# Versuch 1

> ########## ########## Vergleich des einfachen Blocker Beispiels in WinBUGS und STAN ########## ##########

>

>

>

>

> ##### Clear data

> rm(list=ls())

>

>

>

> #### Setting working directory

> setwd("C:/Users/IvanB/Desktop/Masterarbeit/Ergebnisse/zusätzliche Experimente/einfacher Blocker Vergleich WinBUGS und STAN")

>

>

>

> #### Requiering stan

> library("rstan")

> library("rstantools")

> rstan\_options(auto\_write = TRUE)

> #options(mc.cores = parallel::detectCores(1))

> Sys.setenv(LOCAL\_CPPFLAGS = '-march=native')

>

>

>

> #### Assignment data to stan

> rt = c(3, 7, 5, 102, 28, 4, 98, 60, 25, 138, 64, 45, 9, 57, 25, 33,

+ 28, 8, 6, 32, 27, 22)

> nt = c(38, 114, 69, 1533, 355, 59, 945, 632, 278, 1916, 873, 263,

+ 291, 858, 154, 207, 251, 151, 174, 209, 391, 680)

> rc = c(3, 14, 11, 127, 27, 6, 152, 48, 37, 188, 52, 47, 16, 45, 31,

+ 38, 12, 6, 3, 40, 43, 39)

> nc =c(39, 116, 93, 1520, 365, 52, 939, 471, 282, 1921, 583, 266,

+ 293, 883, 147, 213, 122, 154, 134, 218, 364, 674)

> N = 22

>

>

> data\_list <- list(N=N, rt=rt, nt=nt, rc=rc, nc=nc)

>

>

>

> #### Read in inits

>

> inits1 <- function(chain\_id = 1) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits2 <- function(chain\_id = 2) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits3 <- function(chain\_id = 3) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> all.inits <- list(inits1, inits2, inits3)

>

>

>

> #### Compiling

> m <- stan\_model('blocker.stan')

>

>

>

> #### Simulation

> stan\_samples <- sampling(m, data=data\_list, iter=20000, init="all.inits", verbose=T, chain=3, warmup= 10000, control = list(max\_treedepth = 10, adapt\_delta = 0.85)) # !! iter nachher erhöhen

CHECKING DATA AND PREPROCESSING FOR MODEL 'blocker' NOW.

COMPILING MODEL 'blocker' NOW.

STARTING SAMPLER FOR MODEL 'blocker' NOW.

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 1: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 1: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 1: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 1: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 1: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 1: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 1: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 1: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 1: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 1: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 1: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 1:

Chain 1: Elapsed Time: 14.845 seconds (Warm-up)

Chain 1: 21.811 seconds (Sampling)

Chain 1: 36.656 seconds (Total)

Chain 1:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 2: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 2: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 2: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 2: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 2: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 2: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 2: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 2: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 2: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 2: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 2: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 2:

Chain 2: Elapsed Time: 14.502 seconds (Warm-up)

Chain 2: 16.152 seconds (Sampling)

Chain 2: 30.654 seconds (Total)

Chain 2:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 3: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 3: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 3: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 3: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 3: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 3: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 3: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 3: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 3: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 3: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 3: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 3:

Chain 3: Elapsed Time: 14.431 seconds (Warm-up)

Chain 3: 15.925 seconds (Sampling)

Chain 3: 30.356 seconds (Total)

Chain 3:

>

>

> #### Summary

> Stan\_summary <- summary(stan\_samples, pars = c("d", "delta\_new", "sigma\_delta", "sigmasq\_delta" ))$summary

> Stan\_summary

mean se\_mean sd 2.5% 25% 50% 75% 97.5% n\_eff Rhat

d -0.25298831 0.0007408392 0.06191092 -0.3730159222 -0.294859148 -0.253027601 -0.21242926 -0.12945033 6983.716 1.000202

delta\_new -0.25221557 0.0012758108 0.16587380 -0.5913551708 -0.328146975 -0.251897499 -0.17577089 0.07873784 16903.750 1.000163

sigma\_delta 0.09620113 0.0012117934 0.05341343 0.0263480457 0.055904947 0.085546307 0.12481149 0.22730674 1942.870 1.000813

sigmasq\_delta 0.01210756 0.0002764364 0.01437948 0.0006942195 0.003125363 0.007318171 0.01557791 0.05166835 2705.800 1.000556

>

>

>

>

>

> ########## ########## Simulation beendet ########## ##########

# Versuch 2

> ########## ########## Vergleich des einfachen Blocker Beispiels in WinBUGS und STAN ########## ##########

>

>

>

>

> ##### Clear data

> rm(list=ls())

>

>

>

> #### Setting working directory

> setwd("C:/Users/IvanB/Desktop/Masterarbeit/Ergebnisse/zusätzliche Experimente/einfacher Blocker Vergleich WinBUGS und STAN")

>

>

>

> #### Requiering stan

> library("rstan")

> library("rstantools")

> rstan\_options(auto\_write = TRUE)

> #options(mc.cores = parallel::detectCores(1))

> Sys.setenv(LOCAL\_CPPFLAGS = '-march=native')

>

>

>

> #### Assignment data to stan

> rt = c(3, 7, 5, 102, 28, 4, 98, 60, 25, 138, 64, 45, 9, 57, 25, 33,

+ 28, 8, 6, 32, 27, 22)

> nt = c(38, 114, 69, 1533, 355, 59, 945, 632, 278, 1916, 873, 263,

+ 291, 858, 154, 207, 251, 151, 174, 209, 391, 680)

> rc = c(3, 14, 11, 127, 27, 6, 152, 48, 37, 188, 52, 47, 16, 45, 31,

+ 38, 12, 6, 3, 40, 43, 39)

> nc =c(39, 116, 93, 1520, 365, 52, 939, 471, 282, 1921, 583, 266,

+ 293, 883, 147, 213, 122, 154, 134, 218, 364, 674)

> N = 22

>

>

> data\_list <- list(N=N, rt=rt, nt=nt, rc=rc, nc=nc)

>

>

>

> #### Read in inits

>

> inits1 <- function(chain\_id = 1) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits2 <- function(chain\_id = 2) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits3 <- function(chain\_id = 3) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> all.inits <- list(inits1, inits2, inits3)

>

>

>

> #### Compiling

> m <- stan\_model('blocker.stan')

>

>

>

> #### Simulation

> stan\_samples <- sampling(m, data=data\_list, iter=20000, init="all.inits", verbose=T, chain=3, warmup= 10000, control = list(max\_treedepth = 10, adapt\_delta = 0.85)) # !! iter nachher erhöhen

CHECKING DATA AND PREPROCESSING FOR MODEL 'blocker' NOW.

COMPILING MODEL 'blocker' NOW.

STARTING SAMPLER FOR MODEL 'blocker' NOW.

SAMPLING FOR MODEL 'blocker' NOW (CHAIN 1).

Chain 1:

Chain 1: Gradient evaluation took 0.001 seconds

Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 10 seconds.

Chain 1: Adjust your expectations accordingly!

Chain 1:

Chain 1:

Chain 1: Iteration: 1 / 20000 [ 0%] (Warmup)

Chain 1: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 1: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 1: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 1: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 1: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 1: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 1: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 1: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 1: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 1: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 1: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 1:

Chain 1: Elapsed Time: 15.784 seconds (Warm-up)

Chain 1: 22.646 seconds (Sampling)

Chain 1: 38.43 seconds (Total)

Chain 1:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 2: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 2: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 2: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 2: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 2: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 2: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 2: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 2: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 2: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 2: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 2: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 2:

Chain 2: Elapsed Time: 14.084 seconds (Warm-up)

Chain 2: 18.97 seconds (Sampling)

Chain 2: 33.054 seconds (Total)

Chain 2:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 3: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 3: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 3: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 3: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 3: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 3: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 3: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 3: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 3: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 3: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 3: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 3:

Chain 3: Elapsed Time: 14.001 seconds (Warm-up)

Chain 3: 18.133 seconds (Sampling)

Chain 3: 32.134 seconds (Total)

Chain 3:

>

>

> #### Summary

> Stan\_summary <- summary(stan\_samples, pars = c("d", "delta\_new", "sigma\_delta", "sigmasq\_delta" ))$summary

> Stan\_summary

mean se\_mean sd 2.5% 25% 50% 75% 97.5% n\_eff Rhat

d -0.25319312 0.0006761303 0.06231314 -0.3733570505 -0.295145139 -0.253656631 -0.21227655 -0.12825209 8493.732 1.000204

delta\_new -0.25519616 0.0013181935 0.16804329 -0.5844456243 -0.329428857 -0.254995329 -0.17833334 0.07221204 16251.148 1.000066

sigma\_delta 0.09590783 0.0011681270 0.05395500 0.0263068139 0.054444477 0.085180524 0.12588531 0.22676382 2133.454 1.001059

sigmasq\_delta 0.01210936 0.0002653621 0.01446602 0.0006920485 0.002964201 0.007255722 0.01584711 0.05142183 2971.806 1.000624

>

>

>

>

>

# Versuch 3

>

> ########## ########## Vergleich des einfachen Blocker Beispiels in WinBUGS und STAN ########## ##########

>

>

>

>

> ##### Clear data

> rm(list=ls())

>

>

>

> #### Setting working directory

> setwd("C:/Users/IvanB/Desktop/Masterarbeit/Ergebnisse/zusätzliche Experimente/einfacher Blocker Vergleich WinBUGS und STAN")

>

>

>

> #### Requiering stan

> library("rstan")

> library("rstantools")

> rstan\_options(auto\_write = TRUE)

> #options(mc.cores = parallel::detectCores(1))

> Sys.setenv(LOCAL\_CPPFLAGS = '-march=native')

>

>

>

> #### Assignment data to stan

> rt = c(3, 7, 5, 102, 28, 4, 98, 60, 25, 138, 64, 45, 9, 57, 25, 33,

+ 28, 8, 6, 32, 27, 22)

> nt = c(38, 114, 69, 1533, 355, 59, 945, 632, 278, 1916, 873, 263,

+ 291, 858, 154, 207, 251, 151, 174, 209, 391, 680)

> rc = c(3, 14, 11, 127, 27, 6, 152, 48, 37, 188, 52, 47, 16, 45, 31,

+ 38, 12, 6, 3, 40, 43, 39)

> nc =c(39, 116, 93, 1520, 365, 52, 939, 471, 282, 1921, 583, 266,

+ 293, 883, 147, 213, 122, 154, 134, 218, 364, 674)

> N = 22

>

>

> data\_list <- list(N=N, rt=rt, nt=nt, rc=rc, nc=nc)

>

>

>

> #### Read in inits

>

> inits1 <- function(chain\_id = 1) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits2 <- function(chain\_id = 2) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> inits3 <- function(chain\_id = 3) {

+ list(d <- 0,

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ sigmasq\_delta <- 1,

+ mu <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),

+ delta\_new <- 0)

+ }

>

> all.inits <- list(inits1, inits2, inits3)

>

>

>

> #### Compiling

> m <- stan\_model('blocker.stan')

>

>

>

> #### Simulation

> stan\_samples <- sampling(m, data=data\_list, iter=20000, init="all.inits", verbose=T, chain=3, warmup= 10000, control = list(max\_treedepth = 10, adapt\_delta = 0.85)) # !! iter nachher erhöhen

CHECKING DATA AND PREPROCESSING FOR MODEL 'blocker' NOW.

COMPILING MODEL 'blocker' NOW.

STARTING SAMPLER FOR MODEL 'blocker' NOW.

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 1: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 1: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 1: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 1: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 1: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 1: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 1: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 1: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 1: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 1: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 1: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 1:

Chain 1: Elapsed Time: 14.981 seconds (Warm-up)

Chain 1: 17.11 seconds (Sampling)

Chain 1: 32.091 seconds (Total)

Chain 1:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 2: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 2: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 2: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 2: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 2: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 2: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 2: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 2: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 2: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 2: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 2: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 2:

Chain 2: Elapsed Time: 15.341 seconds (Warm-up)

Chain 2: 11.896 seconds (Sampling)

Chain 2: 27.237 seconds (Total)

Chain 2:

SAMPLING FOR MODEL 'blocker' 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 / 20000 [ 0%] (Warmup)

Chain 3: Iteration: 2000 / 20000 [ 10%] (Warmup)

Chain 3: Iteration: 4000 / 20000 [ 20%] (Warmup)

Chain 3: Iteration: 6000 / 20000 [ 30%] (Warmup)

Chain 3: Iteration: 8000 / 20000 [ 40%] (Warmup)

Chain 3: Iteration: 10000 / 20000 [ 50%] (Warmup)

Chain 3: Iteration: 10001 / 20000 [ 50%] (Sampling)

Chain 3: Iteration: 12000 / 20000 [ 60%] (Sampling)

Chain 3: Iteration: 14000 / 20000 [ 70%] (Sampling)

Chain 3: Iteration: 16000 / 20000 [ 80%] (Sampling)

Chain 3: Iteration: 18000 / 20000 [ 90%] (Sampling)

Chain 3: Iteration: 20000 / 20000 [100%] (Sampling)

Chain 3:

Chain 3: Elapsed Time: 14.128 seconds (Warm-up)

Chain 3: 11.479 seconds (Sampling)

Chain 3: 25.607 seconds (Total)

Chain 3:

>

>

> #### Summary

> Stan\_summary <- summary(stan\_samples, pars = c("d", "delta\_new", "sigma\_delta", "sigmasq\_delta" ))$summary

> Stan\_summary

mean se\_mean sd 2.5% 25% 50% 75% 97.5% n\_eff Rhat

d -0.25257491 0.0007521899 0.06177526 -0.3739030443 -0.292933979 -0.253730943 -0.21225752 -0.13066110 6744.881 1.000302

delta\_new -0.25359625 0.0013381059 0.16731339 -0.5880155432 -0.327948428 -0.255208950 -0.17732700 0.07743618 15634.372 1.000023

sigma\_delta 0.09578720 0.0011740272 0.05362484 0.0277271247 0.055213264 0.084858489 0.12442076 0.22869235 2086.295 1.001165

sigmasq\_delta 0.01205072 0.0002733396 0.01470340 0.0007687934 0.003048505 0.007200963 0.01548052 0.05230019 2893.546 1.000377

>

>

>

>

>

> ########## ########## Simulation beendet ########## ##########