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



dat <- read.csv('final_data.csv')
dat = dat %>%
  mutate(cho = 0,
         cho = ifelse(response == "f", 1*risk_index, cho),
         cho = ifelse(response == "j", -1*risk_index, cho))

dat <- dat %>%
  filter(skew != 'control')


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

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

dat <- dat %>%
  mutate(con = ifelse(test_part == "cc", 1, -1))
dat$rt <- dat$rt/1000


# Assuming your dataframe is named 'df'
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
```

```
oa = as.matrix(dat[, c("O_A1", "O_A2")])
ob = as.matrix(dat[, c("O_B1", "O_B2")])
pa = as.matrix(dat[, c("P_A1", "P_A2")])
pb = as.matrix(dat[, c("P_B1", "P_B2")])
```

```
dataList  = list(cho = dat$cho,rt = dat$rt, participant = dat$tid,N=nrow(dat),  L
= length(tids),starting_point=0.5,
                 oa = as.matrix(dat[, c("O_A1", "O_A2")]),
                 ob = as.matrix(dat[, c("O_B1", "O_B2")]),
                 pa = as.matrix(dat[, c("P_A1", "P_A2")]),
                 pb = as.matrix(dat[, c("P_B1", "P_B2")]),
                 con =  dat$con
                 )

parameters = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu
_theta","transf_mu_delta_theta", 'sd_threshold',"sd_alpha","sd_ndt", 'sd_theta', '
sd_delta_theta', "alpha_sbj","threshold_sbj","ndt_sbj",'theta_sbj','delta_theta_sb
j', "log_lik")
```

```
initFunc <-function (i) {
  initList=list()
  for (ll in 1:i){
    initList[[ll]] = list(mu_alpha = runif(1,-1.4578,2.5413),
                          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_delta_theta = runif(1, -1, 1),
                          sd_delta_theta = 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_delta_theta = runif(length(tids),-0.1,0.1)

    )
  }

  return(initList)
}
```

```
m <- stan_model("EU_Baseline.stan")
```

```
## hash mismatch so recompiling; make sure Stan code ends with a blank line
```

```
## Trying to compile a simple C file
```

```
## 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  -DBOOST_NO
_AUTO_PTR  -include '/Library/Frameworks/R.framework/Versions/4.2/Resources/librar
y/StanHeaders/include/stan/math/prim/mat/fun/Eigen.hpp'  -D_REENTRANT -DRCPP_PARAL
LEL_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/mat/fun/Eigen.hpp:13:
## 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/mat/fun/Eigen.hpp:13:
## 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
```

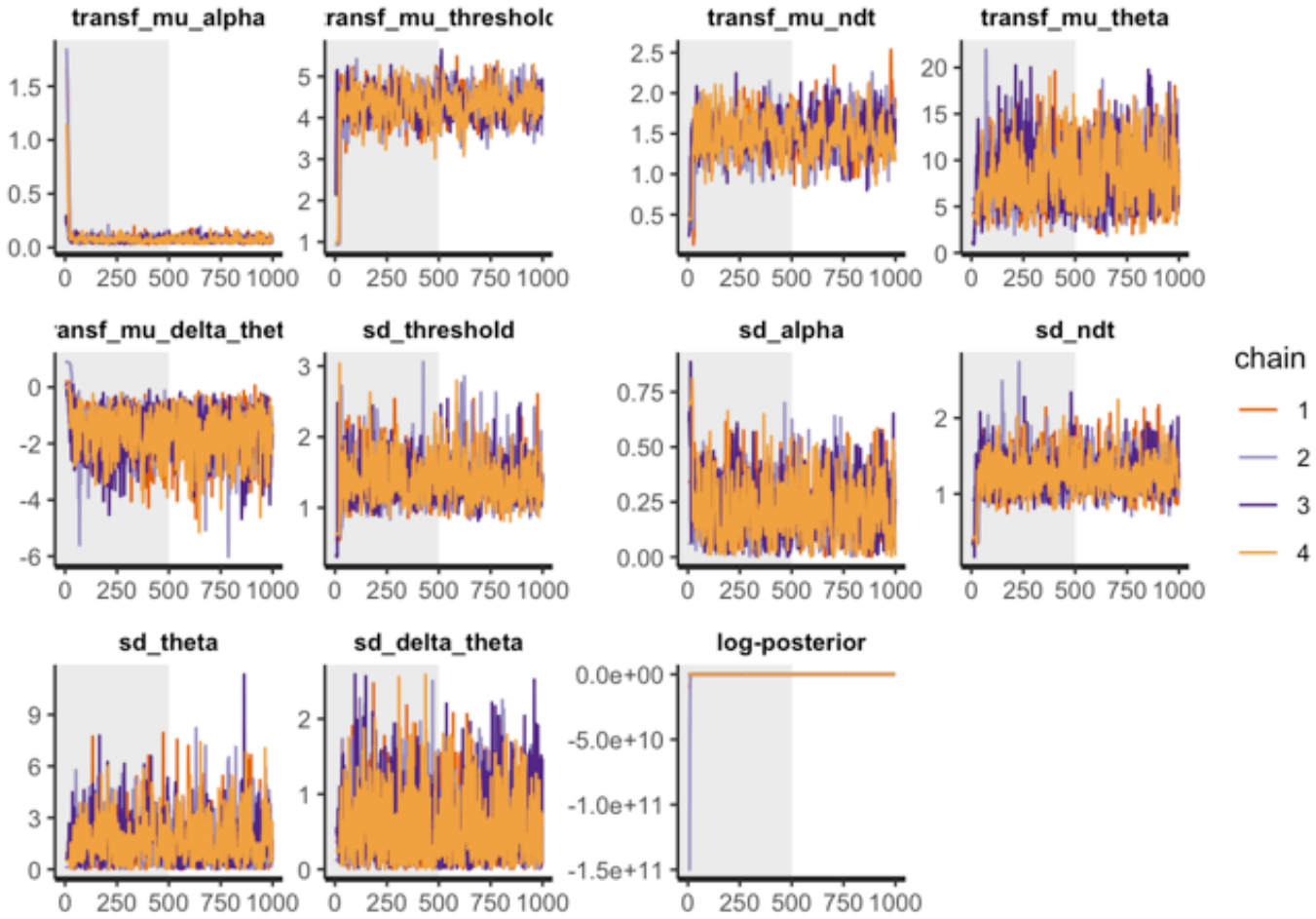
```
dsamples <- sampling(m,
                data=dataList,
                pars=parameters,
                iter=1000,
                chains=4,#If not specified, gives random inits
                init = initFunc(4),
                warmup = 500,  # Stands for burn-in; Default = iter/2
                seed = 12,  # Setting seed; Default is random seed
                refresh = 0
                )
```

```
## Warning: There were 8 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
```

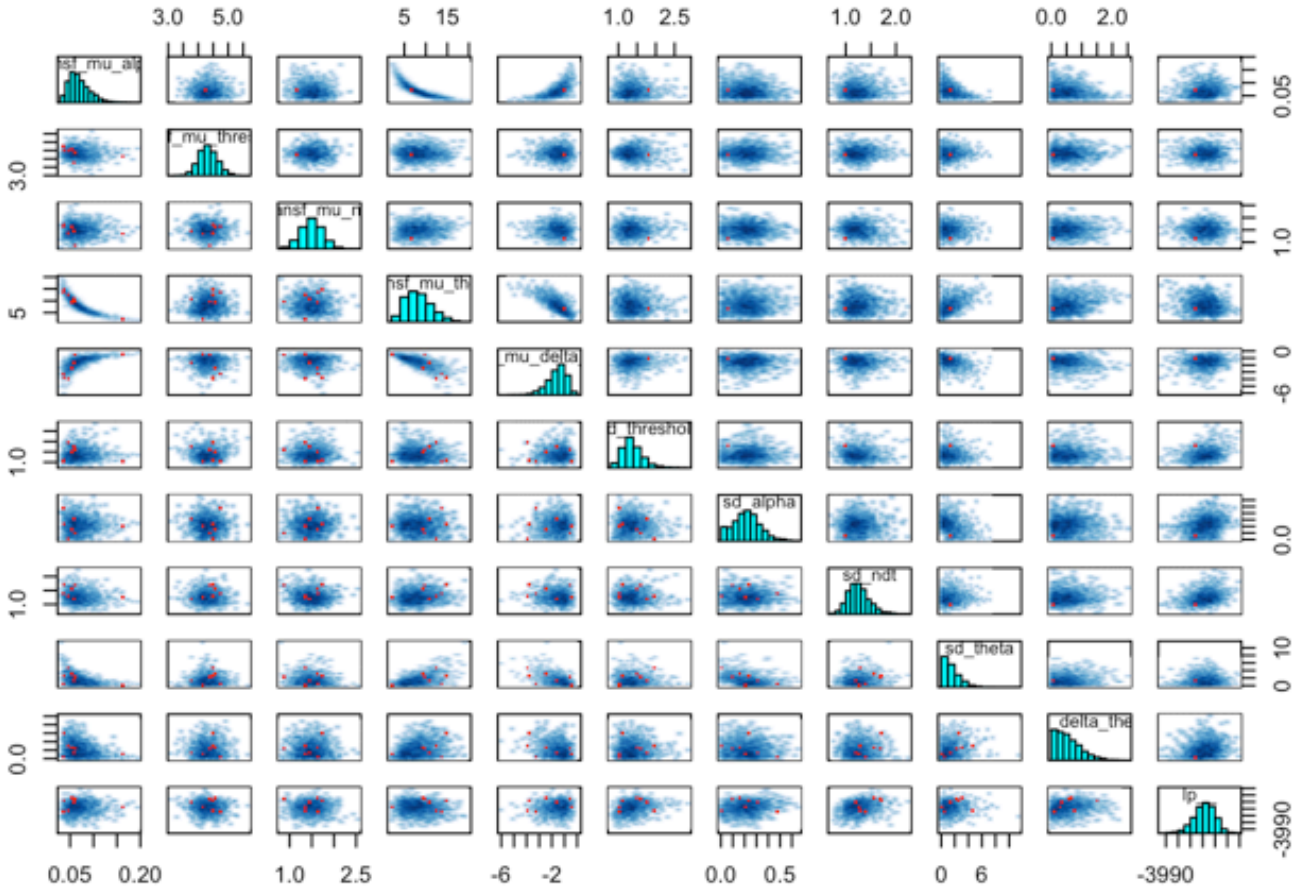
```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior mea
ns and medians may be unreliable.
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess
```

```
#"transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu_theta","trans
f_mu_delta_theta", 'sd_threshold',"sd_alpha","sd_ndt", 'sd_theta', 'sd_delta_thet
a', "alpha_sbj","threshold_sbj","ndt_sbj",'theta_sbj','delta_theta_sbj', "log_lik"

rstan::traceplot(dsamples, pars=c("transf_mu_alpha","transf_mu_threshold","transf_
mu_ndt", "transf_mu_theta","transf_mu_delta_theta", 'sd_threshold',"sd_alpha","sd_
ndt", 'sd_theta', 'sd_delta_theta', "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_theta", 'sd_threshold',"sd_alpha","sd_ndt", 'sd
_theta', 'sd_delta_theta', "lp__"))
```

```
## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter
```



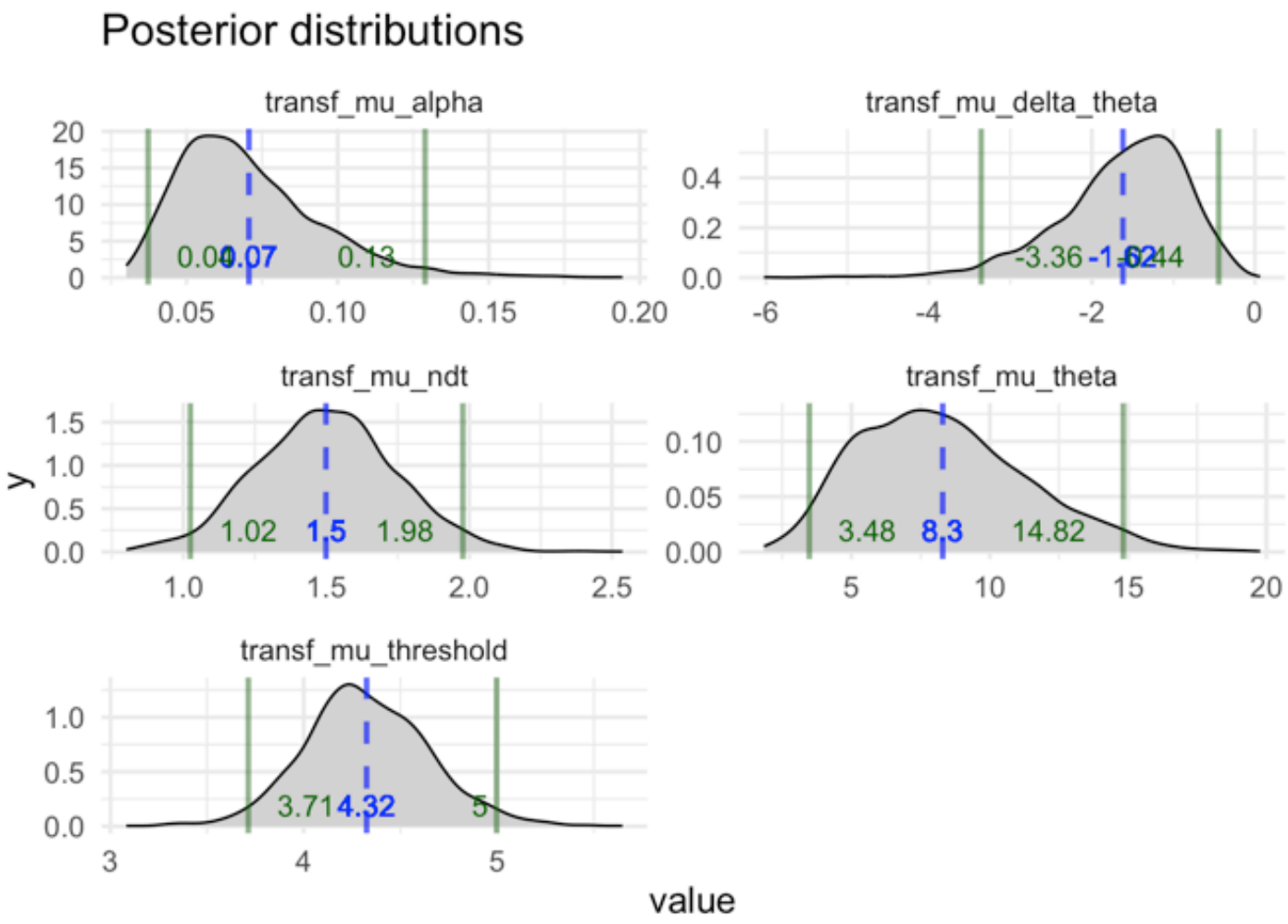
```
print(dsamples, pars = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt",
"transf_mu_theta","transf_mu_delta_theta", 'sd_threshold',"sd_alpha","sd_ndt", 'sd
_theta', 'sd_delta_theta', "lp__"))
```

```
## Inference for Stan model: EU_Baseline.
## 4 chains, each with iter=1000; warmup=500; thin=1;
## post-warmup draws per chain=500, total post-warmup draws=2000.
##
##                          mean se_mean   sd     2.5%      25%      50%      75%
## transf_mu_alpha          0.07    0.00 0.02     0.04     0.05     0.07     0.08
## transf_mu_threshold      4.32    0.02 0.32     3.71     4.12     4.31     4.53
## transf_mu_ndt            1.50    0.02 0.24     1.02     1.33     1.50     1.65
## transf_mu_theta          8.30    0.13 3.02     3.48     6.00     7.99    10.23
## transf_mu_delta_theta   -1.62    0.03 0.78    -3.36    -2.05    -1.50    -1.07
## sd_threshold             1.38    0.01 0.29     0.94     1.19     1.34     1.53
## sd_alpha                 0.22    0.01 0.12     0.01     0.14     0.21     0.29
## sd_ndt                   1.26    0.01 0.22     0.89     1.10     1.24     1.38
## sd_theta                 1.56    0.06 1.27     0.04     0.62     1.28     2.17
## sd_delta_theta           0.56    0.01 0.42     0.02     0.23     0.49     0.81
## lp__                 -3958.67    0.40 8.86 -3977.08 -3964.05 -3958.20 -3952.57
##                         97.5% n_eff Rhat
## transf_mu_alpha          0.13   638 1.01
## transf_mu_threshold      5.00   266 1.00
## transf_mu_ndt            1.98   217 1.02
## transf_mu_theta         14.82   542 1.00
## transf_mu_delta_theta   -0.44   561 1.01
## sd_threshold             2.08   623 1.01
## sd_alpha                 0.48   477 1.00
## sd_ndt                   1.77   626 1.00
## sd_theta                 4.63   383 1.01
## sd_delta_theta           1.57  1001 1.00
## lp__                 -3943.25   493 1.01
##
## Samples were drawn using NUTS(diag_e) at Wed Nov 22 13:14:59 2023.
## 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_theta")

# 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 = FALSE) +
  labs(title = "Posterior distributions")
```

```
print(p)
```


Posterior distributions
transf_mu_alpha
transf_mu_delta_theta
transf_mu_ndt
transf_mu_theta
transf_mu_threshold
0.04 0.07 0.13
-3.36 -1.62 0.44
1.02 1.5 1.98
3.48 8.3 14.82
3.71 4.32 5
y
value