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

```r
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')
```

```r
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
```

```r
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', 'transf_mu_gamma', 'sd_threshold',"sd_alpha","sd_ndt",
'sd_theta', 'sd_gamma','sd_delta', "alpha_sbj","threshold_sbj","ndt_sbj",'theta_sb
j', 'gamma_sbj', 'delta_sbj', "log_lik")
```

```r
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 = runif(1,-1, 1),
                           sd_delta = 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 = runif(length(tids),-0.1,0.1)

    )
  }

  return(initList)
}
```

```r
m <- stan_model("EU_prob.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=1000,
                chains=4,#If not specified, gives random inits
                init = initFunc(4),
                warmup = 500,  # Stands for burn-in; Default = iter/2
                seed = 12

                )
```
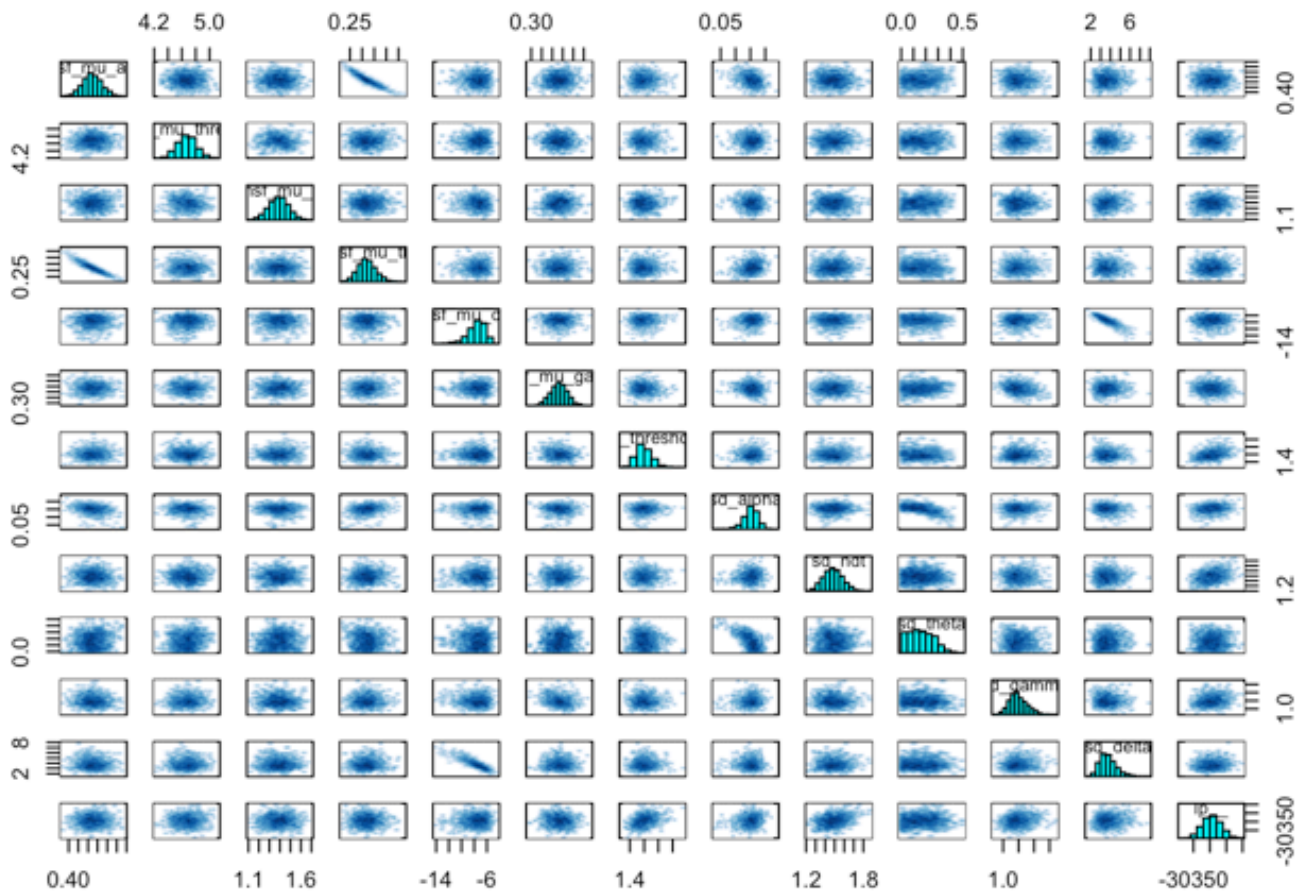
```
#"transf_mu_alpha","transf_mu_threshold","transf_mu_ndt", "transf_mu_theta",'trans
f_mu_delta', 'transf_mu_gamma', 'sd_threshold',"sd_alpha","sd_ndt", 'sd_theta', 's
d_gamma','sd_delta', "alpha_sbj","threshold_sbj","ndt_sbj",'theta_sbj', 'gamma_sb
j', 'delta_sbj',
rstan::traceplot(dsamples, pars=c("transf_mu_alpha","transf_mu_threshold","transf_
mu_ndt", "transf_mu_theta",'transf_mu_delta', 'transf_mu_gamma', 'sd_threshold',"s
d_alpha","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta', "lp__"), inc_warmup = TRUE,
nrow = 3)
```
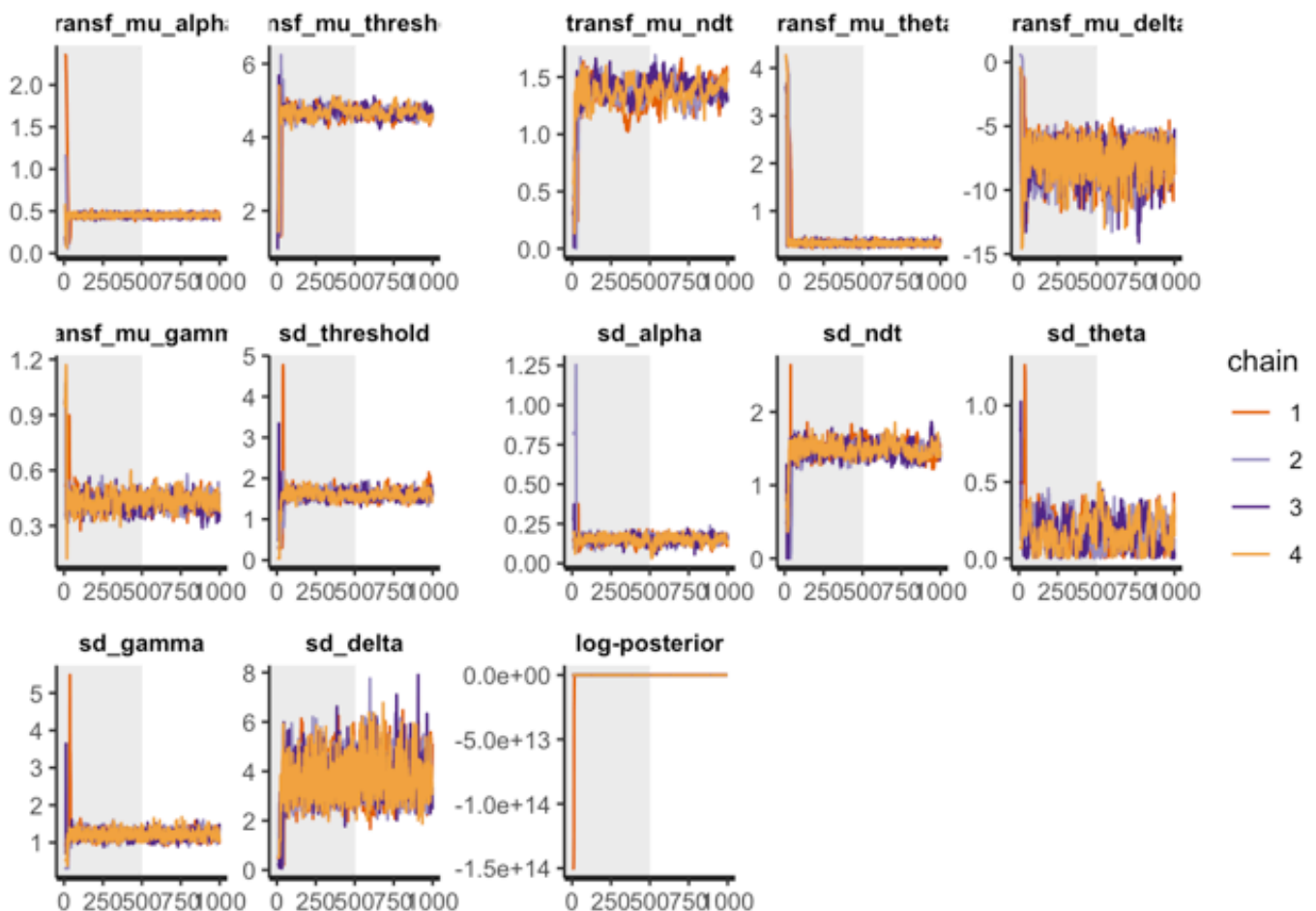
```
pairs(dsamples, pars = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt",
"transf_mu_theta",'transf_mu_delta', 'transf_mu_gamma', 'sd_threshold',"sd_alph
a","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta', "lp__"))
```



```
print(dsamples, pars = c("transf_mu_alpha","transf_mu_threshold","transf_mu_ndt",
"transf_mu_theta",'transf_mu_delta', 'transf_mu_gamma', 'sd_threshold',"sd_alph
a","sd_ndt", 'sd_theta', 'sd_gamma','sd_delta', "lp__"))
```

```
## Inference for Stan model: anon_model.
## 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%
## transf_mu_alpha        0.45    0.00   0.02      0.41      0.44      0.45
## transf_mu_threshold    4.67    0.01   0.14      4.41      4.58      4.67
## transf_mu_ndt          1.39    0.01   0.09      1.20      1.32      1.39
## transf_mu_theta        0.32    0.00   0.04      0.25      0.29      0.32
## transf_mu_delta       -7.65    0.05   1.39    -11.12     -8.39     -7.46
## transf_mu_gamma        0.43    0.00   0.04      0.35      0.40      0.43
## sd_threshold           1.60    0.01   0.11      1.41      1.52      1.59
## sd_alpha               0.15    0.00   0.02      0.10      0.14      0.15
## sd_ndt                 1.48    0.01   0.10      1.30      1.41      1.48
## sd_theta               0.16    0.01   0.10      0.01      0.08      0.16
## sd_gamma               1.20    0.01   0.12      0.99      1.12      1.18
## sd_delta               3.75    0.03   0.86      2.39      3.13      3.63
## lp__              -30294.68    1.71  28.26 -30347.42 -30314.57 -30294.58
##                         75%     97.5% n_eff Rhat
## transf_mu_alpha        0.46      0.49   796 1.01
## transf_mu_threshold    4.76      4.93   104 1.01
## transf_mu_ndt          1.45      1.57    56 1.08
## transf_mu_theta        0.34      0.40   894 1.01
## transf_mu_delta       -6.65     -5.54   725 1.01
## transf_mu_gamma        0.46      0.51   144 1.03
## sd_threshold           1.67      1.84   144 1.02
## sd_alpha               0.17      0.19   108 1.03
## sd_ndt                 1.55      1.69    78 1.03
## sd_theta               0.24      0.36    89 1.03
## sd_gamma               1.27      1.45   434 1.01
## sd_delta               4.25      5.84   616 1.01
## lp__              -30275.39 -30240.09   274 1.00
##
## Samples were drawn using NUTS(diag_e) at Mon Jan  8 13:09:01 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).
```

```r
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
```

```r
  u_theta",'transf_mu_delta', '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