```
knitr::opts_chunk$set(message = FALSE, warning = FALSE)
```

```
library(rstan); rstan_options(javascript=FALSE)
library(bayesplot)
library(dplyr)
options(mc.cores = parallel::detectCores())
rstan_options(auto_write = T)

dat <- read.csv('final_data.csv')
```

```
dat <- dat %>%
  filter(skew != 'control')
dat <- dat %>%
  mutate(cho = ifelse(true_response == 'f', 1, -1))


dat$P_A1 <- dat$P_A1 / 100
dat$P_A2 <- dat$P_A2 / 100
dat$P_B1 <- dat$P_B1 / 100
dat$P_B2 <- dat$P_B2 / 100
```

```
ids <- unique(dat$Prolific_ID)
for(j in 1:length(ids)){
  dat$tid[dat$Prolific_ID==ids[j]] <- j
}
tids <- unique(dat$tid)

dat$rt <- as.numeric(dat$rt/1000)

dat <- dat %>%
  filter(test_part == 'cc' | test_part == 'ss')

dat <- dat %>%
  mutate(con = ifelse(test_part == 'cc', 1 , -1)  )
```

```
dat <- dat %>%
  mutate(index1 = as.numeric(ifelse(O_A1<O_A2, 1, -1)) ,
         index2 = as.numeric(ifelse(O_B1<O_B2, 1, -1)),)

dat <- dat %>%
  # Swap values if oa_condition is not 0
  rowwise() %>%
  mutate(
    oa3 = if_else(index1 == 1, O_A1, O_A2),
    oa4 = if_else(index1 == 1, O_A2, O_A1),
    pa3 = if_else(index1 == 1, P_A1, P_A2),
    pa4 = if_else(index1 == 1, P_A2, P_A1),
    ob3 = if_else(index2 == 1, O_B1, O_B2),
    ob4 = if_else(index2 == 1, O_B2, O_B1),
    pb3 = if_else(index2 == 1, P_B1, P_B2),
    pb4 = if_else(index2 == 1, P_B2, P_B1),
  ) %>%
  ungroup()
```

```
oa = as.matrix(dat[, c("oa3", "oa4")])
ob = as.matrix(dat[, c("ob3", "ob4")])
pa = as.matrix(dat[, c("pa3", "pa4")])
pb = as.matrix(dat[, c("pb3", "pb4")])
```

```
dataList  = list(cho = dat$cho,rt = dat$rt, participant = dat$tid,N=nrow(dat),  L
= length(tids),starting_point=0.5,
                 oa = as.matrix(dat[, c("oa3", "oa4")]),
                 ob = as.matrix(dat[, c("ob3", "ob4")]),
                 pa = as.matrix(dat[, c("pa3", "pa4")]),
                 pb = as.matrix(dat[, c("pb3", "pb4")]),
                 con = dat$con
                 )



parameters = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu
_theta",'transf_mu_delta_gamma', 'transf_mu_gamma', 'sd_threshold',"sd_alpha","sd_
ndt", 'sd_theta', 'sd_gamma','sd_delta_gamma', "alpha_sbj","threshold_sbj","ndt_sb
j",'theta_sbj', 'gamma_sbj', 'delta_gamma_sbj', "log_lik")
```

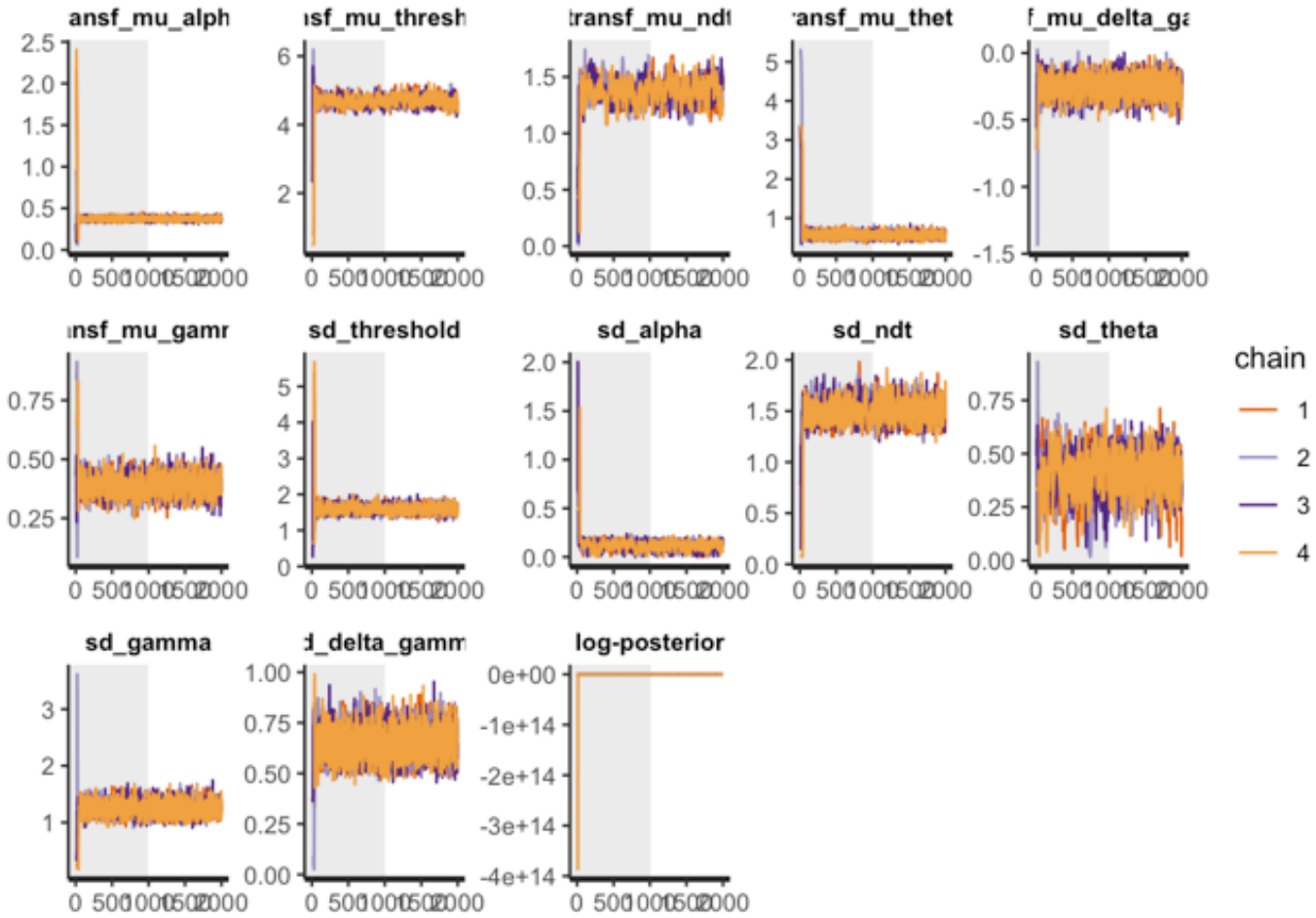
```
initFunc <-function (i) {
  initList=list()
  for (ll in 1:i){
    initList[[ll]] = list(
                          mu_alpha = runif(1,-1.4587,2.5413),
                          sd_alpha = runif(1,0,1),
                          mu_threshold = runif(1,-0.5, 2.5),
                          sd_threshold = runif(1,0,1),
                          mu_ndt = runif(1, -1.5, 0),
                          sd_ndt = runif(1, 0, 1),
                          mu_theta = runif(1,0, 6),
                          sd_theta = runif(1,0,1),
                          mu_gamma = runif(1,-1, 1),
                          sd_gamma = runif(1, 0, 1),
                          mu_delta_gamma = runif(1,-1, 1),
                          sd_delta_gamma = runif(1, 0, 1),
                          z_alpha = runif(length(tids),-0.1,0.1),
                          z_theta = runif(length(tids),-0.1,0.1),
                          z_threshold = runif(length(tids),-0.1,0.1),
                          z_ndt = runif(length(tids),-0.1,0.1),
                          z_gamma = runif(length(tids),-0.1,0.1),
                          z_delta_gamma = runif(length(tids),-0.1,0.1)
    )
  }

  return(initList)
}
```

```
m <- stan_model("EU_prob cc.stan")
```

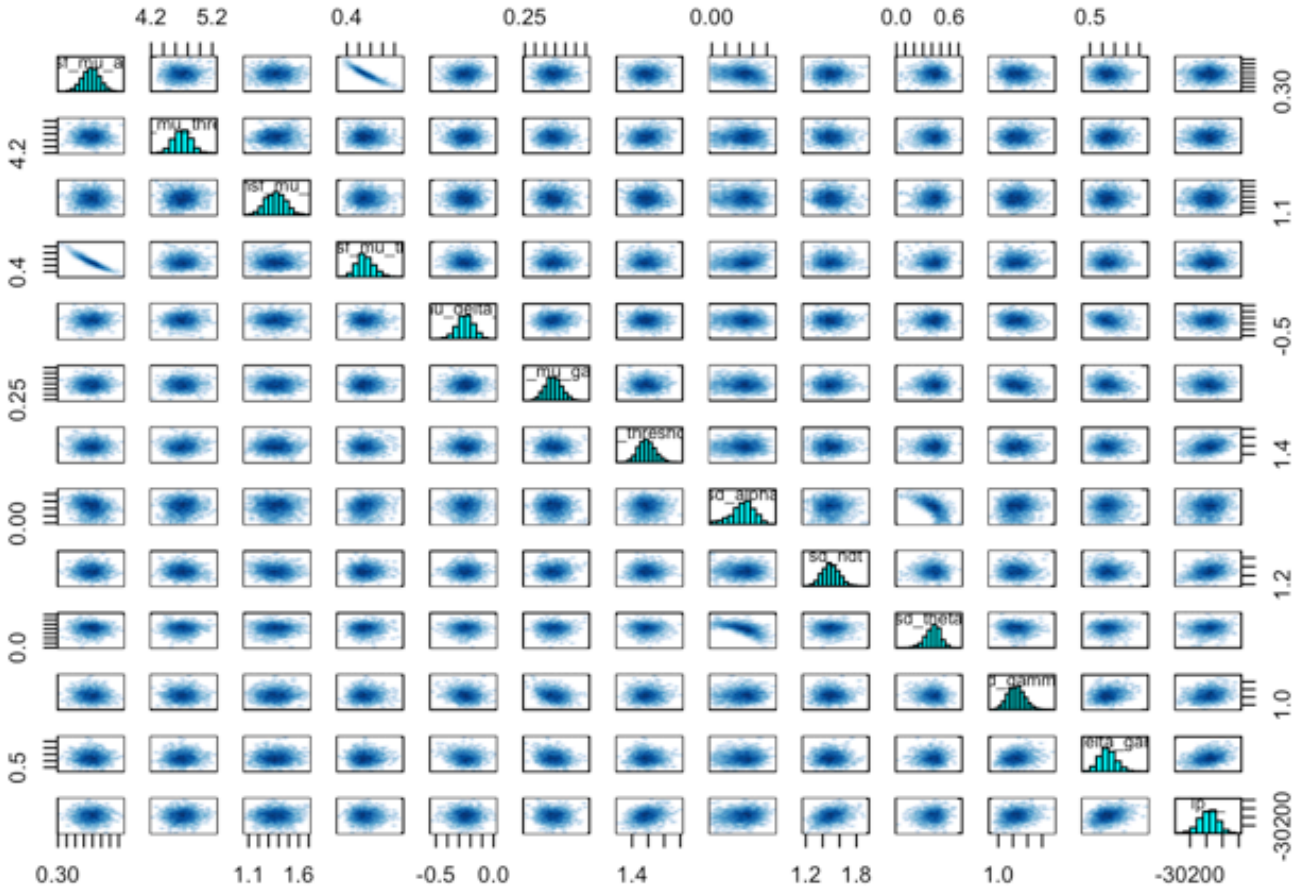
```
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -mmacosx-version-min=10.13 -I"/Library/Frameworks/R.framework/Resources/i
nclude" -DNDEBUG   -I"/Library/Frameworks/R.framework/Versions/4.2/Resources/libra
ry/Rcpp/include/"  -I"/Library/Frameworks/R.framework/Versions/4.2/Resources/libra
ry/RcppEigen/include/"  -I"/Library/Frameworks/R.framework/Versions/4.2/Resources/
library/RcppEigen/include/unsupported"  -I"/Library/Frameworks/R.framework/Version
s/4.2/Resources/library/BH/include" -I"/Library/Frameworks/R.framework/Versions/4.
2/Resources/library/StanHeaders/include/src/"  -I"/Library/Frameworks/R.framework/
Versions/4.2/Resources/library/StanHeaders/include/"  -I"/Library/Frameworks/R.fra
mework/Versions/4.2/Resources/library/RcppParallel/include/"  -I"/Library/Framewor
ks/R.framework/Versions/4.2/Resources/library/rstan/include" -DEIGEN_NO_DEBUG  -DB
OOST_DISABLE_ASSERTS  -DBOOST_PENDING_INTEGER_LOG2_HPP  -DSTAN_THREADS  -DUSE_STAN
C3 -DSTRICT_R_HEADERS  -DBOOST_PHOENIX_NO_VARIADIC_EXPRESSION  -D_HAS_AUTO_PTR_ETC
=0  -include '/Library/Frameworks/R.framework/Versions/4.2/Resources/library/StanH
eaders/include/stan/math/prim/fun/Eigen.hpp'  -D_REENTRANT -DRCPP_PARALLEL_USE_TBB
=1   -I/usr/local/include   -fPIC  -Wall -g -O2  -c foo.c -o foo.o
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/li
brary/StanHeaders/include/stan/math/prim/fun/Eigen.hpp:22:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/li
brary/RcppEigen/include/Eigen/Dense:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/li
brary/RcppEigen/include/Eigen/Core:88:
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/includ
e/Eigen/src/Core/util/Macros.h:628:1: error: unknown type name 'namespace'
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/includ
e/Eigen/src/Core/util/Macros.h:628:16: error: expected ';' after top level declara
tor
## namespace Eigen {
##                ^
##                ;
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/li
brary/StanHeaders/include/stan/math/prim/fun/Eigen.hpp:22:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/li
brary/RcppEigen/include/Eigen/Dense:1:
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/includ
e/Eigen/Core:96:10: fatal error: 'complex' file not found
## #include <complex>
##          ^~~~~~~~
## 3 errors generated.
## make: *** [foo.o] Error 1
```

```
dsamples <- sampling(m,
                data=dataList,
                pars=parameters,
                iter=2000,
                chains=4,#If not specified, gives random inits
                init = initFunc(4),
                warmup = 1000,  # Stands for burn-in; Default = iter/2
                seed = 12,
                refresh = 0


                )
```

```
#"transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu_theta",'transf_mu_delta_gamma', 'transf_mu_gamma', 'sd_threshold',"sd_alpha","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta_gamma', "alpha_sbj","threshold_sbj","ndt_sbj",'theta_sbj', 'gamma_sbj', 'delta_gamma_sbj',
rstan::traceplot(dsamples, pars=c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu_theta",'transf_mu_delta_gamma', 'transf_mu_gamma', 'sd_threshold',"sd_alpha","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta_gamma', "lp__"), inc_warmup = TRUE, nrow = 3)
```



```
pairs(dsamples, pars = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt",
"transf_mu_theta",'transf_mu_delta_gamma', 'transf_mu_gamma', 'sd_threshold',"sd_a
lpha","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta_gamma', "lp__"))
```

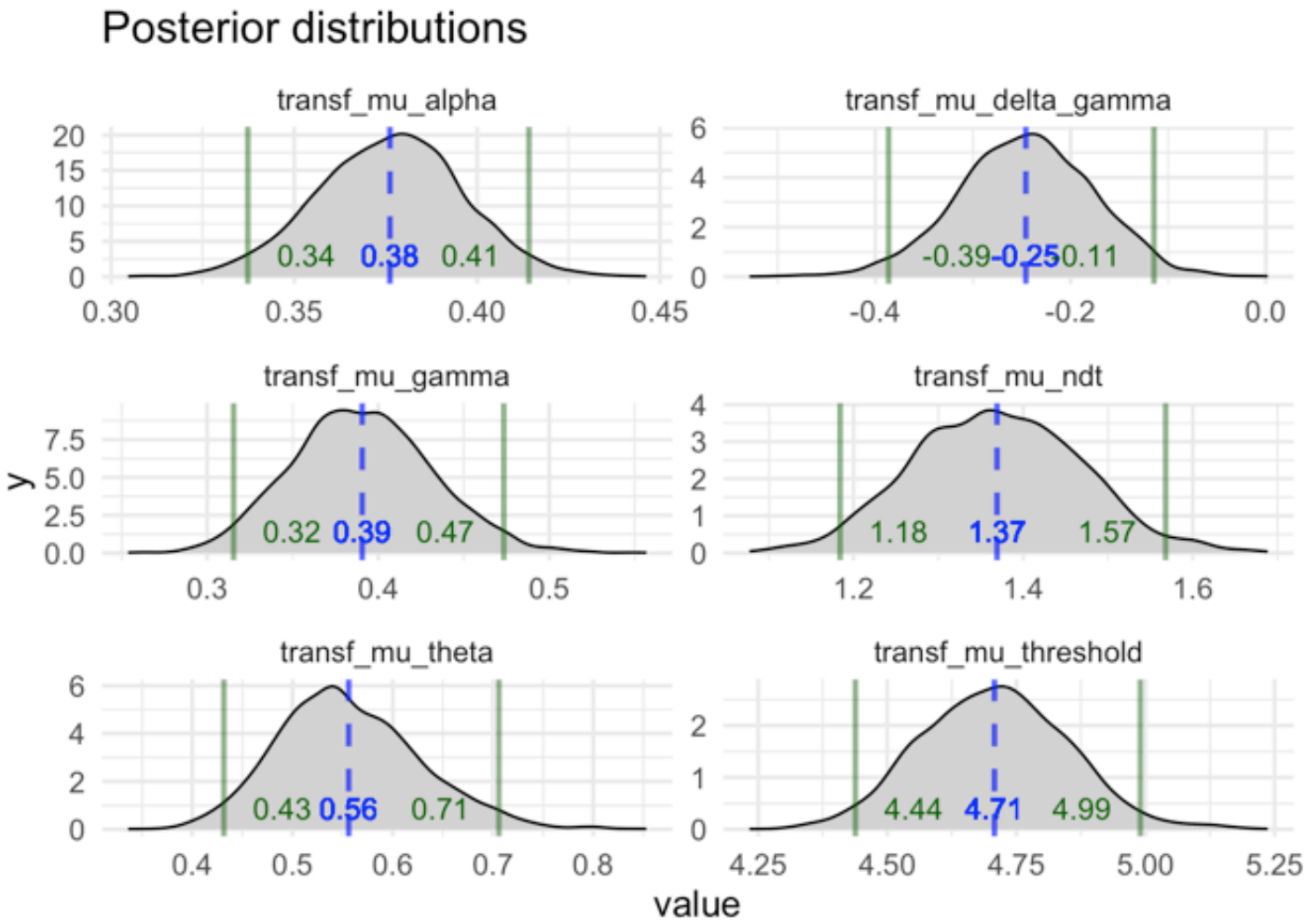



```
print(dsamples, pars = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt",
"transf_mu_theta",'transf_mu_delta_gamma', 'transf_mu_gamma', 'sd_threshold',"sd_a
lpha","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta_gamma', "lp__"))
```

```
## Inference for Stan model: anon_model.
## 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%
## transf_mu_alpha           0.38    0.00  0.02      0.34      0.36      0.38
## transf_mu_threshold       4.71    0.01  0.14      4.44      4.61      4.71
## transf_mu_ndt             1.37    0.01  0.10      1.18      1.30      1.37
## transf_mu_theta           0.56    0.00  0.07      0.43      0.51      0.55
## transf_mu_delta_gamma    -0.25    0.00  0.07     -0.39     -0.29     -0.25
## transf_mu_gamma           0.39    0.00  0.04      0.32      0.36      0.39
## sd_threshold              1.58    0.00  0.10      1.39      1.51      1.58
## sd_alpha                  0.11    0.00  0.04      0.02      0.09      0.12
## sd_ndt                    1.49    0.00  0.10      1.31      1.43      1.49
## sd_theta                  0.41    0.00  0.09      0.22      0.36      0.42
## sd_gamma                  1.25    0.00  0.12      1.04      1.16      1.24
## sd_delta_gamma            0.65    0.00  0.07      0.52      0.60      0.64
## lp__                 -30143.37    1.16 27.77 -30200.49 -30161.96 -30142.06
##                           75%     97.5% n_eff Rhat
## transf_mu_alpha           0.39      0.41  3423 1.00
## transf_mu_threshold       4.80      4.99   239 1.00
## transf_mu_ndt             1.44      1.57   165 1.01
## transf_mu_theta           0.60      0.71  4053 1.00
## transf_mu_delta_gamma    -0.20     -0.11  1503 1.00
## transf_mu_gamma           0.42      0.47   736 1.01
## sd_threshold              1.65      1.80   620 1.00
## sd_alpha                  0.14      0.18   336 1.01
## sd_ndt                    1.56      1.70   435 1.01
## sd_theta                  0.47      0.56   465 1.00
## sd_gamma                  1.32      1.48  1130 1.00
## sd_delta_gamma            0.69      0.79  1470 1.00
## lp__                 -30124.75 -30092.00   574 1.00
##
## Samples were drawn using NUTS(diag_e) at Sun Jan 14 16:19:56 2024.
## 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).
```

```
library(ggplot2)
library(tidyverse) # for the gather function

samples_matrix <- as.matrix(dsamples)
means <- colMeans(samples_matrix)
hpd_interval <- t(apply(samples_matrix, 2, function(x) quantile(x, probs=c(0.025,
0.975))))



parameters <- c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_m
```

```
u_theta",'transf_mu_delta_gamma', 'transf_mu_gamma')

# Reshape data to a long format
df_long <- as.data.frame(samples_matrix) %>%
  gather(key = "parameter", value = "value", parameters)

# Convert hpd_interval to a data frame and name the columns
hpd_interval_sub <- hpd_interval[parameters, ]
hpd_df <- as.data.frame(hpd_interval_sub)
colnames(hpd_df) <- c("lower", "upper")
rownames(hpd_df) <- parameters
hpd_df$parameter <- rownames(hpd_df)



# Aesthetic enhancements
theme_set(theme_minimal(base_size = 14)) # Set the default theme

custom_palette <- c("density_fill" = "lightgray",
                    "mean_line" = "blue",
                    "hpd_line" = "darkgreen")

# Add text labels for mean, lower, and upper HPD values
df_long <- df_long %>%
  group_by(parameter) %>%
  mutate(mean = means[parameter])

hpd_df <- hpd_df %>%
  mutate(mid = (lower + upper) / 2)

p <- ggplot(df_long, aes(x = value)) +
  geom_density(aes(fill = "density_fill")) +
  scale_fill_manual(values = custom_palette, guide = FALSE) +
  geom_vline(aes(xintercept = mean, color = "mean_line"), linetype = "dashed", siz
e = 1, alpha = 0.7) +
  geom_text(data = df_long, aes(x = mean, y = 0, label = round(mean, 2)), vjust =
-0.5, hjust = 0.5, size = 4, color = custom_palette["mean_line"]) +
  geom_vline(data = hpd_df, aes(xintercept = lower, color = "hpd_line"), linetype
= "solid", size = 1, alpha = 0.5) +
  geom_text(data = hpd_df, aes(x = lower, y = 0, label = round(lower, 2)), vjust =
-0.5, hjust = -0.5, size = 4, color = custom_palette["hpd_line"]) +
  geom_vline(data = hpd_df, aes(xintercept = upper, color = "hpd_line"), linetype
= "solid", size = 1, alpha = 0.5) +
  geom_text(data = hpd_df, aes(x = upper, y = 0, label = round(upper, 2)), vjust =
-0.5, hjust = 1.5, size = 4, color = custom_palette["hpd_line"]) +
  facet_wrap(~ parameter, scales = "free", ncol = 2) +
  scale_color_manual(values = custom_palette, guide = 'none') +
  labs(title = "Posterior distributions")

print(p)
```


Posterior distributions
transf_mu_alpha
0.34 0.38 0.41
transf_mu_delta_gamma
-0.39 -0.25 0.11
transf_mu_gamma
0.32 0.39 0.47
transf_mu_ndt
1.18 1.37 1.57
transf_mu_theta
0.43 0.56 0.71
transf_mu_threshold
4.44 4.71 4.99
y
value