

Task Switching Replication

Jon Bakdash & Laura Marusich

2021-12-09

```
#Read in and Check Raw Data
```

```
#Import current data
```

```
task_switching_raw <- read.csv(paste0(workingdir, "/task-switching-replication-recoded.csv"))  
head(task_switching_raw)
```

```
## participant session condition trialType posture blockNum trialNum switchTrialType  
## 1 1 1 1 experiment standing 1 1 buffer  
## 2 1 1 1 experiment standing 1 2 noswitch  
## 3 1 1 1 experiment standing 1 3 switch  
## 4 1 1 1 experiment standing 1 4 noswitch  
## 5 1 1 1 experiment standing 1 5 noswitch  
## 6 1 1 1 experiment standing 1 6 noswitch  
## congruantTrialType cueType shapeType shapeColor response correctResponse correct reactionTime  
## 1 incongruant solid square blue right left no 0.9088130  
## 2 incongruant solid square blue left left yes 0.5947349  
## 3 incongruant dashed square blue right right yes 0.7084870  
## 4 incongruant dashed square blue right right yes 0.5995200  
## 5 congruant dashed square yellow right right yes 0.4399409  
## 6 congruant dashed square yellow right right yes 0.3847258  
## date utcTime  
## 1 2021-11-10 10:22:00 1636561737  
## 2 2021-11-10 10:22:00 1636561744  
## 3 2021-11-10 10:22:00 1636561746  
## 4 2021-11-10 10:22:00 1636561748  
## 5 2021-11-10 10:22:00 1636561750  
## 6 2021-11-10 10:22:00 1636561752
```

```
#does every person have 392 trials?
```

```
ntrials_sub <- task_switching_raw %>%  
  group_by(participant) %>%  
  summarize(ntrials = n()) %>%  
  pull(ntrials)
```

```
all(ntrials_sub == 392)
```

```
## [1] TRUE
```

```
#does every block start with a buffer and have 49 trials?
```

```
task_switching_raw <- task_switching_raw %>%  
  mutate(condblock = paste0(posture, blockNum))
```

```
blocktrials <- task_switching_raw %>%  
  group_by(participant, condblock) %>%
```

```

summarize(ntrials = n(), firsttrial = first(switchTrialType))

## `summarise()` has grouped output by 'participant'. You can override using the `.groups` argument.
all(blocktrials$ntrials == 49)

## [1] TRUE
all(blocktrials$firsttrial == "buffer")

## [1] TRUE
#Clean Data
#Drop buffer trials
task_switching_raw2 <- task_switching_raw %>%
  filter(switchTrialType != "buffer")

#Recode Correct to 1 and Incorrect to 0
task_switching_raw2$correct_bin <- recode(task_switching_raw2$correct,
                                           "no" = 0,
                                           "yes" = 1)

#Calc overall acc by participant
ts_overall_acc <- task_switching_raw2 %>%
  group_by(participant) %>%
  summarize(Accuracy = mean(correct_bin))

#find participants with less than 80% accuracy
low_acc_subs <- ts_overall_acc %>% filter(Accuracy < 0.80) %>%
  select(participant)

#Drop participants w/<=80% acc
#8, 49, 44, 2, 15, 51

task_switching_raw3 <- task_switching_raw2 %>%
  filter(!(participant %in% low_acc_subs$participant))

#Calc mean Acc by participant and conditions (posture, con, switch)
#Narrow format
ts_acc_mean <- task_switching_raw3 %>%
  group_by(participant,
            posture,
            congruantTrialType,
            switchTrialType) %>%
  summarize(Accuracy = mean(correct_bin))

## `summarise()` has grouped output by 'participant', 'posture', 'congruantTrialType'. You can override
#Convert data to wide format
ts_acc_mean_wide <- ts_acc_mean %>%
  pivot_wider(names_from = c(posture,
                              congruantTrialType,
                              switchTrialType),
              values_from = Accuracy)

write.csv(ts_acc_mean_wide, file = "new_recoded_for_statview.csv", row.names = F)

```

```
ts_acc_mean <- data.frame(ts_acc_mean)
ts_acc_mean$posture <- as.factor(ts_acc_mean$posture)
ts_acc_mean$participant <- as.factor(ts_acc_mean$participant)
ts_acc_mean$congruantTrialType <- as.factor(ts_acc_mean$congruantTrialType)
ts_acc_mean$switchTrialType <- as.factor(ts_acc_mean$switchTrialType)
str(ts_acc_mean)

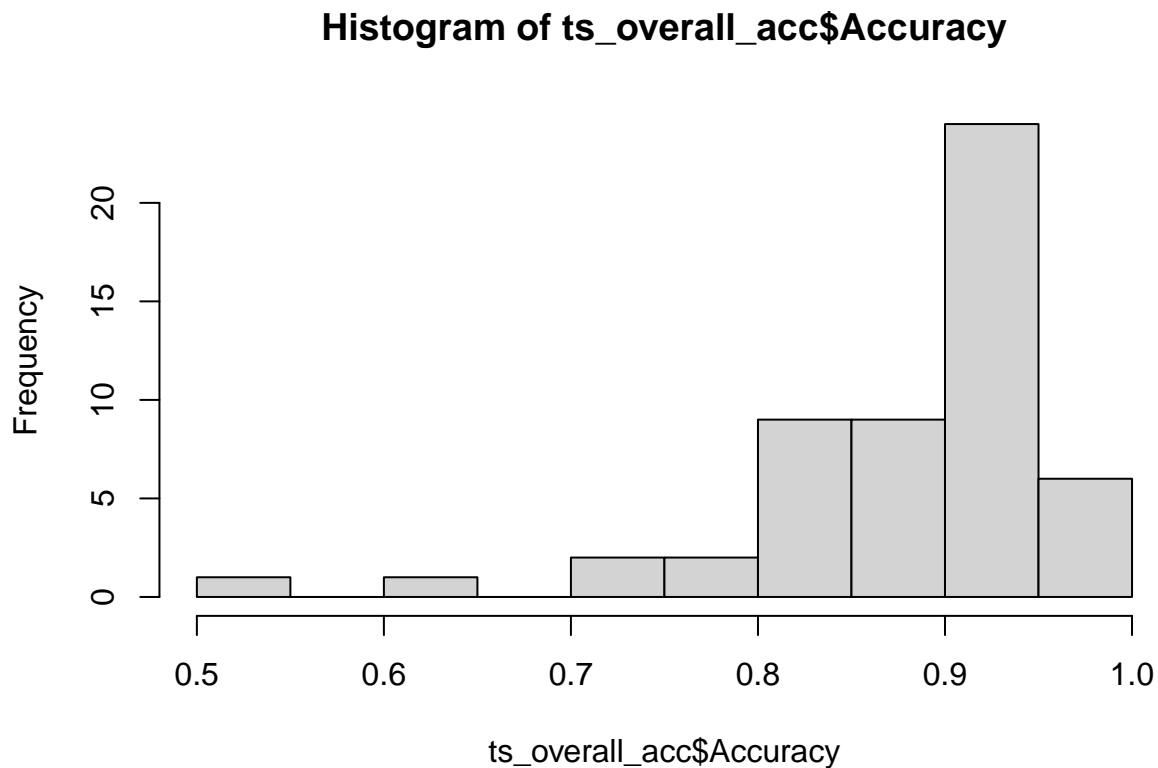
## 'data.frame': 384 obs. of 5 variables:
## $ participant : Factor w/ 48 levels "1","3","4","5",...: 1 1 1 1 1 1 1 1 2 2 ...
## $ posture : Factor w/ 2 levels "sitting","standing": 1 1 1 1 2 2 2 2 1 1 ...
## $ congruantTrialType: Factor w/ 2 levels "congruant","incongruant": 1 1 2 2 1 1 2 2 1 1 ...
## $ switchTrialType : Factor w/ 2 levels "noswitch","switch": 1 2 1 2 1 2 1 2 1 2 ...
## $ Accuracy : num 0.96 0.978 0.957 0.88 0.981 ...

#Total N = 48
length(unique(ts_acc_mean$participant))

## [1] 48
```

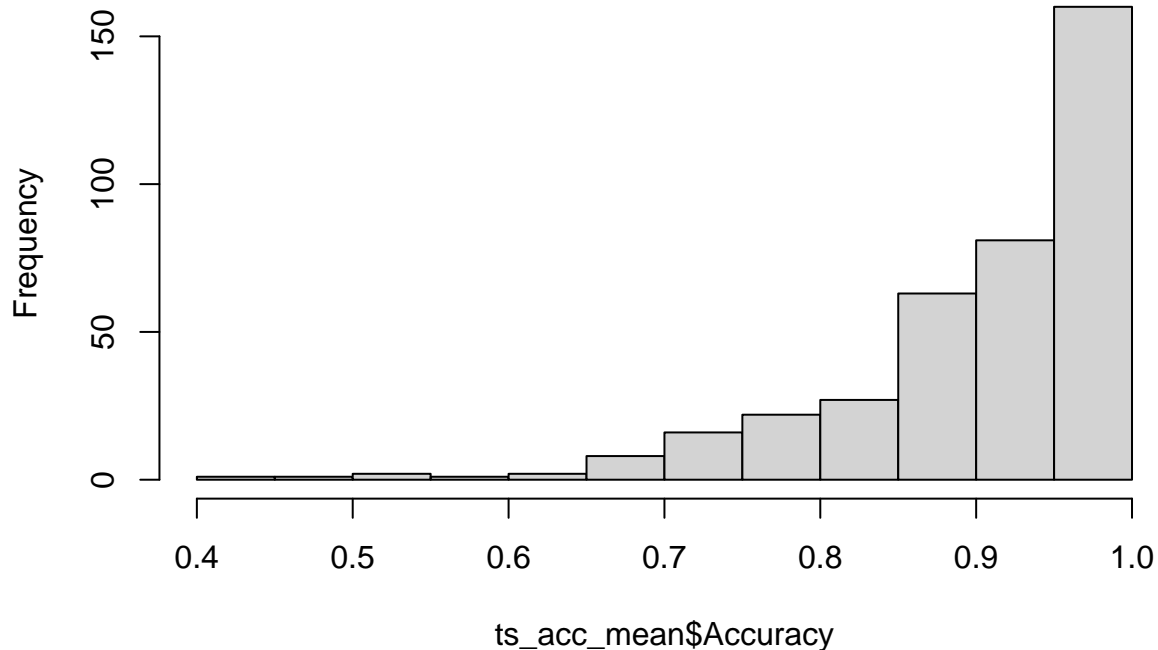
Plots and Analyses

```
#Accuracy by participant, before drops
hist(ts_overall_acc$Accuracy)
```



```
#Accuracy for all cells
hist(ts_acc_mean$Accuracy)
```

Histogram of ts_acc_mean\$Accuracy



```
exp1_anova <- ezANOVA(ts_acc_mean,
  dv = Accuracy,
  wid = participant,
  within = .(posture, congruentTrialType, switchTrialType),
  type = 3,
  detailed = TRUE,
  return_aov = T
)

#data.frame(exp1_anova$ANOVA)
output1 <- aovEffectSize(exp1_anova, effectSize = "pes")
output1 <- data.frame(output1$ANOVA)

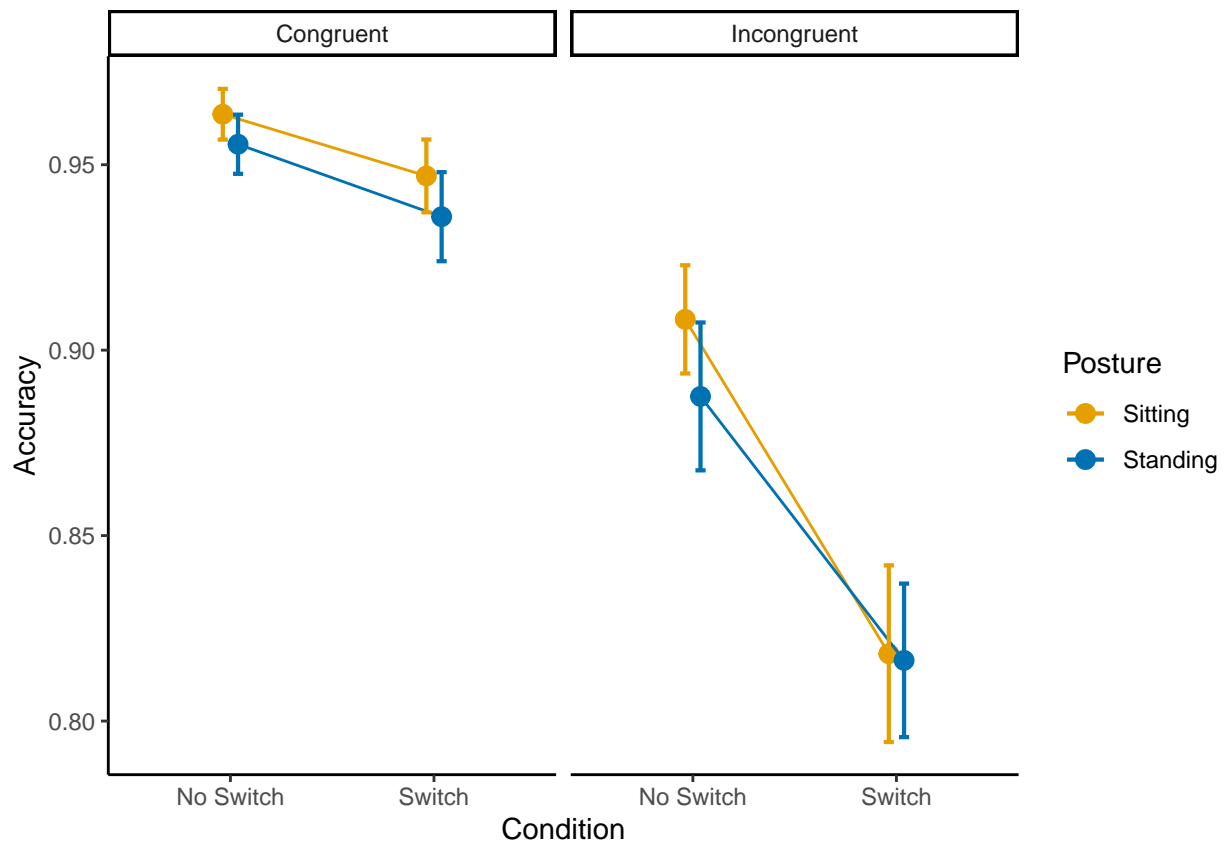
congruent.labs <- c("Congruent", "Incongruent")
names(congruent.labs) <- c("1", "2")

#make plot like Smith et al's
superbPlot(ts_acc_mean_wide,
  WSFactors = c("Condition(2)", "Congruent(2)", "Posture(2)"),
  variables = colnames(ts_acc_mean_wide)[2:9],
  errorbar = "SE",
  plotStyle = "line",
  factorOrder = c("Condition", "Posture", "Congruent"),
  adjustments = list(purpose = "difference"))+
  theme_classic()+
  facet_wrap(vars(Congruent), labeller = labeller(Congruent = congruent.labs)) +
  scale_x_discrete(labels=c("1" = "No Switch", "2" = "Switch"))+
```

```
scale_color_manual(values=c("#E69F00", "#0072B2"), labels = c("Sitting", "Standing")) +
labs(y = "Accuracy")
```

```
## superb::FYI: Here is how the within-subject variables are understood:
```

```
## Condition Congruent Posture variable
##      1      1      1 sitting_congruant_noswitch
##      2      1      1 sitting_congruant_switch
##      1      2      1 sitting_incongruant_noswitch
##      2      2      1 sitting_incongruant_switch
##      1      1      2 standing_congruant_noswitch
##      2      1      2 standing_congruant_switch
##      1      2      2 standing_incongruant_noswitch
##      2      2      2 standing_incongruant_switch
```



```
# write.csv(output, "output_ez.csv")
#importantly there was a significant condition [switch] × posture interaction,
#F(1, 29) = 5.54, p = .026, 2p = .16.
#posture:switchTrialType 1 52 6.705827e-03 0.15814085 2.205015e+00
#6.016033e-01 1.117445e-04
#p = 0.14
#GES = 0.07
# expl_anova.b <- aov(Accuracy ~
#                       posture*congruantTrialType*switchTrialType +
#                       Error(participant/posture*congruantTrialType*switchTrialType),
#                       ts_acc_mean)
# #str(ts_acc_mean)
```

```
# summary(exp1_anova.b)
# output <- anova_stats(exp1_anova.b)
# write.csv(output, "output_aov.csv")
```

```
#look at reaction time for correct trials?
ts_correct_only <- task_switching_raw3 %>%
  filter(correct_bin == 1)

ts_rt_mean <- ts_correct_only %>%
  group_by(participant,
            posture,
            congruantTrialType,
            switchTrialType) %>%
  summarize(mean_rt = mean(reactionTime))
```

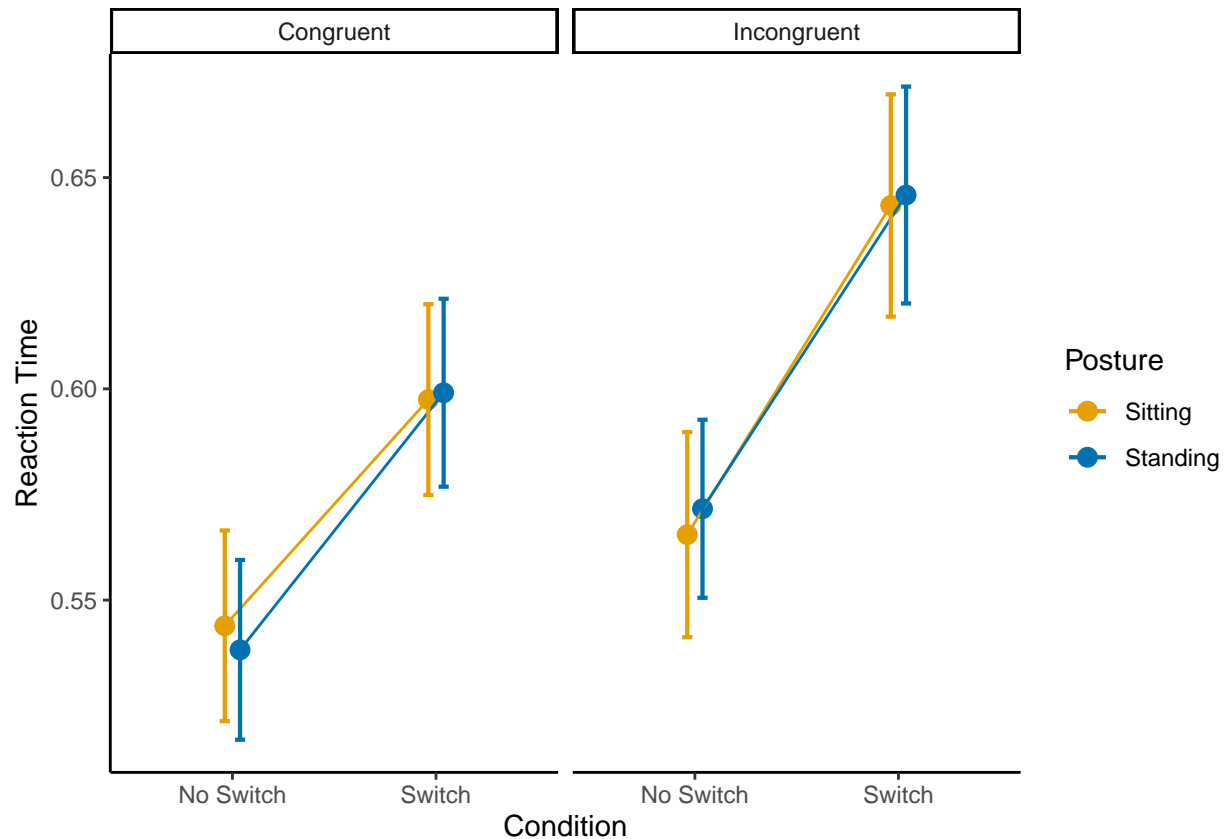
`summarise()` has grouped output by 'participant', 'posture', 'congruantTrialType'. You can override

```
#Convert data to wide format
ts_rt_mean_wide <- ts_rt_mean %>%
  pivot_wider(names_from = c(posture,
                              congruantTrialType,
                              switchTrialType),
              values_from = mean_rt)

superbPlot(ts_rt_mean_wide,
            WSFactors = c("Condition(2)", "Congruent(2)", "Posture(2)"),
            variables = colnames(ts_acc_mean_wide)[2:9],
            errorbar = "SE",
            plotStyle = "line",
            factorOrder = c("Condition", "Posture", "Congruent"),
            adjustments = list(purpose = "difference"))+
theme_classic()+
facet_wrap(vars(Congruent), labeller = labeller(Congruent = congruent.labs)) +
scale_x_discrete(labels=c("1" = "No Switch", "2" = "Switch"))+
scale_color_manual(values=c("#E69F00", "#0072B2"), labels = c("Sitting", "Standing")) +
labs(y = "Reaction Time")
```

superb::FYI: Here is how the within-subject variables are understood:

```
## Condition Congruent Posture variable
##          1          1          1 sitting_congruant_noswitch
##          2          1          1 sitting_congruant_switch
##          1          2          1 sitting_incongruant_noswitch
##          2          2          1 sitting_incongruant_switch
##          1          1          2 standing_congruant_noswitch
##          2          1          2 standing_congruant_switch
##          1          2          2 standing_incongruant_noswitch
##          2          2          2 standing_incongruant_switch
```



```
exp1_anova_rt <- ezANOVA(ts_rt_mean,
  dv = mean_rt,
  wid = participant,
  within = .(posture, congruantTrialType, switchTrialType),
  type = 3,
  detailed = TRUE,
  return_aov = T
)
```

```
## Warning: Converting "participant" to factor for ANOVA.
## Warning: Converting "posture" to factor for ANOVA.
## Warning: Converting "congruantTrialType" to factor for ANOVA.
## Warning: Converting "switchTrialType" to factor for ANOVA.
```

```
#data.frame(exp1_anova$ANOVA)
output_rt <- aovEffectSize(exp1_anova, effectSize = "pes")
output_rt <- data.frame(output_rt$ANOVA)
```