BDA - Assignment 7

Anonymous

Contents

```
# To install aaltobda, see the General information in the assignment.
remotes::install_github("avehtari/BDA_course_Aalto", subdir = "rpackage", upgrade = "never")
```

```
## Skipping install of 'aaltobda' from a github remote, the SHA1 (38f34d35) has not changed since last
##   Use `force = TRUE` to force installation
```

```
library(aaltobda)
library(rstan)
```

```
## Loading required package: StanHeaders
```

```
## Loading required package: ggplot2
```

```
## rstan (Version 2.21.3, GitRev: 2e1f913d3ca3)
```

```
## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)
```

```
## Do not specify '-march=native' in 'LOCAL_CPPFLAGS' or a Makevars file
```

1 Linear model: Drowing data with Stan

```
data("drowning")
```

a) Find the three mistakes

Below I marked the three errors within the Stan script.

```
writeLines(readLines("listing1.stan"))
```

```
## Warning in readLines("listing1.stan"): incomplete final line found on
## 'listing1.stan'
```

```
## // The corrected
## data {
##     int<lower=0> N;     // number of observations
##     vector[N] x;        // observations per year
##     vector[N] y;        // observation number of drowned
##     real xpred;         // prediction year --- ERROR 3: See below
## }
## parameters {
##     real alpha;
##     real beta;
##     real<lower=0> sigma;    // ERROR 1: sigma must be > 0
## }
## transformed parameters {
##     vector[N] mu = alpha + beta*x;
## }
## model {
##     // Priors
##     beta ~ normal(0, 25);
##
##     // Likelihood function
##     y ~ normal(mu, sigma);   // ERROR 2: line not ending in ";"
## }
## generated quantities {
##    // ERROR 3: mu was not using xpred
##    real ypred = normal_rng(alpha + beta*xpred, sigma);
## }
```

b)

I chose $\beta \sim N(0, \sigma_{beta} = 25)$ since I show below that $Pr(-69 < \beta < 69)$ is alittle above .99.

```
integrate(function(beta) {dnorm (beta, 0, 25)}, -69, 69)
```

```
## 0.9942199 with absolute error < 1.1e-07
```

c)

Below I show how I added the priors, before the likelihood function. Notice that in my Stan script, I already have the priors added.

```
// Priors beta ~ normal(0, 25); alpha ~ normal(1980+143, 28)
```

d)

Because I have no information about the constant (e.g., what the value of drownings was when Jesus was borned), then I rather just keep it as uniform.

Extra:

I show if I obtain similar figures to those in the assignment.

```
# Creating the data
data <- list(N = length(drowning$year),
             x = drowning$year,
             y = drowning$drownings,
             xpred = 2020)
fit <- stan(file = 'listing1.stan', data = data, verbose = FALSE)
```

```
## Warning in readLines(file, warn = TRUE): incomplete
## final line found on 'C:\Users\noe.nava\OneDrive -
## USDA\Drive\courses\vehtari_bayesian_data_analysis\BDA_course_Aalto\navaAssignments\assignment
## 7\listing1.stan'
```

```
##
## SAMPLING FOR MODEL 'listing1' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 1.796 seconds (Warm-up)
## Chain 1:                2.407 seconds (Sampling)
## Chain 1:                4.203 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'listing1' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.948 seconds (Warm-up)
## Chain 2:                2.433 seconds (Sampling)
## Chain 2:                3.381 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'listing1' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 1.575 seconds (Warm-up)
## Chain 3:                2.383 seconds (Sampling)
## Chain 3:                3.958 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'listing1' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 1.375 seconds (Warm-up)
## Chain 4:                2.253 seconds (Sampling)
## Chain 4:                3.628 seconds (Total)
## Chain 4:
```

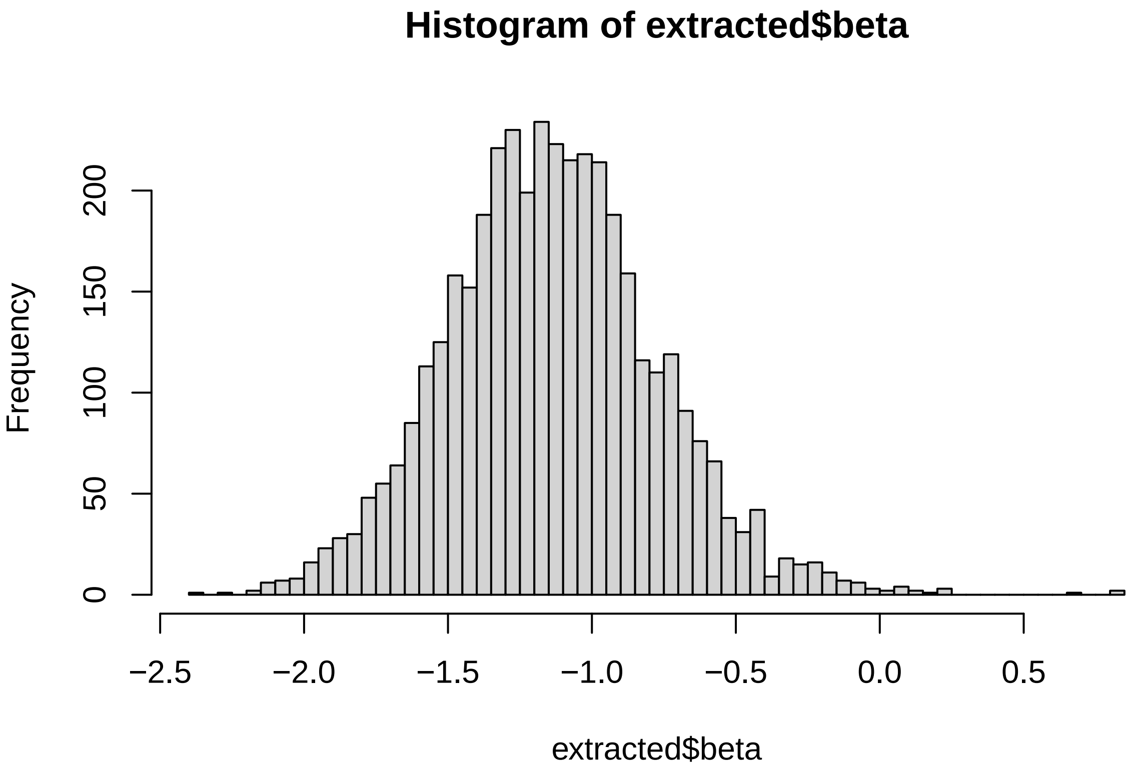
```
## Warning: There were 855 transitions after warmup that exceeded the maximum treedepth. Increase max_tr
## https://mc-stan.org/misc/warnings.html#maximum-treedepth-exceeded
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```
extracted <- extract(fit)
```

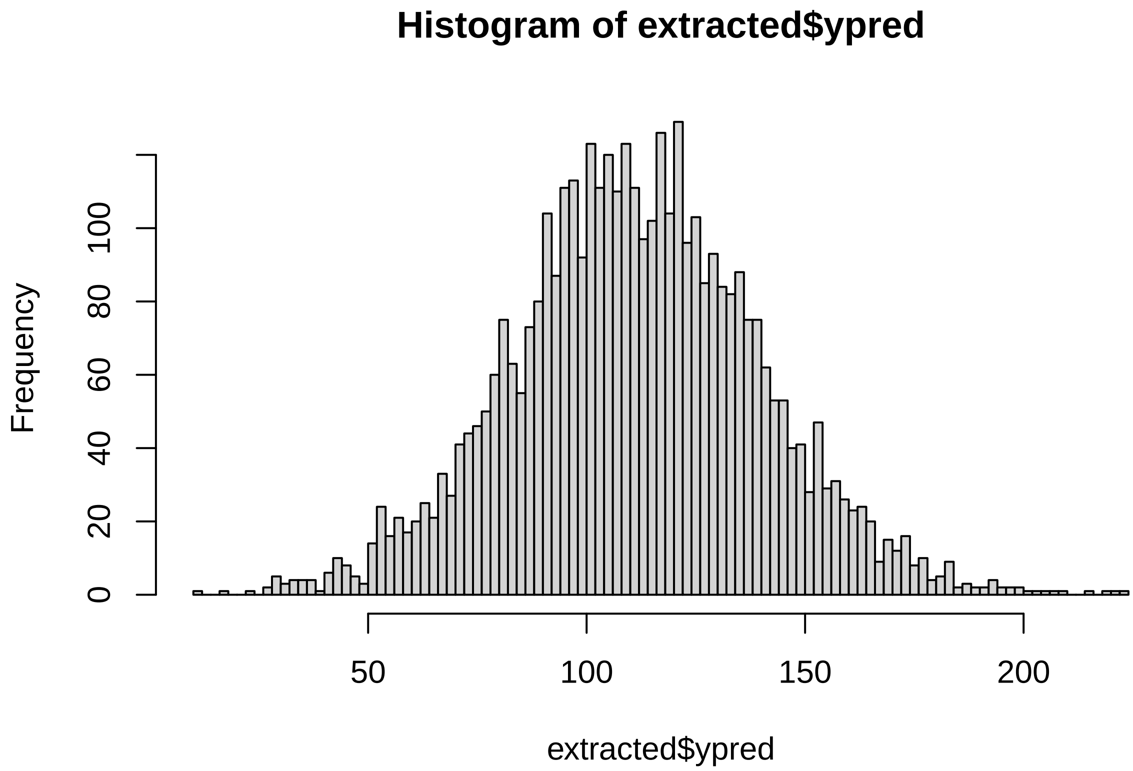
Below I show the histogram for beta:

```
hist(extracted$beta, breaks = 100)
```



Below I show the histogram for the value in 2020:

```
hist(extracted$ypred, breaks = 100)
```



2 Hierarchical model: factory data with Stan

```
data("factory")
ypool <- matrix(as.matrix(factory), ncol = 1)
ypool <- as.vector(ypool) # previous dimensions were 30x1, but Stan expects 30 ... do not know the diff
stan_data <- list(
  y = factory,         # Data in 5x6 dimension
  N = nrow(factory),   # Number of observations per machine
  J = ncol(factory),   # Number of machines
  ypool = ypool
)
test <- stan(file = 'test.stan', data = stan_data)
```

```
##
## SAMPLING FOR MODEL 'test' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.067 seconds (Warm-up)
## Chain 1:                0.053 seconds (Sampling)
## Chain 1:                0.12 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'test' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.064 seconds (Warm-up)
## Chain 2:                0.053 seconds (Sampling)
## Chain 2:                0.117 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'test' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.064 seconds (Warm-up)
## Chain 3:                0.054 seconds (Sampling)
## Chain 3:                0.118 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'test' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.064 seconds (Warm-up)
## Chain 4:                0.05 seconds (Sampling)
## Chain 4:                0.114 seconds (Total)
## Chain 4:
```

```
print(test)
```

```
## Inference for Stan model: test.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##             mean se_mean    sd    2.5%     25%     50%     75%   97.5% n_eff
## mu[1]       0.15    0.01  1.01   -1.85   -0.52    0.15    0.83    2.12  7885
## mu[2]       0.09    0.01  1.02   -1.85   -0.62    0.07    0.78    2.08  9064
## mu[3]       0.11    0.01  1.00   -1.87   -0.56    0.11    0.75    2.03  8120
## mu[4]       0.07    0.01  1.00   -1.96   -0.60    0.08    0.72    2.02  8199
## mu[5]       0.12    0.01  0.97   -1.73   -0.56    0.11    0.80    2.03  7763
## mu[6]       0.12    0.01  1.03   -1.97   -0.56    0.12    0.81    2.14  8476
## sigma[1]   60.06    0.22 15.47   38.28   49.33   57.33   67.44   97.15  5029
## sigma[2]   82.07    0.32 21.70   51.66   67.27   78.16   92.34  136.44  4618
## sigma[3]   67.78    0.24 17.05   43.72   56.28   64.77   75.68  109.23  5006
## sigma[4]   86.37    0.34 22.60   54.59   70.62   82.24   97.03  140.53  4536
## sigma[5]   69.30    0.26 17.43   44.89   56.98   65.89   77.90  112.49  4366
## sigma[6]   67.36    0.26 17.41   42.86   55.31   64.24   75.54  109.92  4531
## ypred       1.05    0.97 61.85 -121.63  -36.70   -0.26   39.71  128.70  4030
## lp__     -287.47    0.06  2.56 -293.49 -288.92 -287.17 -285.60 -283.53  1569
##          Rhat
## mu[1]       1
## mu[2]       1
## mu[3]       1
## mu[4]       1
## mu[5]       1
## mu[6]       1
## sigma[1]    1
## sigma[2]    1
## sigma[3]    1
## sigma[4]    1
## sigma[5]    1
## sigma[6]    1
## ypred       1
## lp__        1
##
## Samples were drawn using NUTS(diag_e) at Tue Mar 29 19:54:27 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
testExtracted <- extract(test)
```

a) Description of models:

- Separate model

Our machine observations are assumed to come from

$y_{ij} \sim N(\mu_j, \sigma_j)$

where the prior distribution of the parameters are

$\mu_j \sim N(0, 1)$

$\sigma_j \sim N(0, 1)$

Notice here that what is happening is that the modelling discerns between observations as they come from different machines.

- Pooled model

Our machine observations are assumed to come from a common *pooled* distribution as defined by

$y_i \sim N(\mu, \sigma)$

where in addition, the prior distribution of the parameters are

$\mu \sim N(0, 1)$

$\sigma \sim N(0, 1)$

Notice here that what is happening is that the modelling does not discern between observations as they come from different machines; they are pooled together.

- Hierarchical model

Our machine observations are assumed to come from

$y_{ij} \sim N(\mu_j, \sigma)$

where in addition, the prior distribution of the parameters are

$\mu_j \sim N(0, \tau)$

where τ $N(0,1)$ is our *hyper-parameters.* The distribution of the shape parameter is drawn as:

$\sigma \sim N(0,1)$

Notice that here, we are restricting our parameters of location mu_j to be drawn from a common distribution whose parameter of location is, in turn, drawn from another normal distribution, a weekly *hyper-prior*: $\tau \sim N(0,10)$

b) Stan scripts for each model

- Pooled model:

```r
pooled <- stan(file = 'pooled.stan', data = stan_data, verbose = FALSE)
```

```
##
## SAMPLING FOR MODEL 'pooled' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.015 seconds (Warm-up)
## Chain 1:                0.015 seconds (Sampling)
## Chain 1:                0.03 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'pooled' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.015 seconds (Warm-up)
## Chain 2:                0.014 seconds (Sampling)
## Chain 2:                0.029 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'pooled' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.016 seconds (Warm-up)
## Chain 3:                0.014 seconds (Sampling)
## Chain 3:                0.03 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'pooled' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
```

```
## Chain 4:  Elapsed Time: 0.017 seconds (Warm-up)
## Chain 4:                0.012 seconds (Sampling)
## Chain 4:                0.029 seconds (Total)
## Chain 4:
```

```
#print(pooled)
pooledExtracted <- extract(pooled)
writeLines(readLines("pooled.stan"))
```

```
## data {
##     int<lower=0> N; // Numnber of observations per machine
##     int<lower=0> J; // Number of machines
##     vector[J] y[N]; // This seems to create a matrix
##     vector[J*N] ypool; // pooled ys
## }
##
## parameters {
##     real mu;
##     real<lower=0> sigma;
## }
##
## model {
##     // Priors
##     mu ~ normal(0,1);
##     sigma ~ normal(0,10);
##
##     // Likelihood
##     ypool ~ normal(mu, sigma);
## }
##
## generated quantities {
##    real ypred6;
##    real ypred5;
##    real ypred7;
## // Compute predictive distribution
## // for the sixth machine
##    ypred6 = normal_rng(mu, sigma);
## // for the fifth machine
##    ypred5 = normal_rng(mu, sigma);
## // for the seventh machine
##    ypred7 = normal_rng(mu, sigma);
## }
```

- separate model:

```
separate <- stan(file = 'separate.stan', data = stan_data, verbose = FALSE)
```

```
##
## SAMPLING FOR MODEL 'separate' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.063 seconds (Warm-up)
## Chain 1:                0.048 seconds (Sampling)
## Chain 1:                0.111 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'separate' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.061 seconds (Warm-up)
## Chain 2:                0.047 seconds (Sampling)
## Chain 2:                0.108 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'separate' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.061 seconds (Warm-up)
## Chain 3:                0.048 seconds (Sampling)
## Chain 3:                0.109 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'separate' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.062 seconds (Warm-up)
## Chain 4:                0.045 seconds (Sampling)
## Chain 4:                0.107 seconds (Total)
## Chain 4:
```

```r
#print(separate)
separateExtracted <- extract(separate)
writeLines(readLines("separate.stan"))
```

```
## data {
##     int<lower=0> N; // Numnber of observations per machine
##     int<lower=0> J; // Number of machines
##     vector[J] y[N]; // This seems to create a matrix
##     vector[J*N] ypool; // pooled ys
## }
##
## parameters {
##     vector[J] mu;
##     vector<lower=0>[J] sigma;
## }
##
## model {
```

```
##     // Priors
##     for(j in 1:J) {
##         mu[j] ~ normal(0,1);
##         sigma[j] ~ normal(0,10);
##     }
##
##     // Likelihood
##     for(j in 1:J) {
##         y[,j] ~ normal(mu[j], sigma[j]);
##     }
## }
##
## generated quantities {
##    real ypred6;
##    real ypred5;
## // Compute predictive distribution
## // for the sixth machine
##    ypred6 = normal_rng(mu[6], sigma[6]);
## // for the fifth machine
##    ypred5 = normal_rng(mu[5], sigma[5]);
## }
```

- Hierarchical model

```
hierarchical <- stan(file = 'hierarchical.stan', data = stan_data, verbose = FALSE)
```

```
##
## SAMPLING FOR MODEL 'hierarchical' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.139 seconds (Warm-up)
## Chain 1:                0.102 seconds (Sampling)
## Chain 1:                0.241 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'hierarchical' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.119 seconds (Warm-up)
## Chain 2:                0.09 seconds (Sampling)
## Chain 2:                0.209 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'hierarchical' NOW (CHAIN 3).
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -0.346777, but must be > 0!  (in 'model600c7bf464
##
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -0.411272, but must be > 0!  (in 'model600c7bf464
##
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -1.76811, but must be > 0!  (in 'model600c7bf464a
##
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -1.4772, but must be > 0!  (in 'model600c7bf464aa
##
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.098 seconds (Warm-up)
## Chain 3:                0.09 seconds (Sampling)
## Chain 3:                0.188 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'hierarchical' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.131 seconds (Warm-up)
## Chain 4:                0.113 seconds (Sampling)
## Chain 4:                0.244 seconds (Total)
## Chain 4:
```

```
## Warning: There were 499 divergent transitions after warmup. See
## https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```
## Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#tail-ess
```

```
#print(hierarchical)
hierarchicalExtracted <- extract(hierarchical)
writeLines(readLines("hierarchical.stan"))
```

```
## data {
##     int<lower=0> N; // Numnber of observations per machine
##     int<lower=0> J; // Number of machines
##     vector[J] y[N]; // This seems to create a matrix
##     vector[J*N] ypool; // pooled ys
## }
##
## parameters {
##     vector[J] mu;
##     real<lower=0> sigma;
##     real tau; // Hyper-parameter
## }
##
## model {
## // Hyper-priors
##     tau ~ normal(0,1);
##
## // Priors
##     for(j in 1:J) {
##         mu[j] ~ normal(0,tau);
##     }
##     sigma ~ normal(0,10);
##
## // Likelihood
##     for(j in 1:J) {
##         y[,j] ~ normal(mu[j], sigma);
##     }
## }
##
## generated quantities {
##    real ypred6;
##    real ypred5;
##    // Compute predictive distribution
##    // for the sixth machine
##    ypred6 = normal_rng(mu[6], sigma);
##    // Compute predictive distribution
##    // for the fifth machine
##    ypred5 = normal_rng(mu[5], sigma);
## }
```

c)

- **the posterior distribution of the mean of the quality measurements of the sixth machine**
    - separate:
- the predictive distribution for another quality measurements of the sixth machine
- the posterior distrubtion of the mean of the quality of the seventh machine.

d)

First, we create the function to report the mean with the 90% true intervals:

```
estIntervals <- function(sims) {

  est <- mean(sims)
  low <- quantile(sims, .05)
  upp <- quantile(sims, .95)

  value <- list(
    est = est,
    low = low,
    upp = upp
  )

  return(value)
}
```

- Pooled model:

```
pooledv2 <- stan(file = 'pooledv2.stan', data = stan_data, verbose = FALSE)
```

```
##
## SAMPLING FOR MODEL 'pooledv2' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.016 seconds (Warm-up)
## Chain 1:                0.013 seconds (Sampling)
## Chain 1:                0.029 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'pooledv2' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.019 seconds (Warm-up)
## Chain 2:                0.015 seconds (Sampling)
## Chain 2:                0.034 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'pooledv2' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.017 seconds (Warm-up)
## Chain 3:                0.012 seconds (Sampling)
## Chain 3:                0.029 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'pooledv2' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.021 seconds (Warm-up)
## Chain 4:                0.012 seconds (Sampling)
## Chain 4:                0.033 seconds (Total)
## Chain 4:
```

```
print(pooledv2)
```

```
## Inference for Stan model: pooledv2.
## 4 chains, each with iter=2000; warmup=1000; thin=1; 
## post-warmup draws per chain=1000, total post-warmup draws=4000.
## 
##          mean se_mean   sd    2.5%     25%     50%     75%   97.5% n_eff Rhat
## mu      85.59    0.08 3.32   78.35   83.57   85.78   87.84   91.50  1613    1
## sigma   16.02    0.04 1.84   12.89   14.74   15.89   17.10   20.04  1784    1
## lp__  -155.55    0.03 1.07 -158.51 -155.97 -155.21 -154.80 -154.55  1508    1
## 
## Samples were drawn using NUTS(diag_e) at Tue Mar 29 19:56:26 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at 
## convergence, Rhat=1).
```

```
pooledv2Extracted <- extract(pooledv2)

estIntervals(pooledv2Extracted$mu)
```

```
## $est
## [1] 85.58823
## 
## $low
##       5% 
## 79.70592 
## 
## $upp
##     95% 
## 90.6226
```

- separate model:

```
separatev2 <- stan(file = 'separatev2.stan', data = stan_data, verbose = FALSE)
```

```
## 
## SAMPLING FOR MODEL 'separatev2' NOW (CHAIN 1).
## Chain 1: 
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1: 
## Chain 1: 
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.304 seconds (Warm-up)
## Chain 1:                0.152 seconds (Sampling)
## Chain 1:                0.456 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'separatev2' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.338 seconds (Warm-up)
## Chain 2:                0.105 seconds (Sampling)
## Chain 2:                0.443 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'separatev2' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.281 seconds (Warm-up)
## Chain 3:                0.127 seconds (Sampling)
## Chain 3:                0.408 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'separatev2' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.299 seconds (Warm-up)
## Chain 4:                0.143 seconds (Sampling)
## Chain 4:                0.442 seconds (Total)
## Chain 4:
```

```
print(separatev2)
```

```
## Inference for Stan model: separatev2.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##             mean se_mean    sd    2.5%     25%     50%     75%   97.5% n_eff
## mu[1]      50.69    0.16  8.96   31.83   44.78   51.36   57.16   66.06  3110
## mu[2]      48.92    0.20 11.93   26.35   40.61   48.64   57.01   72.68  3506
## mu[3]      58.26    0.22 12.07   32.47   50.13   59.11   67.25   78.95  3007
## mu[4]      47.38    0.20 11.62   25.07   39.59   47.17   55.10   70.95  3426
## mu[5]      60.50    0.22 12.73   34.43   51.94   61.06   69.82   82.70  3427
## mu[6]      51.59    0.19 10.74   29.52   44.45   52.11   59.18   70.65  3194
## sigma[1]   15.96    0.06  3.35   10.28   13.54   15.66   18.13   23.26  3400
## sigma[2]   24.63    0.08  4.49   15.89   21.61   24.55   27.54   33.71  3541
## sigma[3]   16.07    0.08  4.51    8.26   12.80   15.66   19.04   25.48  3112
## sigma[4]   26.54    0.07  4.35   18.17   23.58   26.50   29.41   34.98  3466
## sigma[5]   15.81    0.08  4.74    7.62   12.22   15.68   19.06   25.60  3148
## sigma[6]   18.52    0.07  3.99   11.72   15.67   18.17   21.18   26.80  3112
## lp__     -346.93    0.06  2.38 -352.50 -348.27 -346.62 -345.21 -343.23  1475
##          Rhat
## mu[1]       1
## mu[2]       1
## mu[3]       1
## mu[4]       1
## mu[5]       1
## mu[6]       1
## sigma[1]    1
## sigma[2]    1
## sigma[3]    1
## sigma[4]    1
## sigma[5]    1
## sigma[6]    1
## lp__        1
##
## Samples were drawn using NUTS(diag_e) at Tue Mar 29 19:56:57 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
separatev2Extracted <- extract(separatev2)

estIntervals(separatev2Extracted$mu[1])
```

```
## $est
## [1] 47.32908
##
## $low
##       5%
## 47.32908
##
## $upp
##      95%
## 47.32908
```

- Hierarchical model

```
hierarchicalv2 <- stan(file = 'hierarchicalv2.stan', data = stan_data, verbose = FALSE)
```

```
##
## SAMPLING FOR MODEL 'hierarchicalv2' NOW (CHAIN 1).
## Chain 1: Rejecting initial value:
## Chain 1:   Error evaluating the log probability at the initial value.
## Chain 1: Exception: normal_lpdf: Scale parameter is -0.640663, but must be > 0!  (in 'model600c1cf23
##
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.153 seconds (Warm-up)
## Chain 1:                0.045 seconds (Sampling)
## Chain 1:                0.198 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'hierarchicalv2' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.154 seconds (Warm-up)
## Chain 2:                0.047 seconds (Sampling)
## Chain 2:                0.201 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'hierarchicalv2' NOW (CHAIN 3).
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -0.255411, but must be > 0!  (in 'model600c1cf230
##
## Chain 3: Rejecting initial value:
## Chain 3:   Error evaluating the log probability at the initial value.
## Chain 3: Exception: normal_lpdf: Scale parameter is -1.96122, but must be > 0!  (in 'model600c1cf2309
##
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.14 seconds (Warm-up)
## Chain 3:                0.046 seconds (Sampling)
## Chain 3:                0.186 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'hierarchicalv2' NOW (CHAIN 4).
## Chain 4: Rejecting initial value:
## Chain 4:   Error evaluating the log probability at the initial value.
## Chain 4: Exception: normal_lpdf: Scale parameter is -1.68691, but must be > 0!  (in 'model600c1cf2309
##
## Chain 4: Rejecting initial value:
## Chain 4:   Error evaluating the log probability at the initial value.
## Chain 4: Exception: normal_lpdf: Scale parameter is -0.553764, but must be > 0!  (in 'model600c1cf230
##
## Chain 4: Rejecting initial value:
## Chain 4:   Error evaluating the log probability at the initial value.
## Chain 4: Exception: normal_lpdf: Scale parameter is -1.26253, but must be > 0!  (in 'model600c1cf2309
##
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.15 seconds (Warm-up)
## Chain 4:                0.046 seconds (Sampling)
## Chain 4:                0.196 seconds (Total)
## Chain 4:
```

```
print(hierarchical)
```

```
## Inference for Stan model: hierarchical.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##           mean se_mean    sd    2.5%     25%     50%     75%   97.5% n_eff Rhat
## mu[1]     0.13    0.02  1.14   -2.12   -0.38    0.06    0.58    2.84  3447 1.00
## mu[2]     0.18    0.02  1.20   -2.15   -0.39    0.08    0.63    2.93  2372 1.00
## mu[3]     0.17    0.02  1.22   -2.12   -0.40    0.04    0.60    3.09  3619 1.00
## mu[4]     0.17    0.02  1.23   -2.26   -0.38    0.07    0.65    3.07  2558 1.00
## mu[5]     0.19    0.02  1.21   -2.11   -0.39    0.10    0.65    3.11  2814 1.00
## mu[6]     0.15    0.02  1.24   -2.37   -0.40    0.05    0.59    3.02  4016 1.00
## sigma    62.65    0.13  4.18   55.17   59.70   62.28   65.32   71.46  1030 1.00
## tau       0.98    0.04  0.63    0.18    0.47    0.86    1.34    2.51   294 1.01
## ypred6   -0.26    1.00 62.54 -118.54  -42.94   -1.43   41.19  125.70  3936 1.00
## ypred5    0.06    1.00 62.88 -123.76  -42.06    0.69   42.19  123.03  3991 1.00
## lp__   -176.59    0.36  5.26 -186.98 -180.24 -176.82 -172.70 -166.69   211 1.01
##
## Samples were drawn using NUTS(diag_e) at Tue Mar 29 19:55:57 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
hierarchicalv2Extracted <- extract(hierarchicalv2)

estIntervals(hierarchicalv2Extracted$mu[1])
```

```
## $est
## [1] 71.16178
##
## $low
##       5%
## 71.16178
##
## $upp
##      95%
## 71.16178
```