96%]  (Sampling)
## Chain 2 Iteration: 13600 / 14000 [ 97%]  (Sampling)
## Chain 2 Iteration: 13700 / 14000 [ 97%]  (Sampling)
```

```
## Chain 3 Iteration: 12500 / 14000 [ 89%]  (Sampling)
## Chain 3 Iteration: 12600 / 14000 [ 90%]  (Sampling)
## Chain 3 Iteration: 12700 / 14000 [ 90%]  (Sampling)
## Chain 2 Iteration: 13800 / 14000 [ 98%]  (Sampling)
## Chain 2 Iteration: 13900 / 14000 [ 99%]  (Sampling)
## Chain 2 Iteration: 14000 / 14000 [100%]  (Sampling)
## Chain 3 Iteration: 12800 / 14000 [ 91%]  (Sampling)
## Chain 3 Iteration: 12900 / 14000 [ 92%]  (Sampling)
## Chain 3 Iteration: 13000 / 14000 [ 92%]  (Sampling)
## Chain 3 Iteration: 13100 / 14000 [ 93%]  (Sampling)
## Chain 2 finished in 6.4 seconds.
## Chain 3 Iteration: 13200 / 14000 [ 94%]  (Sampling)
## Chain 3 Iteration: 13300 / 14000 [ 95%]  (Sampling)
## Chain 3 Iteration: 13400 / 14000 [ 95%]  (Sampling)
## Chain 3 Iteration: 13500 / 14000 [ 96%]  (Sampling)
## Chain 3 Iteration: 13600 / 14000 [ 97%]  (Sampling)
## Chain 3 Iteration: 13700 / 14000 [ 97%]  (Sampling)
## Chain 3 Iteration: 13800 / 14000 [ 98%]  (Sampling)
## Chain 3 Iteration: 13900 / 14000 [ 99%]  (Sampling)
## Chain 3 Iteration: 14000 / 14000 [100%]  (Sampling)
## Chain 3 finished in 6.6 seconds.
##
## All 4 chains finished successfully.
## Mean chain execution time: 6.1 seconds.
## Total execution time: 7.3 seconds.

## # A tibble: 5 x 10
##   variable   mean median     sd    mad      q5    q95  rhat ess_bulk ess_tail
##   <chr>     <dbl>  <dbl>  <dbl>  <dbl>   <dbl>  <dbl> <dbl>    <dbl>    <dbl>
## 1 lp__     40.3   40.7   1.59   1.36   37.2    42.2    1.00   13987.   17560.
## 2 a         2.67   2.66  0.0841 0.0741  2.55    2.82   1.00   13759.   11269.
## 3 b         1.01   1.00  0.0842 0.0781  0.877   1.15   1.00   18602.   18475.
## 4 g         0.863  0.867 0.0368 0.0319  0.800   0.914  1.00   14404.   13412.
## 5 sigma     0.104  0.102 0.0168 0.0156  0.0804  0.135  1.00   21901.   22257.
```

Using the NUTS sampler provided with Stan, we have the following traceplots:



It appears to be doing pretty well. Lets compare the variables:

```
## # A tibble: 4 x 3
##   variable `2.5%` `97.5%`
##   <chr>     <dbl>   <dbl>
## 1 a         2.53    2.86
## 2 b         0.853   1.18
## 3 g         0.780   0.922
## 4 sigma     0.0772  0.142
```

```
##         lower  upper
## sigma 0.07501 0.1384
## a     2.51200 2.8320
## b     0.84360 1.1700
## g     0.78890 0.9267
## attr(,"Probability")
## [1] 0.95
```

Appendix

```
knitr::opts_chunk$set(echo = FALSE, fig.align = 'center', message = FALSE, eval = FALSE)
library(rstan)
library(R2jags)
```

```
library(cmdstanr)
library(loo)

dugong <- read.table("dugong.dat", header=TRUE)
dugong

plot(dugong, ylim = c(0,2.8))
model1 <- "
model {
for (i in 1:27) {
  y[i] ~ dnorm(mu[i], 1/s2error)
  mu[i] = a - b*g^(x[i])
}
  a ~ dnorm(3, 0.001)
  b ~ dgamma(6, 4)
  g ~ dbeta(5.5, 1.5)
  s2error ~ dgamma(1.1, 0.05)
}
"

writeLines(model1, 'model1.txt')

y <- dugong$length
x <- dugong$age

data.jags <- c('y', 'x')
parms <- c('a', 'b', 'g', 's2error')

dugong.sim <- jags(model.file = 'model1.txt', data = data.jags, parameters.to.save = parms,
                   n.iter = 50000, n.burnin = 2000, n.chains = 4, n.thin = 1, inits = NULL)

dugong.sim
sims <- as.mcmc(dugong.sim)
gelman.diag(sims)
chains <- as.matrix(sims)
sims <- as.mcmc(chains)
raftery.diag(sims)
effectiveSize(sims)
autocorr.diag(sims)
geweke.diag(sims)
HPDinterval(sims, prob = 0.95)
library(dplyr)
quantile(sims[,1], c(0.025, 0.975))
quantile(sims[,2], c(0.025, 0.975))
quantile(sims[,4], c(0.025, 0.975))
quantile(sims[,5], c(0.025, 0.975))
mod <- cmdstan_model(stan_file='hw9.stan')

data_list <- list(x = x,y = y, N = length(dugong$age))
fit <- mod$sample(data_list,
                  chains = 4,
                  parallel_chains = 4,
                  sig_figs = 4,
```

```
                  iter_sampling = 12000,
                  iter_warmup = 2000)

fit$summary(variables = c('lp__','a', 'b', 'g', 'sigma'))
rstan::traceplot(read_stan_csv(fit$output_files()), pars = c('a', 'b', 'g', 'sigma'))
fit$summary(variables = c('a', 'b', 'g', 'sigma'), ~quantile(., probs = c(0.025, 0.975)))

stanfit <- rstan::read_stan_csv(fit$output_files())
samps <- extract(stanfit)
chains <- cbind(samps[[1]], samps[[2]], samps[[3]], samps[[4]])
colnames(chains) <- names(samps[1:4])
sims <- as.mcmc(chains)
HPDinterval(sims, prob = 0.95)
```