

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

```
##### 0 - safe choice A, 1 - risky choice B #####
```

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

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

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

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

```
dat <- dat %>%
  filter(test_part == 'cs' | test_part == 'sc')
```

```
dat <- dat %>%
  mutate(
    oa_complex = ifelse(test_part == 'cs', 1, -1),
    evd = evd * oa_complex,
    sdd = sdd * oa_complex,
    chose_complex = ifelse((oa_complex == 1 & cho == 1) | (oa_complex == -1 & cho
== -1), 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
```

```

library(dplyr)
library(stringr)

df <- dat %>%
  # Swap values if oa_condition is not 0
  rowwise() %>%
  mutate(
    oc1 = if_else(test_part == 'sc', O_B1, O_A1),
    oc2 = if_else(test_part == 'sc', O_B2, O_A2),
    pc1 = if_else(test_part == 'sc', P_B1, P_A1),
    pc2 = if_else(test_part == 'sc', P_B2, P_A2),
    os1 = if_else(test_part == 'sc', O_A1, O_B1),
    os2 = if_else(test_part == 'sc', O_A2, O_B2),
    ps1 = if_else(test_part == 'sc', P_A1, P_B1),
    ps2 = if_else(test_part == 'sc', P_A2, P_B2),
  ) %>%
  ungroup()

```

```

df <- df %>%
  mutate(index1 = as.numeric(ifelse(oc1>oc2, 1, -1)) ,
         index2 = as.numeric(ifelse(os1>os2, 1, -1)),)

df <- df %>%
  # Swap values if oa_condition is not 0
  rowwise() %>%
  mutate(
    oc3 = if_else(index1 == 1, oc1, oc2),
    oc4 = if_else(index1 == 1, oc2, oc1),
    pc3 = if_else(index1 == 1, pc1, pc2),
    pc4 = if_else(index1 == 1, pc2, pc1),
    os3 = if_else(index2 == 1, os1, os2),
    os4 = if_else(index2 == 1, os2, os1),
    ps3 = if_else(index2 == 1, ps1, ps2),
    ps4 = if_else(index2 == 1, ps2, ps1),
  ) %>%
  ungroup()

```

```

dataList = list(cho = df$chose_complex, rt = df$rt, participant = df$tid, N=nrow(df),
  L = length(tids),
    oc = as.matrix(df[, c("oc3", "oc4")]),
    os = as.matrix(df[, c("os3", "os4")]),
    pc = as.matrix(df[, c("pc3", "pc4")]),
    ps = as.matrix(df[, c("ps3", "ps4")]),
    starting_point = 0.5
  )

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_sbj',
  'gamma_sbj', 'delta_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 = 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)
}

```

```

m <- stan_model("EU_prob.stan")

```

```

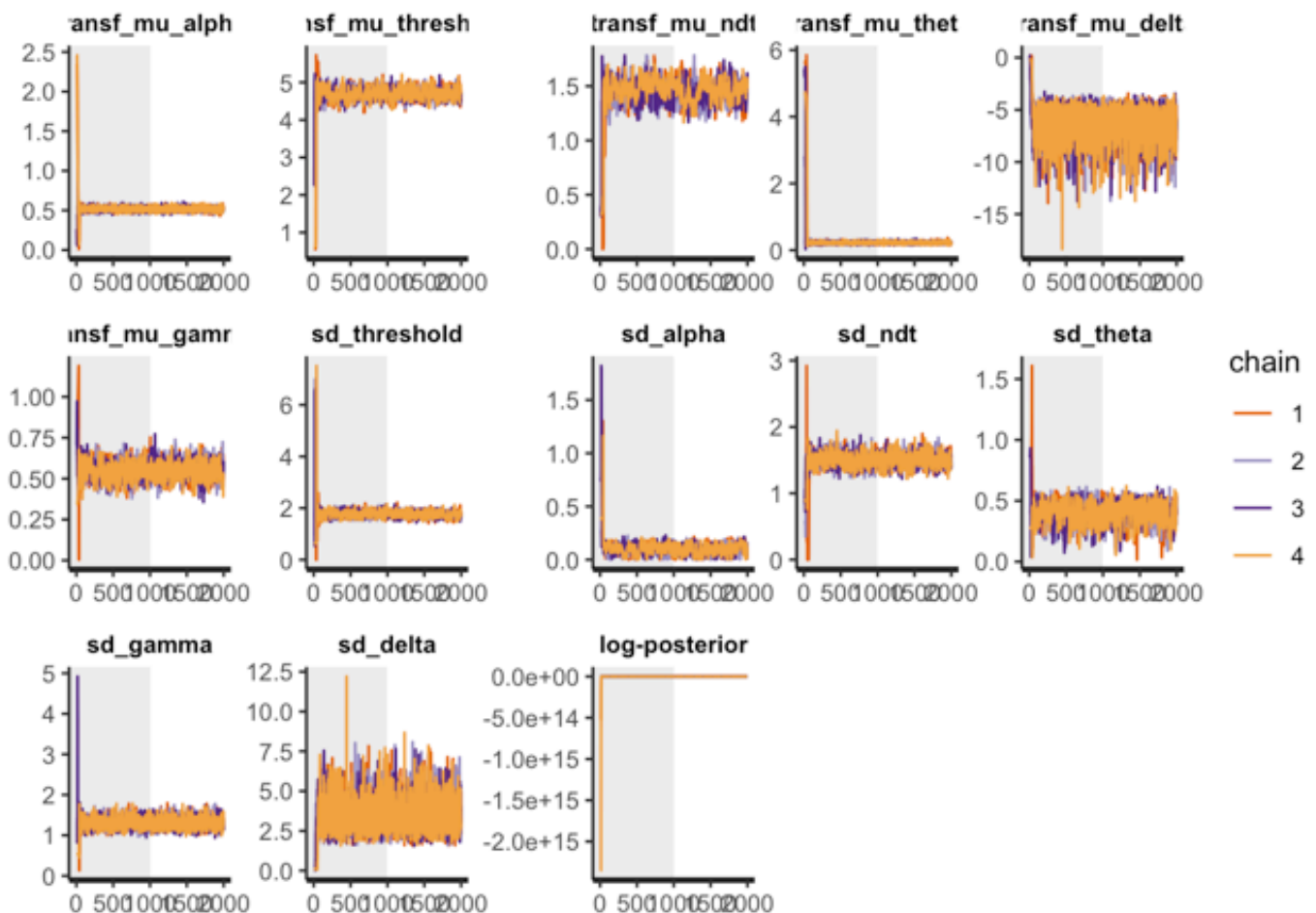
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -mmacosx-version-min=10.13 -I"/Library/Frameworks/R.framework/Resources/i
nclude" -DNDEBUG -I"/Library/Frameworks/R.framework/Versions/4.2/Resources/libra
ry/Rcpp/include/" -I"/Library/Frameworks/R.framework/Versions/4.2/Resources/libra
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/Framework
s/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

```

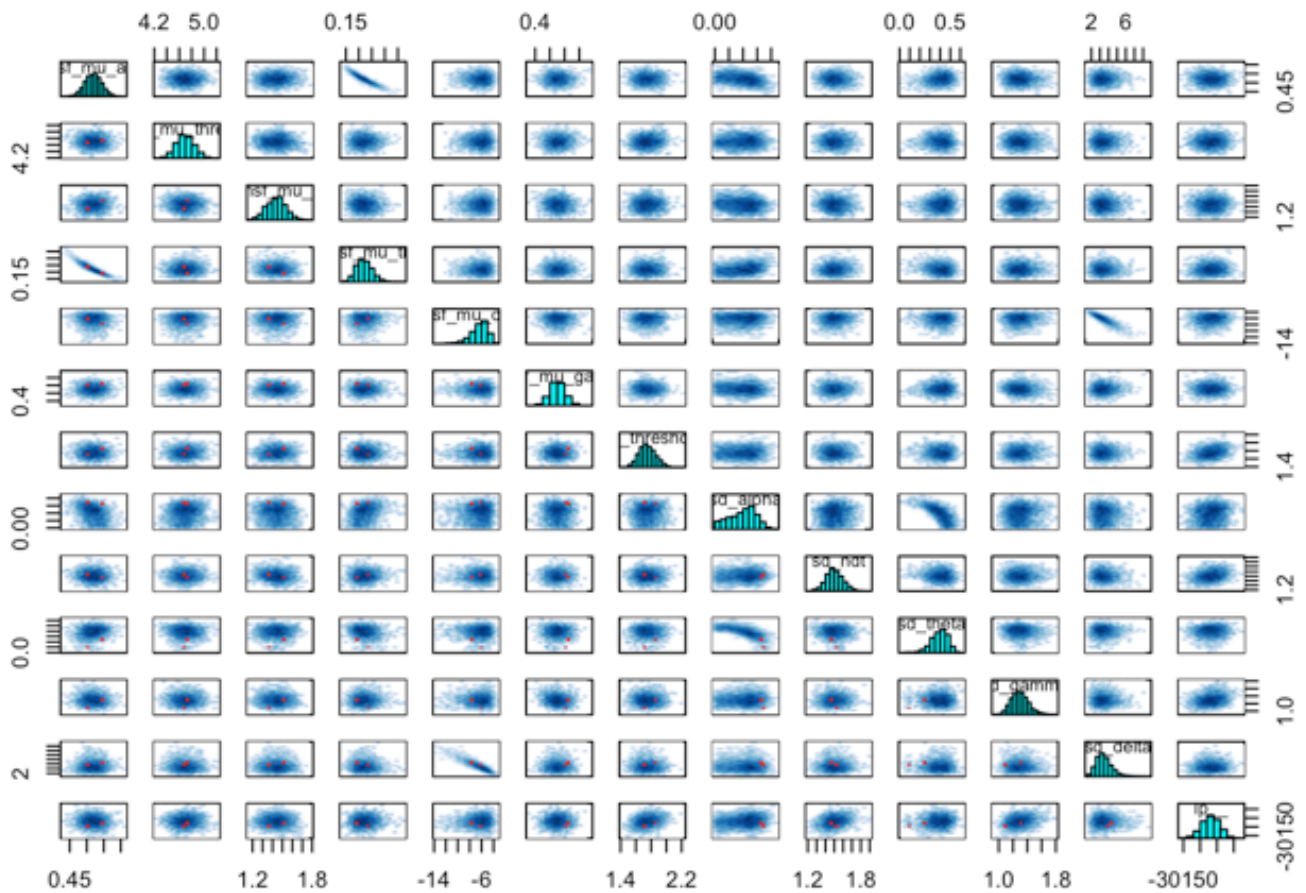
```
dsamples <- sampling(m,
  data=dataList,
  pars=parameters,
  iter=2000,
  chains=4, #If not specified, gives random inits
  init = initFunc(4),
  warmup = 1000, # Stands for burn-in; Default = iter/2
  seed = 12,
  refresh = 0

)
```

```
#"transf_mu_alpha", "transf_mu_threshold", "transf_mu_ndt", "transf_mu_theta", 'transf_mu_delta', 'transf_mu_gamma', 'sd_threshold', "sd_alpha", "sd_ndt", 'sd_theta', 'sd_gamma', 'sd_delta', "alpha_sbj", "threshold_sbj", "ndt_sbj", 'theta_sbj', 'gamma_sbj', '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', "sd_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=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      0.52    0.00  0.02      0.47      0.50      0.52
## transf_mu_threshold    4.70    0.01  0.14      4.42      4.60      4.70
## transf_mu_ndt         1.44    0.01  0.10      1.25      1.37      1.44
## transf_mu_theta       0.22    0.00  0.03      0.17      0.20      0.22
## transf_mu_delta      -6.43    0.03  1.47     -9.97     -7.21     -6.19
## transf_mu_gamma       0.55    0.00  0.05      0.45      0.52      0.55
## sd_threshold         1.75    0.01  0.11      1.55      1.67      1.74
## sd_alpha            0.10    0.00  0.05      0.01      0.06      0.10
## sd_ndt             1.50    0.00  0.10      1.32      1.44      1.50
## sd_theta           0.39    0.01  0.09      0.20      0.34      0.40
## sd_gamma           1.30    0.00  0.12      1.08      1.21      1.29
## sd_delta           3.49    0.02  0.94      2.02      2.84      3.36
## lp__              -30068.77    0.99 26.44 -30122.27 -30086.40 -30068.86
##               75%      97.5% n_eff Rhat
## transf_mu_alpha      0.53      0.57 1372 1.01
## transf_mu_threshold    4.79      4.98  265 1.01
## transf_mu_ndt         1.50      1.63  165 1.01
## transf_mu_theta       0.24      0.29 1627 1.00
## transf_mu_delta      -5.39     -4.26 1988 1.00
## transf_mu_gamma       0.59      0.66  258 1.03
## sd_threshold         1.82      1.97  431 1.01
## sd_alpha            0.13      0.18  104 1.05
## sd_ndt             1.56      1.70  478 1.01
## sd_theta           0.45      0.53  158 1.02
## sd_gamma           1.37      1.57 1046 1.00
## sd_delta           4.00      5.77 2021 1.00
## lp__              -30050.69 -30019.26  708 1.00
##
## Samples were drawn using NUTS(diag_e) at Mon Jan  8 03:58:12 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', '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", size = 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

