Code

For information on the OSARI task used in this project, please see the following paper:

He, J. L., Hirst, R. J., Puri, R., Coxon, J., Byblow, W., Hinder, M., Skippen, P., Matzke, D., Heathcote, A., Wadsley, C. G., Silk, T., Hyde, C., Parmar, D., Pedapati, E., Gilbert, D. L., Huddleston, D. A., Mostofsky, S., Leunissen, I., MacDonald, H. J., Chowdhury, N. S., … Puts, N. A. J. (2022). OSARI, an open-source anticipated response inhibition task. Behavior Research Methods, 54, 1530-1540. https://doi.org/10.3758/s13428-021-01680-9

For the response time modeling of the data, we used the exgSS model in the Dynamic Models of Choice (DMC) software. To learn more about the DMC material, please see: https://osf.io/pbwx8/wiki/home/ for the Wiki. The code is available on their GitHub: https://github.com/humanfactors/dmc?tab=readme-ov-file
All credit for the creation of this code goes to Andrew Heathcote and his team (including Michael Wilson, Brandon Turner, Scott Brown, Dora Matzke, Yishin Lin, Luke Strickland, Angus Reynolds, and Matthew Gretton).

In our analyses, we utilized the Rethinking package for RStudio, that was created by Richard McElreath. Please see his Rethinking GitHub here: https://github.com/rmcelreath/rethinking

Analysis Code (Created by Joseph Houpt and Bryanna Scheuler; picks up after DMC code for exgSS model)

```
#View the parameters after DMC fitting

library(dplyr)

require(ggplot2)

load("osari_posterior.RData")


group_by_week <- function(dat, parameters) {

  weeks <- 1:length(dat[[1]])

  subjects <- 1:length(dat)

  samp_dim <- dim(dat[[1]][[1]]$theta[,1,])
```

```
  out_df <- matrix(NA, 0, 3 + length(parameters))
  colnames(out_df) <- c("subject", "week", "sample", parameters)
  for (wk in weeks) {
   for (sj in subjects) {
    out_mat <- matrix(NA, prod(samp_dim), 0)
    for (parameter in parameters) {
     out_mat <- cbind(out_mat, c(dat[[sj]][[wk]]$theta[,parameter,1:samp_dim[2]]))
    }
    out_df <- rbind(out_df, cbind(sj, wk, 1:prod(samp_dim), out_mat))
   }
  }
  out_df <- data.frame(out_df)
  out_df$subject <- factor(out_df$subject)
  out_df$week <- factor(out_df$week, ordered=TRUE)
  return(out_df)
}

parameters <- c("muS", "tauS","sigmaS", "mu.true", "tau.true", "sigma.true")
beests_df <- group_by_week(samples_subject_week, parameters)

beests_subj_means <- beests_df %>%
  group_by(subject, week) %>%
summarise(muS = mean(muS), tauS=mean(tauS), sigmaS = mean(sigmaS), mu.true=mean(mu.true),
tau.true=mean(tau.true), sigma.true=mean(sigma.true))

beests_week_means <- beests_df %>%
  group_by(week, sample) %>%
summarise(muS = mean(muS), tauS=mean(tauS), sigmaS = mean(sigmaS), mu.true=mean(mu.true),
tau.true=mean(tau.true), sigma.true=mean(sigma.true))
```

```
#Combine to make SSRT for week group means
beests_week_means$SSRT <- with(beests_week_means, muS + tauS)

#Combine mu and tau to make SSRT for individuals
beests_subj_means$SSRT <- with(beests_subj_means, muS + tauS)

save(beests_subj_means, file = "BeestsSubjMeans.RData")

##SSRT Plot
png("posterior_SSRT.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=SSRT)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()

##Go Trial Mu Plot
png("posterior_muGo.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=mu.true)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()

##Go Trial Sigma Plot
png("posterior_sigmaGo.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=sigma.true)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()
```

```
##Go Trial Tau Plot
png("posterior_tauGo.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=tau.true)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()

##Stop Trial Mu Plot
png("posterior_muS.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=muS)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()

##Stop Trial Sigma Plot
png("posterior_sigmaS.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=sigmaS)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
dev.off()


##Stop Trial Tau Plot
png("posterior_tauS.png", 500,500, res=100)
ggplot(beests_week_means, aes(x=week, y=tauS)) + geom_violin() +
  geom_point(data=beests_subj_means, aes(color=subject)) +
  geom_line(data= beests_subj_means, aes(x=week, group= subject, color =subject))
```

```
dev.off()



#Executive Control Model Comparison --- SSRTs


library(rethinking)

load("BeestsSubjMeans.RData")

SRdata <- read.csv("CleanSelfRep.csv")


ECdata <- SRdata[,c("Participant", "Week", "PSS_Total", "SF_EmoWB")]

#ECdata$Participant <- factor(ECdata$Participant, levels=1:5, labels=c("Cascade", "Glacier", "Harbor", "Horizon", "Meadow"))

beests_subj_means$subject <- factor(beests_subj_means$subject, levels=1:5, labels=c("Cascade", "Glacier", "Harbor", "Horizon", "Meadow"))


for (sj in unique(ECdata$Participant)) {

  if (!any(sj==levels(beests_subj_means$subject) )) {

    ECdata <- subset(ECdata, Participant != sj)

  }

}


ECdata$SSRT <- NA

subjects <- unique(ECdata$Participant)

weeks <- unique(ECdata$Week)

for (sj in subjects) {

  for (wk in weeks) {

      ECdata[ECdata$Participant==sj & ECdata$Week == wk, "SSRT"] <- beests_subj_means$SSRT[beests_subj_means$subject==sj & beests_subj_means$week == wk]

  }

}
```

```
ECmodData <- list(
  part = as.integer(as.factor(ECdata$Participant)),
  Week3 = 1*(ECdata$Week == 3),
  Week2 = 1*(ECdata$Week == 2),
  PSS = (ECdata$PSS_Total)/50, #scaled to percent of maximum
  EmoWB = (ECdata$SF_EmoWB)/100, #scaled to percent of maximum
  SSRT = ECdata$SSRT
  )

#Model including stress and time
m_StressTimeEC <- ulam(
  alist(
    ## Time -> Exec. Con. <- Stress
    #distribution for EC parameter
    SSRT ~ dnorm(mu_SSRT, sigma_SSRT),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 ar zero) plus changes for wk 2 and wk 3
    mu_SSRT <- p_SSRT[part] + p_SSRTwk2[part]*Week2 + p_SSRTwk3[part]*Week3 + b_PSS*PSS +
b_EmoWB*EmoWB,

    #multivariate normal priors where individuals can experience different changes per week
    c(p_SSRT,p_SSRTwk2, p_SSRTwk3)[part] ~ multi_normal( c(a_SSRT,b1_SSRTwk, b2_SSRTwk) , Rho_SSRT
, sigma_indSSRT),

    #Matzke et al 2021 for SSRT priors
    a_SSRT ~ normal(.169,.06),
    b1_SSRTwk ~ normal(0,.1),
```

```
    b2_SSRTwk ~ normal(0,.1),

    #Priors for stress
    b_PSS ~ normal(0,.1),
    b_EmoWB ~ normal(0,.1),

    sigma_SSRT ~ exponential(1),
    sigma_indSSRT ~ exponential(1),
    Rho_SSRT ~ lkj_corr(2),

    ##Time -> Stress
    PSS ~ dnorm(mu_PSS, sigma_distPSS),
    EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),

    #Set up participant change over time
    mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,
    mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,

    c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS ,
sigma_PSS),
    c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) ,
Rho_EMO , sigma_EMO),

    a_PSS ~ normal(.5,.1),
    b1_PSSwk ~ normal(0,.1),
    b2_PSSwk ~ normal(0,.1),

    a_EMO ~ normal(.5, .1),
    b1_EMOwk ~ normal(0,.1),
```

```
  b2_EMOwk ~ normal(0,.1),

  sigma_PSS ~ exponential(1),
  sigma_EMO ~ exponential(1),

  Rho_PSS ~ lkj_corr(2),
  Rho_EMO ~ lkj_corr(2),

  sigma_EmoWB ~ exponential(1),
  sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

precis(m_StressTimeEC, depth=3)

#Full Mediation Model     Time -> Stress -> Exec. Con.
m_FullMedEC <- ulam(
  alist(
    ## Stress -> Exec. Con.
    #distribution for EC parameter
    SSRT ~ dnorm(mu_SSRT, sigma_SSRT),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 ar zero) plus changes for wk 2 and wk 3
    mu_SSRT <- p_SSRT[part] + b_PSS*PSS + b_EmoWB*EmoWB,

    #Matzke et al 2021 for SSRT priors
    p_SSRT[part] ~ normal(a_SSRT, sigma_indSSRT),
    a_SSRT ~ normal(.169,.06),
```

```
    #Priors for stress
    b_PSS ~ normal(0,.1),
    b_EmoWB ~ normal(0,.1),

    sigma_SSRT ~ exponential(1),
    sigma_indSSRT ~ exponential(1),

    ##Time -> Stress
    PSS ~ dnorm(mu_PSS, sigma_distPSS),
    EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),

    #Set up participant change over time
    mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,
    mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,

    c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS ,
sigma_PSS),
    c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) ,
Rho_EMO , sigma_EMO),

    a_PSS ~ normal(.5,.1),
    b1_PSSwk ~ normal(0,.1),
    b2_PSSwk ~ normal(0,.1),

    a_EMO ~ normal(.5, .1),
    b1_EMOwk ~ normal(0,.1),
    b2_EMOwk ~ normal(0,.1),

    sigma_PSS ~ exponential(1),
```

```
    sigma_EMO ~ exponential(1),

    Rho_PSS ~ lkj_corr(2),
    Rho_EMO ~ lkj_corr(2),

    sigma_EmoWB ~ exponential(1),
    sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

precis(m_FullMedEC, depth=3)

#Model including only time (no mediation)
m_TimeEC <- ulam(
  alist(
    ## Time -> Exec. Con.
    #distribution for EC parameter
    SSRT ~ dnorm(mu_SSRT, sigma_SSRT),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 ar zero) plus changes for wk 2 and wk 3
    mu_SSRT <- p_SSRT[part] + p_SSRTwk2[part]*Week2 + p_SSRTwk3[part]*Week3,

    #multivariate normal priors where individuals can experience different changes per week
    c(p_SSRT,p_SSRTwk2, p_SSRTwk3)[part] ~ multi_normal( c(a_SSRT,b1_SSRTwk, b2_SSRTwk) ,
Rho_SSRT , sigma_indSSRT),

    #Matzke et al 2021 for SSRT priors
    a_SSRT ~ normal(.169,.06),
    b1_SSRTwk ~ normal(0,.1),
```

```
b2_SSRTwk ~ normal(0,.1),



sigma_SSRT ~ exponential(1),
sigma_indSSRT ~ exponential(1),
Rho_SSRT ~ lkj_corr(2),

##Time -> Stress
PSS ~ dnorm(mu_PSS, sigma_distPSS),
EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),

#Set up participant change over time
mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,
mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,

c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS , sigma_PSS),
c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) , Rho_EMO , sigma_EMO),

a_PSS ~ normal(.5,.1),
b1_PSSwk ~ normal(0,.1),
b2_PSSwk ~ normal(0,.1),

a_EMO ~ normal(.5, .1),
b1_EMOwk ~ normal(0,.1),
b2_EMOwk ~ normal(0,.1),

sigma_PSS ~ exponential(1),
```

```
    sigma_EMO ~ exponential(1),

    Rho_PSS ~ lkj_corr(2),
    Rho_EMO ~ lkj_corr(2),

    sigma_EmoWB ~ exponential(1),
    sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

 precis(m_TimeEC, depth=3)

#Compare the three models of Executive Control
compare(m_StressTimeEC, m_FullMedEC, m_TimeEC, func = WAIC)
plot(compare(m_StressTimeEC, m_FullMedEC, m_TimeEC, func = WAIC))
##POSTERIOR PREDICTIVE PLOTS

#How well did the StressTime Model approximate the posterior distribution?
mu1 <- link(m_StressTimeEC, data =ECmodData)
mu1_SSRT <- mu1$mu_SSRT  # Extract MuGo predictions
mu1_mean <- apply(mu1_SSRT, 2, mean)
mu1_PI <- apply(mu1_SSRT, 2, PI)
m1sim <- sim(m_StressTimeEC, n=1e4)
m1PIsim <- apply(m1sim, 2, PI)

#Plot it out
plot(mu1_mean ~ECmodData$SSRT, col = rangi2, ylim = range(mu1_PI),
   xlab = "Observed SSRT",  ylab = "Predicted SSRT")
abline(a=0, b=1, lty=2)
```

```
for(i in 1:length(ECmodData$SSRT)) {
  lines(rep(ECmodData$SSRT[i], 2), mu1_PI[, i], col = rangi2)}


#Executive Control Model Comparison --- MuGo
library(rethinking)
load("BeestsSubjMeans.RData")
SRdata <- read.csv("CleanSelfRep.csv")


ECdata <- SRdata[,c("Participant", "Week", "PSS_Total", "SF_EmoWB")]
#ECdata$Participant <- factor(ECdata$Participant, levels=1:5, labels=c("Cascade", "Glacier", "Harbor",
"Horizon", "Meadow"))
beests_subj_means$subject <- factor(beests_subj_means$subject, levels=1:5, labels=c("Cascade",
"Glacier", "Harbor", "Horizon", "Meadow"))


for (sj in unique(ECdata$Participant)) {
  if (!any(sj==levels(beests_subj_means$subject) )) {
    ECdata <- subset(ECdata, Participant != sj)
  }
}


ECdata$Mu <- NA
subjects <- unique(ECdata$Participant)
weeks <- unique(ECdata$Week)
for (sj in subjects) {
  for (wk in weeks) {
    ECdata[ECdata$Participant==sj & ECdata$Week == wk, "Mu"] <-
beests_subj_means$mu.true[beests_subj_means$subject==sj & beests_subj_means$week == wk]
  }
}
```

```
ECmodData <- list(
  part = as.integer(as.factor(ECdata$Participant)),
  Week3 = 1*(ECdata$Week == 3),
  Week2 = 1*(ECdata$Week == 2),
  PSS = (ECdata$PSS_Total)/50, #scaled to percent of maximum
  EmoWB = (ECdata$SF_EmoWB)/100, #scaled to percent of maximum
  MuGo = ECdata$Mu
)

#Model including stress and time
m_StressTimeEC <- ulam(
  alist(
    ## Time -> Exec. Con. <- Stress
    #distribution for EC parameter (using MuGo)
    MuGo ~ dnorm(mu_MuGo, sigma_MuGo),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 ar zero) plus changes for wk 2 and wk 3
    mu_MuGo <- p_MuGo[part] + p_MuGowk2[part]*Week2 + p_MuGowk3[part]*Week3 + b_PSS*PSS + 
b_EmoWB*EmoWB,

    #multivariate normal priors where individuals can experience different changes per week
    c(p_MuGo,p_MuGowk2, p_MuGowk3)[part] ~ multi_normal( c(a_MuGo,b1_MuGowk, b2_MuGowk) , 
Rho_MuGo, sigma_indMuGo),

    #Matzke et al 2021 for MuGo priors
    a_MuGo ~ normal(.824,.04),
    b1_MuGowk ~ normal(0,.1),
    b2_MuGowk ~ normal(0,.1),
```

```
    #Priors for stress

    b_PSS ~ normal(0,.1),

    b_EmoWB ~ normal(0,.1),


    sigma_MuGo ~ exponential(1),

    sigma_indMuGo ~ exponential(1),

    Rho_MuGo ~ lkj_corr(2),


    ##Time -> Stress

    PSS ~ dnorm(mu_PSS, sigma_distPSS),

    EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),


    #Set up participant change over time

    mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,

    mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,


    c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS ,
sigma_PSS),

    c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) ,
Rho_EMO , sigma_EMO),


    a_PSS ~ normal(.5,.1),

    b1_PSSwk ~ normal(0,.1),

    b2_PSSwk ~ normal(0,.1),


    a_EMO ~ normal(.5, .1),

    b1_EMOwk ~ normal(0,.1),

    b2_EMOwk ~ normal(0,.1),
```

```
    sigma_PSS ~ exponential(1),
    sigma_EMO ~ exponential(1),

    Rho_PSS ~ lkj_corr(2),
    Rho_EMO ~ lkj_corr(2),

    sigma_EmoWB ~ exponential(1),
    sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

precis(m_StressTimeEC, depth=3)

#Full Mediation Model     Time -> Stress -> Exec. Con.
m_FullMedEC <- ulam(
  alist(
    ## Stress -> Exec. Con.
    #distribution for EC parameter (using MuGo)
    MuGo ~ dnorm(mu_MuGo, sigma_MuGo),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 ar zero) plus changes for wk 2 and wk 3
    mu_MuGo <- p_MuGo[part] + b_PSS*PSS + b_EmoWB*EmoWB,



    #Matzke et al 2021 for MuGo priors
    p_MuGo[part] ~ normal(a_MuGo, sigma_indMuGo),
    a_MuGo ~ normal(.824,.04),
```

```
#Priors for stress
b_PSS ~ normal(0,.1),
b_EmoWB ~ normal(0,.1),

sigma_MuGo ~ exponential(1),
sigma_indMuGo ~ exponential(1),

##Time -> Stress
PSS ~ dnorm(mu_PSS, sigma_distPSS),
EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),

#Set up participant change over time
mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,
mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,

c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS ,
sigma_PSS),
c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) ,
Rho_EMO , sigma_EMO),

a_PSS ~ normal(.5,.1),
b1_PSSwk ~ normal(0,.1),
b2_PSSwk ~ normal(0,.1),

a_EMO ~ normal(.5, .1),
b1_EMOwk ~ normal(0,.1),
b2_EMOwk ~ normal(0,.1),

sigma_PSS ~ exponential(1),
```

```
    sigma_EMO ~ exponential(1),

    Rho_PSS ~ lkj_corr(2),
    Rho_EMO ~ lkj_corr(2),

    sigma_EmoWB ~ exponential(1),
    sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

precis(m_FullMedEC, depth=3)

#Model including only time
m_TimeEC <- ulam(
  alist(
    ## Time -> Exec. Con.
    #distribution for EC parameter (using MuGo)
    MuGo ~ dnorm(mu_MuGo, sigma_MuGo),
    #Set up participant change over time
    #Week 1 (where wk2 and wk3 are zero) plus changes for wk 2 and wk 3
    mu_MuGo <- p_MuGo[part] + p_MuGowk2[part]*Week2 + p_MuGowk3[part]*Week3,

    #multivariate normal priors where individuals can experience different changes per week
    c(p_MuGo,p_MuGowk2, p_MuGowk3)[part] ~ multi_normal( c(a_MuGo,b1_MuGowk, b2_MuGowk) ,
Rho_MuGo , sigma_indMuGo),

    #Matzke et al 2021 for MuGo priors
    a_MuGo ~ normal(.824,.04),
    b1_MuGowk ~ normal(0,.1),
```

```
    b2_MuGowk ~ normal(0,.1),

    sigma_MuGo ~ exponential(1),
    sigma_indMuGo ~ exponential(1),
    Rho_MuGo ~ lkj_corr(2),

    ##Time -> Stress
    PSS ~ dnorm(mu_PSS, sigma_distPSS),
    EmoWB ~ dnorm(mu_EmoWB, sigma_EmoWB),

    #Set up participant change over time
    mu_PSS <- p_PSS[part] + p_PSSwk2[part]*Week2 + p_PSSwk3[part]*Week3,
    mu_EmoWB <- p_EmoWB[part] + p_EMOwk2[part]*Week2 + p_EMOwk3[part]*Week3,

    c(p_PSS,p_PSSwk2, p_PSSwk3)[part] ~ multi_normal( c(a_PSS,b1_PSSwk, b2_PSSwk) , Rho_PSS ,
sigma_PSS),
    c(p_EmoWB,p_EMOwk2, p_EMOwk3)[part] ~ multi_normal( c(a_EMO,b1_EMOwk, b2_EMOwk) ,
Rho_EMO , sigma_EMO),

    a_PSS ~ normal(.5,.1),
    b1_PSSwk ~ normal(0,.1),
    b2_PSSwk ~ normal(0,.1),

    a_EMO ~ normal(.5, .1),
    b1_EMOwk ~ normal(0,.1),
    b2_EMOwk ~ normal(0,.1),

    sigma_PSS ~ exponential(1),
    sigma_EMO ~ exponential(1),
```

```
    Rho_PSS ~ lkj_corr(2),
    Rho_EMO ~ lkj_corr(2),

    sigma_EmoWB ~ exponential(1),
    sigma_distPSS ~ exponential(1)

  ), data = ECmodData, chains = 4, iter = 5000, log_lik = TRUE)

precis(m_TimeEC, depth=3)

#Compare the three models of Executive Control
compare(m_StressTimeEC, m_FullMedEC, m_TimeEC, func = WAIC)
plot(compare(m_StressTimeEC, m_FullMedEC, m_TimeEC, func = WAIC))

#How well did the StressTime Model approximate the posterior distribution?
mu1 <- link(m_StressTimeEC, data =ECmodData)
mu1_MuGo <- mu1$mu_MuGo  # Extract MuGo predictions
mu1_mean <- apply(mu1_MuGo, 2, mean)
mu1_PI <- apply(mu1_MuGo, 2, PI)
m1sim <- sim(m_StressTimeEC, n=1e4)
m1PIsim <- apply(m1sim, 2, PI)

#Plot it out
plot(mu1_mean ~ECmodData$MuGo, col = rangi2, ylim = range(mu1_PI),
     xlab = "Observed Mu(Go)",  ylab = "Predicted Mu(Go)")
abline(a=0, b=1, lty=2)
for(i in 1:length(ECmodData$MuGo)) {
  lines(rep(ECmodData$MuGo[i], 2), mu1_PI[, i], col = rangi2)}
```