

# Smith Replication: Primary Data Analysis

Laura Marusich, Michael Reynolds, Jon Bakdash, and Emilie Caron

2022-11-07

## Session Information

```
# References for R packages
# install.packages(devtools)
# require(devtools)
# devtools::install_github("crsh/papaja")
# require(papaja)
#
# papaja::r_refs(file = "results/r-package-refs.bib") #<-- writes bib file w/refs

#Save package versions
#renv::snapshot()

sessionInfo()

## R version 4.2.1 (2022-06-23 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8 LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats      graphics  grDevices datasets  utils      methods    base
##
## other attached packages:
## [1] labeling_0.4.2      MBESS_4.9.2          highr_0.9            farver_2.1.1
## [5] ggstance_0.3.5      scales_1.2.1         rlang_1.0.6          cowplot_1.1.1
## [9] readxl_1.4.1        reshape2_1.4.4       superb_0.95.0        psychReport_3.0.2
## [13] apaTables_2.0.8     psych_2.2.9          BayesFactor_0.9.12-4.4 coda_0.19-4
## [17] ez_4.4-0            afex_1.2-0           lme4_1.1-31          Matrix_1.5-1
## [21] forcats_0.5.2       stringr_1.4.1        dplyr_1.0.10         purrr_0.3.5
## [25] readr_2.1.3         tidyr_1.2.1          tibble_3.1.8         ggplot2_3.4.0
## [29] tidyverse_1.3.2     plyr_1.8.7           rmarkdown_2.17       knitr_1.40
## [33] pacman_0.5.1
##
```

```
## loaded via a namespace (and not attached):
## [1] googledrive_2.0.0 minqa_1.2.5 colorspace_2.0-3 ellipsis_0.3.2
## [5] fs_1.5.2 rstudioapi_0.14 MatrixModels_0.5-1 fansi_1.0.3
## [9] mvtnorm_1.1-3 lubridate_1.8.0 xml2_1.3.3 splines_4.2.1
## [13] mnormt_2.1.1 jsonlite_1.8.3 nloptr_2.0.3 lsr_0.5.2
## [17] broom_1.0.1 dbplyr_2.2.1 shiny_1.7.3 compiler_4.2.1
## [21] httr_1.4.4 backports_1.4.1 assertthat_0.2.1 fastmap_1.1.0
## [25] gargle_1.2.1 cli_3.4.1 later_1.3.0 htmltools_0.5.3
## [29] tools_4.2.1 lmerTest_3.1-3 gtable_0.3.1 glue_1.6.2
## [33] Rcpp_1.0.9 carData_3.0-5 cellranger_1.1.0 vctrs_0.5.0
## [37] nlme_3.1-157 xfun_0.34 rbibutils_2.2.9 rvest_1.0.3
## [41] mime_0.12 lifecycle_1.0.3 renv_0.16.0 googlesheets4_1.0.1
## [45] MASS_7.3-57 hms_1.1.2 promises_1.2.0.1 shinyBS_0.61.1
## [49] parallel_4.2.1 yaml_2.3.6 pbapply_1.5-0 stringi_1.7.8
## [53] boot_1.3-28 Rdpack_2.4 pkgconfig_2.0.3 evaluate_0.17
## [57] lattice_0.20-45 tidyselect_1.2.0 magrittr_2.0.3 R6_2.5.1
## [61] generics_0.1.3 DBI_1.1.3 pillar_1.8.1 haven_2.5.1
## [65] foreign_0.8-82 withr_2.5.0 mgcv_1.8-40 abind_1.4-5
## [69] modelr_0.1.9 crayon_1.5.2 car_3.1-1 utf8_1.2.2
## [73] tzdb_0.3.0 grid_4.2.1 reprex_2.0.2 digest_0.6.30
## [77] xtable_1.8-4 httpuv_1.6.6 numDeriv_2016.8-1.1 munsell_0.5.0
```

## Unconventional Analysis:

Time-Outs ( $\geq 1500$  ms to respond) are treated as errors. The primary results are reported using this atypical approach to be consistent with the original paper by Smith et al. Near the end we also present a *Conventional* Analysis which drops the time-outs for Percent Error (PE) results.

## Experiment 1 - Stroop

### Import and clean data

```
stroop_files = list.files(path = "data/Experiment 1 Data/", full.names = T)
stroop_files = stroop_files[str_detect(stroop_files, pattern="(?.*SJ)(?.*.txt)")]

mergedStroopData <- ldply(stroop_files,
  read.delim,
  header=FALSE,
  stringsAsFactors = FALSE,
  sep = "") #for each item in the list apply the function read.delim

names(mergedStroopData) = c("sj",
  "cb",
  "blockNumber",
  "blockType",
  "trialNum",
  "congruency",
  "posture",
  "wordStim",
```

```

        "inkColour",
        "rt",
        "cResp",
        "resp",
        "ac")
#...remove problem subjects
#.. No subjects pre-identified as needing to be removed (see ethics protocol)

#...check number of observations per condition
ftable(posture+congruency~sj, mergedStroopData)

```

##	posture	SITTING			STANDING		
##	congruency	congruent	incongruent	neutral	congruent	incongruent	neutral
## sj							
## 1		60	60	60	60	60	60
## 2		60	60	60	60	60	60
## 3		60	60	60	60	60	60
## 4		60	60	60	60	60	60
## 5		60	60	60	60	60	60
## 6		60	60	60	60	60	60
## 7		60	60	60	60	60	60
## 8		60	60	60	60	60	60
## 9		60	60	60	60	60	60
## 10		60	60	60	60	60	60
## 11		60	60	60	60	60	60
## 12		60	60	60	60	60	60
## 13		60	60	60	60	60	60
## 14		60	60	60	60	60	60
## 15		60	60	60	60	60	60
## 16		60	60	60	60	60	60
## 17		60	60	60	60	60	60
## 18		60	60	60	60	60	60
## 19		60	60	60	60	60	60
## 20		60	60	60	60	60	60
## 21		60	60	60	60	60	60
## 22		120	120	120	120	120	120
## 24		60	60	60	60	60	60
## 25		60	60	60	60	60	60
## 26		60	60	60	60	60	60
## 27		60	60	60	60	60	60
## 28		60	60	60	60	60	60
## 29		60	60	60	60	60	60
## 30		60	60	60	60	60	60
## 31		60	60	60	60	60	60
## 32		60	60	60	60	60	60
## 33		60	60	60	60	60	60
## 34		60	60	60	60	60	60
## 35		60	60	60	60	60	60
## 36		60	60	60	60	60	60
## 37		60	60	60	60	60	60
## 38		60	60	60	60	60	60
## 39		60	60	60	60	60	60
## 40		60	60	60	60	60	60

## 41	60	60	60	60	60	60
## 42	60	60	60	60	60	60
## 43	60	60	60	60	60	60
## 44	60	60	60	60	60	60
## 45	60	60	60	60	60	60
## 46	60	60	60	60	60	60
## 47	60	60	60	60	60	60
## 48	60	60	60	60	60	60
## 49	60	60	60	60	60	60
## 50	60	60	60	60	60	60

```
ftable(blockType~sj, mergedStroopData)
```

##	blockType	experimental	practice
## sj			
## 1		288	72
## 2		288	72
## 3		288	72
## 4		288	72
## 5		288	72
## 6		288	72
## 7		288	72
## 8		288	72
## 9		288	72
## 10		288	72
## 11		288	72
## 12		288	72
## 13		288	72
## 14		288	72
## 15		288	72
## 16		288	72
## 17		288	72
## 18		288	72
## 19		288	72
## 20		288	72
## 21		288	72
## 22		576	144
## 24		288	72
## 25		288	72
## 26		288	72
## 27		288	72
## 28		288	72
## 29		288	72
## 30		288	72
## 31		288	72
## 32		288	72
## 33		288	72
## 34		288	72
## 35		288	72
## 36		288	72
## 37		288	72
## 38		288	72
## 39		288	72
## 40		288	72

```
## 41          288      72
## 42          288      72
## 43          288      72
## 44          288      72
## 45          288      72
## 46          288      72
## 47          288      72
## 48          288      72
## 49          288      72
## 50          288      72
```

*#...need to fix SJ - same one was used with two counterbalances*

```
mergedStroopData$sj = paste(mergedStroopData$sj, "_", mergedStroopData$cb, sep="")
```

*#...check for missing data*

```
mergedStroopData[!complete.cases(mergedStroopData),]
```

##	sj	cb	blockNumber	blockType	trialNum	congruency	posture	wordStim	inkColour	rt	cResp
## 646	10_2	2	8	experimental	34	incongruent	STANDING	RED	green	0	2
## 1081	12_2	2	1	practice	1	neutral	SITTING	XXXXX	green	0	2
## 1117	12_2	2	2	experimental	1	incongruent	SITTING	GREEN	red	0	1
## 1445	13_1	1	1	practice	5	incongruent	STANDING	GREEN	red	0	1
## 1801	14_2	2	1	practice	1	neutral	SITTING	XXX	green	0	2
## 2162	15_1	1	1	practice	2	congruent	STANDING	GREEN	green	0	2
## 2163	15_1	1	1	practice	3	neutral	STANDING	XXXXX	green	0	2
## 2615	16_2	2	3	experimental	23	incongruent	SITTING	RED	green	0	2
## 2720	16_2	2	6	practice	20	neutral	STANDING	XXX	red	0	1
## 2737	16_2	2	7	experimental	1	incongruent	STANDING	GREEN	red	0	1
## 2885	17_1	1	1	practice	5	incongruent	STANDING	RED	green	0	2
## 3360	18_2	2	4	experimental	12	incongruent	SITTING	GREEN	red	0	1
## 3457	18_2	2	7	experimental	1	incongruent	STANDING	RED	green	0	2
## 3601	19_1	1	1	practice	1	congruent	STANDING	RED	red	0	1
## 4370	20_2	2	2	experimental	14	neutral	SITTING	XXXXX	red	0	1
## 4505	20_2	2	6	practice	5	incongruent	STANDING	GREEN	red	0	1
## 4681	21_1	1	1	practice	1	congruent	STANDING	RED	red	0	1
## 4682	21_1	1	1	practice	2	neutral	STANDING	XXX	red	0	1
## 4692	21_1	1	1	practice	12	incongruent	STANDING	RED	green	0	2
## 5041	22_1	1	1	practice	1	incongruent	STANDING	GREEN	red	0	1
## 5042	22_1	1	1	practice	2	incongruent	STANDING	RED	green	0	2
## 5043	22_1	1	1	practice	3	neutral	STANDING	XXX	red	0	1
## 5422	22_2	2	1	practice	22	incongruent	SITTING	GREEN	red	0	1
## 5428	22_2	2	1	practice	28	incongruent	SITTING	RED	green	0	2
## 5501	22_2	2	3	experimental	29	neutral	SITTING	XXXXX	red	0	1
## 5530	22_2	2	4	experimental	22	incongruent	SITTING	RED	green	0	2
## 5533	22_2	2	4	experimental	25	incongruent	SITTING	RED	green	0	2
## 5608	22_2	2	6	practice	28	incongruent	STANDING	GREEN	red	0	1
## 5621	22_2	2	7	experimental	5	neutral	STANDING	XXXXX	green	0	2
## 5644	22_2	2	7	experimental	28	neutral	STANDING	XXX	green	0	2
## 5668	22_2	2	8	experimental	16	incongruent	STANDING	GREEN	red	0	1
## 5684	22_2	2	8	experimental	32	neutral	STANDING	XXXXX	red	0	1
## 5741	22_2	2	10	experimental	17	neutral	STANDING	XXX	green	0	2
## 5761	24_2	2	1	practice	1	congruent	SITTING	RED	red	0	1
## 5763	24_2	2	1	practice	3	congruent	SITTING	GREEN	green	0	2
## 5768	24_2	2	1	practice	8	neutral	SITTING	XXX	green	0	2

## 5770	24_2	2	1	practice	10	incongruent	SITTING	RED	green	0	2
## 5772	24_2	2	1	practice	12	congruent	SITTING	GREEN	green	0	2
## 5773	24_2	2	1	practice	13	neutral	SITTING	XXX	red	0	1
## 5775	24_2	2	1	practice	15	incongruent	SITTING	GREEN	red	0	1
## 5776	24_2	2	1	practice	16	neutral	SITTING	XXXXX	green	0	2
## 5797	24_2	2	2	experimental	1	neutral	SITTING	XXXXX	green	0	2
## 5798	24_2	2	2	experimental	2	incongruent	SITTING	RED	green	0	2
## 6018	24_2	2	8	experimental	6	congruent	STANDING	RED	red	0	1
## 6121	25_1	1	1	practice	1	neutral	STANDING	XXX	red	0	1
## 6482	26_2	2	1	practice	2	neutral	SITTING	XXXXX	red	0	1
## 6518	26_2	2	2	experimental	2	incongruent	SITTING	RED	green	0	2
## 6841	27_1	1	1	practice	1	congruent	STANDING	RED	red	0	1
## 6842	27_1	1	1	practice	2	neutral	STANDING	XXXXX	green	0	2
## 6843	27_1	1	1	practice	3	neutral	STANDING	XXXXX	red	0	1
## 7202	28_2	2	1	practice	2	incongruent	SITTING	GREEN	red	0	1
## 7921	3_1	1	1	practice	1	congruent	STANDING	GREEN	green	0	2
## 7957	3_1	1	2	experimental	1	neutral	STANDING	XXXXX	red	0	1
## 8858	31_1	1	7	experimental	2	congruent	SITTING	RED	red	0	1
## 9253	32_2	2	8	experimental	1	neutral	STANDING	XXX	green	0	2
## 9361	33_1	1	1	practice	1	congruent	STANDING	RED	red	0	1
## 9364	33_1	1	1	practice	4	incongruent	STANDING	GREEN	red	0	1
## 9386	33_1	1	1	practice	26	congruent	STANDING	GREEN	green	0	2
## 9390	33_1	1	1	practice	30	neutral	STANDING	XXX	red	0	1
## 9505	33_1	1	5	experimental	1	neutral	STANDING	XXXXX	red	0	1
## 10441	36_2	2	1	practice	1	congruent	SITTING	GREEN	green	0	2
## 10444	36_2	2	1	practice	4	incongruent	SITTING	GREEN	red	0	1
## 10447	36_2	2	1	practice	7	incongruent	SITTING	RED	green	0	2
## 10448	36_2	2	1	practice	8	neutral	SITTING	XXX	green	0	2
## 10535	36_2	2	3	experimental	23	incongruent	SITTING	GREEN	red	0	1
## 10639	36_2	2	6	practice	19	neutral	STANDING	XXXXX	green	0	2
## 10785	36_2	2	10	experimental	21	congruent	STANDING	GREEN	green	0	2
## 11294	38_2	2	4	experimental	26	incongruent	SITTING	RED	green	0	2
## 11387	38_2	2	7	experimental	11	neutral	STANDING	XXX	green	0	2
## 11418	38_2	2	8	experimental	6	incongruent	STANDING	RED	green	0	2
## 11421	38_2	2	8	experimental	9	incongruent	STANDING	GREEN	red	0	1
## 12202	4_2	2	9	experimental	34	incongruent	STANDING	GREEN	red	0	1
## 12241	40_2	2	1	practice	1	incongruent	SITTING	RED	green	0	2
## 13603	43_1	1	8	experimental	31	incongruent	SITTING	RED	green	0	2
## 13688	44_2	2	1	practice	8	incongruent	SITTING	RED	green	0	2
## 13689	44_2	2	1	practice	9	incongruent	SITTING	GREEN	red	0	1
## 13694	44_2	2	1	practice	14	incongruent	SITTING	GREEN	red	0	1
## 13695	44_2	2	1	practice	15	neutral	SITTING	XXXXX	red	0	1
## 13696	44_2	2	1	practice	16	incongruent	SITTING	RED	green	0	2
## 13702	44_2	2	1	practice	22	congruent	SITTING	GREEN	green	0	2
## 13709	44_2	2	1	practice	29	incongruent	SITTING	GREEN	red	0	1
## 13713	44_2	2	1	practice	33	incongruent	SITTING	RED	green	0	2
## 13715	44_2	2	1	practice	35	incongruent	SITTING	GREEN	red	0	1
## 13716	44_2	2	1	practice	36	congruent	SITTING	GREEN	green	0	2
## 13717	44_2	2	2	experimental	1	congruent	SITTING	RED	red	0	1
## 13721	44_2	2	2	experimental	5	incongruent	SITTING	GREEN	red	0	1
## 14041	45_1	1	1	practice	1	congruent	STANDING	GREEN	green	0	2
## 14379	45_1	1	10	experimental	15	neutral	SITTING	XXXXX	green	0	2
## 14864	47_1	1	3	experimental	32	congruent	STANDING	RED	red	0	1
## 14901	47_1	1	4	experimental	33	neutral	STANDING	XXX	green	0	2

## 14958	47_1	1	6	practice	18 incongruent	SITTING	RED	green	0	2
## 15121	48_2	2	1	practice	1 congruent	SITTING	GREEN	green	0	2
## 15842	5_1	1	1	practice	2 incongruent	STANDING	GREEN	red	0	1
## 15843	5_1	1	1	practice	3 congruent	STANDING	RED	red	0	1
## 15845	5_1	1	1	practice	5 congruent	STANDING	GREEN	green	0	2
## 15846	5_1	1	1	practice	6 incongruent	STANDING	RED	green	0	2
## 15847	5_1	1	1	practice	7 neutral	STANDING	XXX	green	0	2
## 16107	5_1	1	8	experimental	15 neutral	SITTING	XXX	red	0	1
## 16201	50_2	2	1	practice	1 incongruent	SITTING	GREEN	red	0	1
## 16565	6_2	2	1	practice	5 congruent	SITTING	RED	red	0	1
## 16957	7_1	1	2	experimental	1 neutral	STANDING	XXX	green	0	2
## 17282	8_2	2	1	practice	2 incongruent	SITTING	GREEN	red	0	1
## 17643	9_1	1	1	practice	3 congruent	STANDING	GREEN	green	0	2
## 17644	9_1	1	1	practice	4 incongruent	STANDING	RED	green	0	2
##	resp ac									
## 646	0	NA								
## 1081	0	NA								
## 1117	0	NA								
## 1445	0	NA								
## 1801	0	NA								
## 2162	0	NA								
## 2163	0	NA								
## 2615	0	NA								
## 2720	0	NA								
## 2737	0	NA								
## 2885	0	NA								
## 3360	0	NA								
## 3457	0	NA								
## 3601	0	NA								
## 4370	0	NA								
## 4505	0	NA								
## 4681	0	NA								
## 4682	0	NA								
## 4692	0	NA								
## 5041	0	NA								
## 5042	0	NA								
## 5043	0	NA								
## 5422	0	NA								
## 5428	0	NA								
## 5501	0	NA								
## 5530	0	NA								
## 5533	0	NA								
## 5608	0	NA								
## 5621	0	NA								
## 5644	0	NA								
## 5668	0	NA								
## 5684	0	NA								
## 5741	0	NA								
## 5761	0	NA								
## 5763	0	NA								
## 5768	0	NA								
## 5770	0	NA								
## 5772	0	NA								
## 5773	0	NA								

## 5775	0 NA
## 5776	0 NA
## 5797	0 NA
## 5798	0 NA
## 6018	0 NA
## 6121	0 NA
## 6482	0 NA
## 6518	0 NA
## 6841	0 NA
## 6842	0 NA
## 6843	0 NA
## 7202	0 NA
## 7921	0 NA
## 7957	0 NA
## 8858	0 NA
## 9253	0 NA
## 9361	0 NA
## 9364	0 NA
## 9386	0 NA
## 9390	0 NA
## 9505	0 NA
## 10441	0 NA
## 10444	0 NA
## 10447	0 NA
## 10448	0 NA
## 10535	0 NA
## 10639	0 NA
## 10785	0 NA
## 11294	0 NA
## 11387	0 NA
## 11418	0 NA
## 11421	0 NA
## 12202	0 NA
## 12241	0 NA
## 13603	0 NA
## 13688	0 NA
## 13689	0 NA
## 13694	0 NA
## 13695	0 NA
## 13696	0 NA
## 13702	0 NA
## 13709	0 NA
## 13713	0 NA
## 13715	0 NA
## 13716	0 NA
## 13717	0 NA
## 13721	0 NA
## 14041	0 NA
## 14379	0 NA
## 14864	0 NA
## 14901	0 NA
## 14958	0 NA
## 15121	0 NA
## 15842	0 NA



```
## 15843    0 NA
## 15845    0 NA
## 15846    0 NA
## 15847    0 NA
## 16107    0 NA
## 16201    0 NA
## 16565    0 NA
## 16957    0 NA
## 17282    0 NA
## 17643    0 NA
## 17644    0 NA
```

```
##...THERE IS MISSING DATA, BUT IT IS EXPECTED:
##...There are trials where the the experiment times out
##...can be identified as RT == 0
##...trials where a response was not made have an RT =0, resp=0 and ac = NA
```

```
##...get the number of time outs
dim(mergedStroopData[!complete.cases(mergedStroopData),])[1]
```

```
## [1] 104
```

```
##...how are the missing trials distributed???
timeOutStroopData = mergedStroopData[!complete.cases(mergedStroopData),]

ftable(posture~congruency, timeOutStroopData)
```

```
##           posture SITTING STANDING
## congruency
## congruent           10          14
## incongruent         29          18
## neutral             13          20
```

```
ftable(blockType~sj, timeOutStroopData)
```

```
##           blockType experimental practice
## sj
## 10_2           1           0
## 12_2           1           1
## 13_1           0           1
## 14_2           0           1
## 15_1           0           2
## 16_2           2           1
## 17_1           0           1
## 18_2           2           0
## 19_1           0           1
## 20_2           1           1
## 21_1           0           3
## 22_1           0           3
## 22_2           8           3
## 24_2           3           8
## 25_1           0           1
```

```
## 26_2          1      1
## 27_1          0      3
## 28_2          0      1
## 3_1           1      1
## 31_1          1      0
## 32_2          1      0
## 33_1          1      4
## 36_2          2      5
## 38_2          4      0
## 4_2           1      0
## 40_2          0      1
## 43_1          1      0
## 44_2          2     10
## 45_1          1      1
## 47_1          2      1
## 48_2          0      1
## 5_1           1      5
## 50_2          0      1
## 6_2           0      1
## 7_1           1      0
## 8_2           0      1
## 9_1           0      2
```

```
##...this code changes the "time-out" trials as errors
##...see Davoli et al.
```

```
mergedStroopData$ac[mergedStroopData$rt==0] = 0
```

```
##...remove practice trials
```

```
mergedStroopData <- mergedStroopData[!mergedStroopData$blockType=="practice",]
```

```
##...check that only experimental trials are left
```

```
unique(mergedStroopData$blockType)
```

```
## [1] "experimental"
```

```
totalStroopTrials = dim(mergedStroopData)[1]
```

```
observationDataStroop = data.frame(ftable(blockType~sj, mergedStroopData))[c(1,3)]
```

```
##...remove trials faster than 100ms
```

```
# mergedStroopData= mergedStroopData[!mergedStroopData$rt==0,] #...greater than 1500ms
```

```
mergedStroopData= mergedStroopData[!(mergedStroopData$rt<=100 & mergedStroopData$rt > 0),]
```

```
validStroopRTTrials = dim(mergedStroopData)[1]
```

```
observationDataStroop$validTrials = data.frame(ftable(blockType~sj, mergedStroopData))[c(3)]
```

```
print(paste("percent invalid trials = ",
            ((totalStroopTrials-validStroopRTTrials)/totalStroopTrials)*100))
```

```
## [1] "percent invalid trials = 0.006944444444444444"
```

```
#Drops "time-out" trials
```

```
mergedStroopData.conventional <- mergedStroopData %>%
  filter(rt != 0 & resp != 0)
```

```

write.csv(mergedStroopData, file = "results/exp1_merged_stroop_data.csv",
          row.names = F)

#Conventional analysis drops the time out trials, rather than treating as errors
write.table(mergedStroopData.conventional, file = "results/exp1_merged_stroop_data_conv.txt",
            row.names = F)

stroopCorrect = mergedStroopData[mergedStroopData$ac == 1, ]

# mergedDataSet = mergedDataSet[mergedDataSet$ac ==1,]
errorsRemoved = dim(stroopCorrect)[1] #...total remaining trials
observationDataStroop$correctTrials = data.frame(ftable(blockType~sj, stroopCorrect))[c(3)]
trimInfo = data.frame(totalStroopTrials, validStroopRTTrials, errorsRemoved)
head(trimInfo)

##      totalStroopTrials validStroopRTTrials errorsRemoved
## 1                14400                14399                13852

#...percent of error trials lost
print(paste("percent errors removed = ",
            (((validStroopRTTrials-errorsRemoved)/totalStroopTrials)*100)))

## [1] "percent errors removed = 3.79861111111111"

#####
#...CHECK 20% CRITERION
#####
observationDataStroop$percentLoss =
  ((observationDataStroop$Freq-observationDataStroop$correctTrials)/
   observationDataStroop$Freq)*100

observationDataStroop$percentLoss

## [1] 3.472222 1.736111 0.347222 6.944444 1.736111 4.166667 4.513889 1.736111
## [9] 1.041667 1.736111 1.736111 3.125000 2.083333 4.513889 1.736111 10.069444
## [17] 8.680556 0.347222 2.083333 4.513889 4.861111 1.388889 1.388889 3.472222
## [25] 16.666667 1.736111 15.972222 1.041667 0.000000 5.555556 2.083333 8.680556
## [33] 0.347222 3.125000 1.736111 2.083333 4.166667 2.777778 2.430556 7.638889
## [41] 1.041667 4.166667 2.777778 1.041667 5.555556 3.125000 10.069444 1.736111
## [49] 3.472222 3.819444

sum(observationDataStroop$percentLoss>20)

## [1] 0

#...None!

#...RUN TRIMMING PROCEDURE
tempList = pjRecursiveTrim2(stroopCorrect, #...dataset
                           "rt", #...dependent variables

```

```

        c("sj",
          "cb",
          "congruency",
          "posture")) #.independent variables

trimmedStroopData=tempList[[1]]
totalStroopN = tempList[[2]]
rejectedStroop = tempList[[3]]
percentTrimmedStroop = tempList[[4]]
NcellsStroop = tempList[[5]]

#...get the trimming info
trimOutputStroop= data.frame(totalStroopN, rejectedStroop,percentTrimmedStroop,NcellsStroop)
head(trimOutputStroop)

```

```

##   totalStroopN rejectedStroop percentTrimmedStroop NcellsStroop
## 1          13852           292           2.107999           300

```

```

stroopRT = plyr::ddply(trimmedStroopData,
                      .(sj, cb,congruency,posture),
                      summarise,
                      meanRT = mean(rt))

head(stroopRT)

```

```

##    sj cb congruency posture  meanRT
## 1 1_1 1   congruent  SITTING 471.6458
## 2 1_1 1   congruent  STANDING 400.0638
## 3 1_1 1 incongruent  SITTING 430.0455
## 4 1_1 1 incongruent  STANDING 439.8444
## 5 1_1 1    neutral  SITTING 454.5455
## 6 1_1 1    neutral  STANDING 408.4565

```

```

#...get error data
stroopPE = plyr::ddply(mergedStroopData,
                      .(sj, cb,congruency,posture),
                      summarise,
                      meanPE = 100 - (mean(ac)*100))

head(stroopPE)

```

```

##    sj cb congruency posture  meanPE
## 1 1_1 1   congruent  SITTING 0.000000
## 2 1_1 1   congruent  STANDING 0.000000
## 3 1_1 1 incongruent  SITTING 4.166667
## 4 1_1 1 incongruent  STANDING 6.250000
## 5 1_1 1    neutral  SITTING 6.250000
## 6 1_1 1    neutral  STANDING 4.166667

```

```
#...combine the RT and error data
stroopCombined = cbind(stroopRT,meanPE =stroopPE$meanPE)

head(stroopCombined)
```

```
##      sj cb congruency posture meanRT meanPE
## 1 1_1 1 congruent SITTING 471.6458 0.000000
## 2 1_1 1 congruent STANDING 400.0638 0.000000
## 3 1_1 1 incongruent SITTING 430.0455 4.166667
## 4 1_1 1 incongruent STANDING 439.8444 6.250000
## 5 1_1 1 neutral SITTING 454.5455 6.250000
## 6 1_1 1 neutral STANDING 408.4565 4.166667
```

```
#...set as factors
stroopCombined$sj = factor(stroopCombined$sj)
stroopCombined$cb = factor(stroopCombined$cb)
```

## Reaction time results

```
rtModelStroop <- ezANOVA(stroopCombined,
                        dv = .(meanRT),
                        wid=.(sj),
                        within=.(posture,congruency),
                        detailed=TRUE,
                        type=3,
                        return_aov=TRUE)
```

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

```
## Warning: Converting "congruency" to factor for ANOVA.
```

```
rtModelStroop$ANOVA
```

##	Effect	DFn	DFd	SSn	SSd	F	p	p<.05	ges
## 1	(Intercept)	1	49	6.530862e+07	1322013.63	2.420643e+03	2.233955e-43	*	0.9748634585
## 2	posture	1	49	8.221421e+02	156217.37	2.578776e-01	6.138604e-01		0.0004879807
## 3	congruency	2	98	7.093105e+04	154676.49	2.247026e+01	9.278220e-09	*	0.0404190166
## 4	posture:congruency	2	98	8.430066e+01	51054.35	8.090852e-02	9.223396e-01		0.0000500584

```
rtStroopMSE = rtModelStroop$ANOVA$SSd/rtModelStroop$ANOVA$DFd
```

```
#...print ANOVA in nice format
paste(rtModelStroop$ANOVA$Effect,": F(",
      rtModelStroop$ANOVA$DFn,
      ", ",
      rtModelStroop$ANOVA$DFd,
      ") = ",
      round(rtModelStroop$ANOVA$F,3),
      ", MSE = ",
```

```

round(rtStroopMSE,3),
", p = ",
round(rtModelStroop$ANOVA$p,3),
", partialEtaSq = ",
round(rtModelStroop$ANOVA$SSn/(rtModelStroop$ANOVA$SSn+rtModelStroop$ANOVA$SSd),4),
sep="")

```

```

## [1] "(Intercept): F(1, 49) = 2420.643, MSE = 26979.87, p = 0, partialEtaSq = 0.9802"
## [2] "posture: F(1, 49) = 0.258, MSE = 3188.11, p = 0.614, partialEtaSq = 0.0052"
## [3] "congruency: F(2, 98) = 22.47, MSE = 1578.332, p = 0, partialEtaSq = 0.3144"
## [4] "posture:congruency: F(2, 98) = 0.081, MSE = 520.963, p = 0.922, partialEtaSq = 0.0016"

```

*#...CALCULATE THE BAYES FACTORS FOR THE RT ANALYSIS*

```

stroopBF = stroopCombined
stroopBF$posture = factor(stroopBF$posture)
stroopBF$congruency = factor(stroopBF$congruency)
bfValues1 = anovaBF(meanRT~congruency*posture+sj,
                    data = stroopBF,
                    whichRandom = "sj",
                    method="laplace")
bfValues1

```

```

## Bayes factor analysis
## -----
## [1] congruency + sj : 35335703 ±NA%
## [2] posture + sj : 0.1461731 ±NA%
## [3] congruency + posture + sj : 5410998 ±NA%
## [4] congruency + posture + congruency:posture + sj : 346149.8 ±NA%
##
## Against denominator:
## meanRT ~ sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

*#...get the Bayes factor for the Null Interaction*

```
bfValues1[3]/bfValues1[4]
```

```

## Bayes factor analysis
## -----
## [1] congruency + posture + sj : 15.63195 ±NA%
##
## Against denominator:
## meanRT ~ congruency + posture + congruency:posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

```
1/(bfValues1[3]/bfValues1[4])
```

```

## Bayes factor analysis
## -----
## [1] congruency + posture + congruency:posture + sj : 0.06397154 ±NA%

```

```
##
## Against denominator:
##   meanRT ~ congruency + posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS

# Difference scores and paired t-tests
#... stroop effect (incongruent - congruent) FOR Standing
standingStroop = stroopCombined[stroopCombined$posture=="STANDING", ]
standingStroop = standingStroop[standingStroop$congruency!="neutral", ]
t.test(standingStroop$meanRT[standingStroop$congruency=="congruent"],
       standingStroop$meanRT[standingStroop$congruency=="incongruent"],
       paired=TRUE )

##
## Paired t-test
##
## data: standingStroop$meanRT[standingStroop$congruency == "congruent"] and standingStroop$meanRT[stan
## t = -4.3805, df = 49, p-value = 6.226e-05
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -53.81756 -19.96796
## sample estimates:
## mean difference
##      -36.89276

#... stroop effect (incongruent - congruent) FOR SITTING
sittingStroop = stroopCombined[stroopCombined$posture=="SITTING", ]
sittingStroop = sittingStroop[sittingStroop$congruency!="neutral", ]
t.test(sittingStroop$meanRT[sittingStroop$congruency=="congruent"],
       sittingStroop$meanRT[sittingStroop$congruency=="incongruent"],
       paired=TRUE )

##
## Paired t-test
##
## data: sittingStroop$meanRT[sittingStroop$congruency == "congruent"] and sittingStroop$meanRT[sitting
## t = -5.1209, df = 49, p-value = 5.104e-06
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -52.27703 -22.81052
## sample estimates:
## mean difference
##      -37.54377
```

## Percent error results

```
errModelStroop <- ezANOVA(stroopCombined,
                          dv = .(meanPE),
                          wid=.(sj),
```

```

within=.(posture, congruency),
detailed=TRUE,
type=3,
return_aov = TRUE)

```

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

```
## Warning: Converting "congruency" to factor for ANOVA.
```

```
errStroopMSE = errModelStroop$ANOVA$SSd/errModelStroop$ANOVA$DFd
```

```

exp1_ANOVA_acc_uncon <-
paste(errModelStroop$ANOVA$Effect,": F(",
      errModelStroop$ANOVA$DFn,
      ", ",
      errModelStroop$ANOVA$DFd,
      ") = ",
      round(errModelStroop$ANOVA$F,3),
      ", MSE = ",
      round(errStroopMSE,3),
      ", p = ",
      round(errModelStroop$ANOVA$p,3),
      ", partialEtaSq = ",
      round(errModelStroop$ANOVA$SSn/(errModelStroop$ANOVA$SSn+errModelStroop$ANOVA$SSd),4),
      sep="")
exp1_ANOVA_acc_uncon

```

```

## [1] "(Intercept): F(1, 49) = 57.526, MSE = 75.297, p = 0, partialEtaSq = 0.54"
## [2] "posture: F(1, 49) = 0.007, MSE = 16.562, p = 0.934, partialEtaSq = 1e-04"
## [3] "congruency: F(2, 98) = 11.598, MSE = 9.222, p = 0, partialEtaSq = 0.1914"
## [4] "posture:congruency: F(2, 98) = 1.59, MSE = 6.228, p = 0.209, partialEtaSq = 0.0314"

```

```
write.csv(exp1_ANOVA_acc_uncon, "results/exp1_ANOVA_PE_uncon.csv")
```

```
#BF for errors
```

```

bfValues1.error = anovaBF(meanPE~congruency*posture+sj,
                          data = stroopBF,
                          whichRandom = "sj",
                          method="laplace")

```

```
bfValues1.error
```

```
## Bayes factor analysis
```

```
## -----
```

```

## [1] congruency + sj : 776.6647 ±NA%
## [2] posture + sj : 0.1172445 ±NA%
## [3] congruency + posture + sj : 91.51836 ±NA%
## [4] congruency + posture + congruency:posture + sj : 13.90884 ±NA%
##

```

```
## Against denominator:
```

```
## meanPE ~ sj
```

```
## ---
```

```
## Bayes factor type: BFlinearModel, JZS
```



```
#...get the Bayes factor for the Null Interaction
bfValues1.error[3]/bfValues1.error[4]
```

```
## Bayes factor analysis
## -----
## [1] congruency + posture + sj : 6.579869 ±NA%
##
## Against denominator:
##   meanPE ~ congruency + posture + congruency:posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
1/(bfValues1.error[3]/bfValues1.error[4])
```

```
## Bayes factor analysis
## -----
## [1] congruency + posture + congruency:posture + sj : 0.1519787 ±NA%
##
## Against denominator:
##   meanPE ~ congruency + posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
#...ERRORs
#... stroop effect (incongruent - congruent) FOR Standing
t.test(standingStroop$meanPE[standingStroop$congruency=="congruent"],
       standingStroop$meanPE[standingStroop$congruency=="incongruent"],
       paired=TRUE )
```

```
##
## Paired t-test
##
## data: standingStroop$meanPE[standingStroop$congruency == "congruent"] and standingStroop$meanPE[standingStroop$congruency == "incongruent"]
## t = -2.0681, df = 49, p-value = 0.04393
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## -2.79325655 -0.04007678
## sample estimates:
## mean difference
## -1.416667
```

```
#... stroop effect (incongruent - congruent) FOR SITTING
t.test(sittingStroop$meanPE[sittingStroop$congruency=="congruent"],
       sittingStroop$meanPE[sittingStroop$congruency=="incongruent"],
       paired=TRUE )
```

```
##
## Paired t-test
##
## data: sittingStroop$meanPE[sittingStroop$congruency == "congruent"] and sittingStroop$meanPE[sittingStroop$congruency == "incongruent"]
## t = -4.6535, df = 49, p-value = 2.51e-05
```

```
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## -3.758593 -1.491407
## sample estimates:
## mean difference
## -2.625
```

## Make plots for Stroop

```
#...pull out summary statistics per condition averaged across subjects for graph
graphRT = describeBy(stroopCombined$meanRT,
                      list(stroopCombined$posture, stroopCombined$congruency),
                      mat=TRUE,
                      digits = 1)

graphPE = describeBy(stroopCombined$meanPE,
                      list(stroopCombined$posture, stroopCombined$congruency),
                      mat=TRUE,
                      digits = 1)

head(graphRT)
```

```
##      item  group1      group2 vars  n mean    sd median trimmed mad   min   max range skew
## X11     1  SITTING  congruent    1 50 450.8  56.3  443.0   446.5 45.5 348.3 598.3 250.0  0.7
## X12     2  STANDING  congruent    1 50 448.5  60.6  440.3   442.2 48.6 341.3 624.4 283.0  1.0
## X13     3  SITTING  incongruent    1 50 488.3  91.9  471.8   476.7 74.4 351.1 803.1 452.0  1.2
## X14     4  STANDING  incongruent    1 50 485.4 100.6  458.7   470.5 79.0 358.3 861.5 503.2  1.8
## X15     5  SITTING    neutral    1 50 465.6  66.3  456.6   460.1 49.7 357.8 702.6 344.7  1.0
## X16     6  STANDING    neutral    1 50 460.9  67.6  452.7   453.6 51.5 346.2 695.4 349.2  1.4
##      kurtosis   se
## X11         0.2  8.0
## X12         0.9  8.6
## X13         1.7 13.0
## X14         3.8 14.2
## X15         1.9  9.4
## X16         2.7  9.6
```

```
#...get rid of irrelevant columns
graphRT = graphRT[,c("group1", "group2", "mean", "se")]
graphPE = graphPE[,c("group1", "group2", "mean", "se")]

#...rename the variables
names(graphRT) = c("posture", "congruency", "mean", "se")
names(graphPE) = c("posture", "congruency", "mean", "se")

#...make sure posture is in UPPERCASE
graphRT$posture = str_to_upper(graphRT$posture)

#..calculate the within subjects confidence intervals based on loftus and masson
#..the confidence intervals are based on the interaction term.

inxn.rt.MSE = rtStroopMSE[4]
```

```

inxn.err.MSE = errStroopMSE[4]

graphRT$se = sqrt((inxn.rt.MSE)/length(unique(stroopCombined$sj)))
graphPE$se= sqrt((inxn.err.MSE)/length(unique(stroopCombined$sj)))

critT = qt(p=.025,df=length(unique(stroopCombined$sj))-2,lower.tail =FALSE)

#---add the min and max for the confidence intervals
graphRT$min = graphRT$mean - (graphRT$se*critT)
graphRT$max = graphRT$mean + (graphRT$se*critT)

####GET AC DATA FROM twoAnimalWordsPRPac.R
graphRT$ac = paste("(",format(round(graphPE$mean,digits=1),nsmall = 1),")",sep="")
head(graphRT)

##      posture congruency mean      se      min      max      ac
## X11  SITTING  congruent 450.8 3.227887 444.3099 457.2901 (2.6)
## X12  STANDING  congruent 448.5 3.227887 442.0099 454.9901 (3.3)
## X13  SITTING  incongruent 488.3 3.227887 481.8099 494.7901 (5.2)
## X14  STANDING  incongruent 485.4 3.227887 478.9099 491.8901 (4.7)
## X15  SITTING   neutral 465.6 3.227887 459.1099 472.0901 (3.7)
## X16  STANDING   neutral 460.9 3.227887 454.4099 467.3901 (3.4)

#...used for positioning the accuracy data on the graph
graphRT$vAdj = 25 #down
graphRT$vAdj[graphRT$congruency=="incongruent"]=25 #up
graphRT$hAdj = 0 #right
#graphRT$hAdj[graphRT$posture=="SITTING"]=-60 #left
graphRT$congruency = factor(graphRT$congruency,labels = c("Congruent","Incongruent","Neutral"))
graphRT$congruency = factor(graphRT$congruency,levels=c("Congruent","Neutral","Incongruent"))

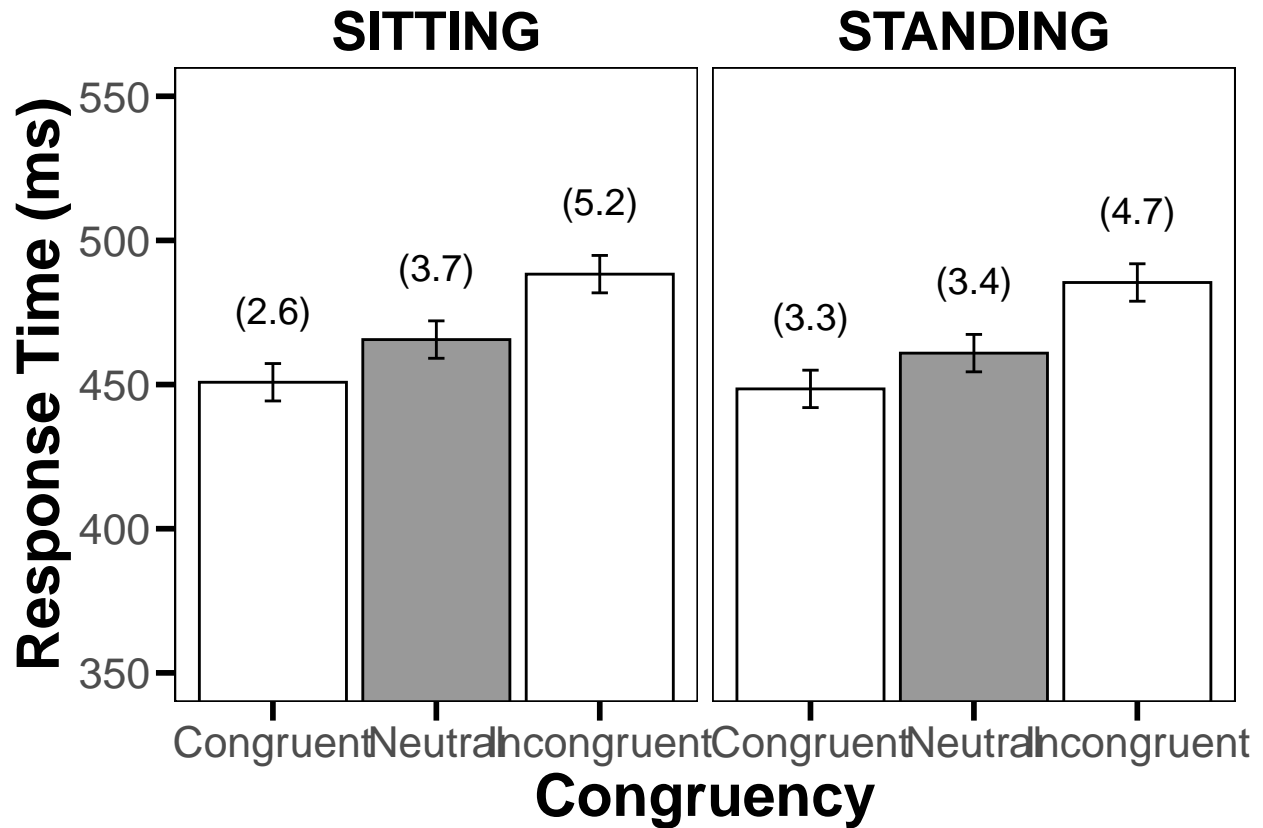
interactionPlot <- ggplot(graphRT, aes(congruency, mean, group=posture)) +
  theme(legend.position = "none")+
  scale_fill_manual(values=c("#FFFFFF","#999999","#FFFFFF","#999999")) +
  coord_cartesian(ylim=c(350,550),expand=TRUE) +
  scale_y_continuous(breaks = round(seq(350, 550, by = 50),0)) +
  geom_text(aes(label=ac),nudge_x=graphRT$hAdj,nudge_y =graphRT$vAdj, size=5) +
  geom_bar(stat="identity", aes(fill=interaction(congruency)),colour="black")+
  geom_errorbar(aes(ymin=min,ymax=max,group=interaction(posture,congruency)), width=.1)+
  labs(x = "Congruency", y = "Response Time (ms)") +
  theme(axis.ticks = element_line(size = 1, colour = "black", linetype = "solid"),
        axis.ticks.length = unit(.25,"cm"),
        #axis.line = element_line(size = 1, colour = "black", linetype = "solid"),
        panel.background = element_rect(fill = "white", colour = "white", size = 1),
        axis.text=element_text(size=16),
        axis.title=element_text(size=22,face="bold"),
        strip.text = element_text(size = 20, face = "bold",colour = "black", angle = 0),
        panel.border = element_rect(colour = "black", fill = NA, size = 0.50),
        strip.background = element_rect(fill=NA,colour="NA",size = 2))+
  facet_grid(~posture)

## Warning: The 'size' argument of 'element_line()' is deprecated as of ggplot2 3.4.0.
## i Please use the 'linewidth' argument instead.

```

```
## Warning: The 'size' argument of 'element_rect()' is deprecated as of ggplot2 3.4.0.
## i Please use the 'linewidth' argument instead.
```

```
interactionPlot
```



```
ggsave(interactionPlot,
        file = "results/plots/fig1_exp1_stroop_interaction_plot.pdf",
        units = "in",
        width = 8.5,
        height = 5,
        dpi = 600)
```

```
exp1.table <-
apa.2way.table(congruency,
               posture,
               meanRT,
               stroopCombined,
               show.conf.interval = TRUE,
               landscape=TRUE,
               filename = "results/plots/exp1_table.doc")
exp1.table
```

```
##
```

```
##
## Means and standard deviations for meanRT as a function of a 3(congruency) X 2(posture) design
##
##           M           M_95%_CI      SD
## posture:SITTING
##   congruency
##     congruent 450.77 [434.78, 466.76] 56.26
##     incongruent 488.31 [462.20, 514.42] 91.87
##     neutral 465.62 [446.78, 484.47] 66.31
##
## posture:STANDING
##   congruency
##     congruent 448.51 [431.30, 465.72] 60.56
##     incongruent 485.40 [456.81, 513.99] 100.61
##     neutral 460.86 [441.64, 480.08] 67.64
##
## Note. M and SD represent mean and standard deviation, respectively.
## LL and UL indicate the lower and upper limits of the
## 95% confidence interval for the mean, respectively.
## The confidence interval is a plausible range of population means
## that could have created a sample mean (Cumming, 2014).
```

## Experiment 2 - Task-switching

### Import and clean data

```
###read in data
ts_path <- "/data/Experiment 2 Data/task-switching-replication-recoded-2.csv"
task_switching_raw <- read.csv(paste0(workingdir, ts_path))
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
## congruentTrialType cueType shapeType shapeColor response correctResponse correct reactionTime
## 1 incongruent solid square blue right left no 0.9088130
## 2 incongruent solid square blue left left yes 0.5947349
## 3 incongruent dashed square blue right right yes 0.7084870
## 4 incongruent dashed square blue right right yes 0.5995200
## 5 congruent dashed square yellow right right yes 0.4399409
## 6 congruent 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
```

### ### check data

```
#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)

#Change RTs from seconds to milliseconds
task_switching_raw2 <- task_switching_raw2 %>%
  mutate(reactionTime = reactionTime * 1000)

#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
#2, 8, 15, 44, 49, 51
#First exclusion criteria
low_acc_subs <- ts_overall_acc %>% filter(Accuracy < 0.80) %>%
  pull(participant)

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

#Summary of time-out trials
timeOutTaskSwitch = task_switching_raw3 %>%
  filter(is.na(reactionTime))

fable(posture~congruentTrialType, timeOutTaskSwitch)

```

```

##                posture sitting standing
## congruentTrialType
## congruent                82          92
## incongruent              98         117

```

```
fable(posture~congruentTrialType~switchTrialType, timeOutTaskSwitch)
```

```

##                congruentTrialType congruent incongruent
## switchTrialType
## noswitch                76          88
## switch                  98         127

```

```
fable(congruentTrialType ~ participant, timeOutTaskSwitch)
```

```

##                congruentTrialType congruent incongruent
## participant
## 1                8          7
## 3                2          3
## 4                3          8
## 7                1          2
## 9                2          0
## 10               1          1
## 11               0          1
## 12               1          2
## 13               4          2
## 14               3          0
## 16               3          4
## 17               1          1
## 18               3          2
## 19              14         21
## 20               1          2
## 21               3         10
## 22               1          1
## 23               6          5
## 24               5          3
## 25               2          3

```

## 26	1	2
## 27	3	5
## 28	4	4
## 29	4	6
## 30	5	6
## 31	11	13
## 32	7	2
## 33	1	4
## 34	3	1
## 35	2	2
## 36	2	2
## 37	1	4
## 38	6	3
## 39	2	5
## 41	3	3
## 42	5	12
## 43	4	5
## 45	0	8
## 46	5	5
## 47	2	0
## 48	10	7
## 50	2	3
## 52	1	0
## 53	1	1
## 54	3	3
## 55	13	27
## 56	6	2
## 57	3	2

```
fable(switchTrialType ~ participant, timeOutTaskSwitch)
```

##	switchTrialType	noswitch	switch
## participant			
## 1	6	9	
## 3	4	1	
## 4	7	4	
## 7	2	1	
## 9	0	2	
## 10	1	1	
## 11	1	0	
## 12	0	3	
## 13	1	5	
## 14	0	3	
## 16	1	6	
## 17	1	1	
## 18	3	2	
## 19	19	16	
## 20	2	1	
## 21	6	7	
## 22	1	1	
## 23	4	7	
## 24	1	7	
## 25	3	2	
## 26	1	2	



```
## 27          5      3
## 28          3      5
## 29          6      4
## 30          5      6
## 31         13     11
## 32          5      4
## 33          1      4
## 34          2      2
## 35          1      3
## 36          2      2
## 37          1      4
## 38          4      5
## 39          3      4
## 41          3      3
## 42          6     11
## 43          5      4
## 45          3      5
## 46          5      5
## 47          0      2
## 48          8      9
## 50          0      5
## 52          1      0
## 53          0      2
## 54          2      4
## 55         11     29
## 56          4      4
## 57          1      4
```

```
#Drop "time-out" trials
task_switching_raw.conventional <- task_switching_raw3 %>%
  filter(!is.na(reactionTime))

write.csv(task_switching_raw.conventional, file = "results/exp2_merged_ts_conv.csv",
  row.names = F)

#Calc mean Acc by participant and conditions (posture, con, switch)
#Narrow format
ts_acc_mean <- task_switching_raw3 %>%
  group_by(participant,
    posture,
    congruentTrialType,
    switchTrialType) %>%
  summarize(Accuracy = mean(correct_bin),
    PE = (1 - Accuracy) * 100)
```

## 'summarise()' has grouped output by 'participant', 'posture', 'congruentTrialType'. You can  
## override using the '.groups' argument.

```
#Convert data to wide format (for statview/SPSS/etc)
ts_acc_mean_wide <- ts_acc_mean %>%
  select(-Accuracy) %>%
  pivot_wider(names_from = c(posture,
    congruentTrialType,
```

```

        switchTrialType),
      values_from = PE)

ts_acc_mean <- ts_acc_mean %>%
  ungroup() %>%
  mutate(across(posture:switchTrialType, as.factor))

str(ts_acc_mean)

## tibble [408 x 6] (S3: tbl_df/tbl/data.frame)
##   $ participant      : int [1:408] 1 1 1 1 1 1 1 1 3 3 ...
##   $ posture          : Factor w/ 2 levels "sitting","standing": 1 1 1 1 2 2 2 2 1 1 ...
##   $ congruentTrialType: Factor w/ 2 levels "congruent","incongruent": 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 [1:408] 0.96 0.978 0.957 0.88 0.981 ...
##   $ PE               : num [1:408] 4 2.17 4.35 12 1.89 ...

```

```

#Total N = 51 (6 dropped for total acc < 80%)
length(unique(ts_acc_mean$participant))

```

```
## [1] 51
```

## Summarize Demographics

```

demo_raw <- read.csv(paste0(workingdir, "/data",
                           "/Experiment 2 Data/Task Switching_February 24, 2022_13.05.csv"),
                    skip = 1) %>%
  slice(-1) %>%
  select(-c(Response.Type, IP.Address, Recipient.Last.Name:Distribution.Channel))
colnames(demo_raw)[10:15] <- c("Gender.Pick", "Gender.Text", "Age", "Race.Pick", "Race.Text", "Eng.First")

dim(demo_raw)

```

```
## [1] 59 15
```

```

#59 records
#first two are test data
# need to match up the 6 dropped participants from behavioral data
demo_df <- demo_raw %>%
  filter(!(X %in% c("test", low_acc_subs)))
dim(demo_df)

```

```
## [1] 51 15
```

```

demo_df <- demo_df %>%
  mutate(Gender.New = ifelse(Gender.Pick %in% c("Man", "Woman"), Gender.Pick, Gender.Text),
         Eng.First = toupper(Eng.First))

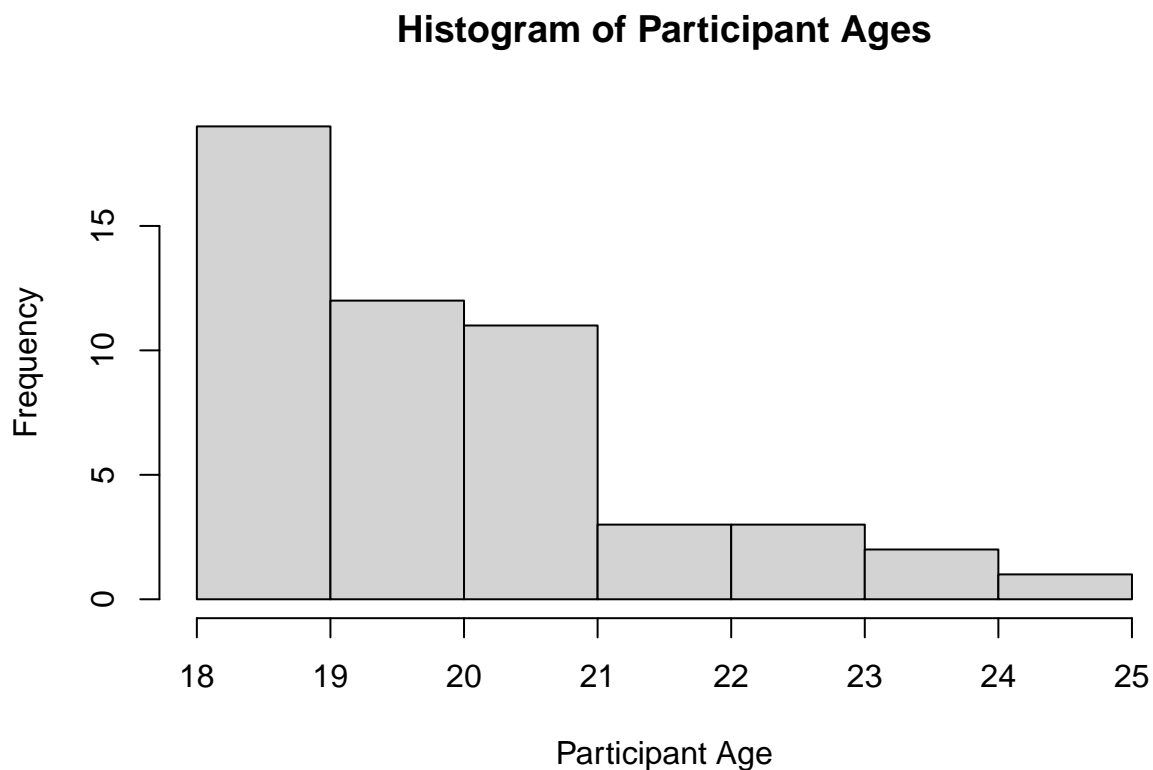
#gender breakdown

```

```
gender_table <- demo_df %>%
  group_by(Gender.New) %>%
  summarize(n = n())
gender_table
```

```
## # A tibble: 3 x 2
##   Gender.New      n
##   <chr>        <int>
## 1 Man          23
## 2 non binaary    1
## 3 Woman        27
```

```
#age breakdown
hist(as.numeric(demo_df$Age),
     main = "Histogram of Participant Ages",
     xlab = "Participant Age")
```



```
age_table <- demo_df %>%
  group_by(Age) %>%
  summarize(n = n())
age_table
```

```
## # A tibble: 8 x 2
##   Age      n
```

```
##   <chr> <int>
## 1 18      9
## 2 19     10
## 3 20     12
## 4 21     11
## 5 22      3
## 6 23      3
## 7 24      2
## 8 25      1
```

```
#age mean and sd
mean_age <- mean(as.numeric(demo_df$Age))
sd_age <- sd(as.numeric(demo_df$Age))

kable(matrix(c(mean_age, sd_age), nrow = 1), col.names = c("Mean of Age", "SD of Age"))
```

Mean of Age	SD of Age
20.21569	1.73567

```
#race breakdown
race_table <- demo_df %>%
  group_by(Race.Pick) %>%
  summarize(n = n()) %>%
  arrange(desc(n))
race_table
```

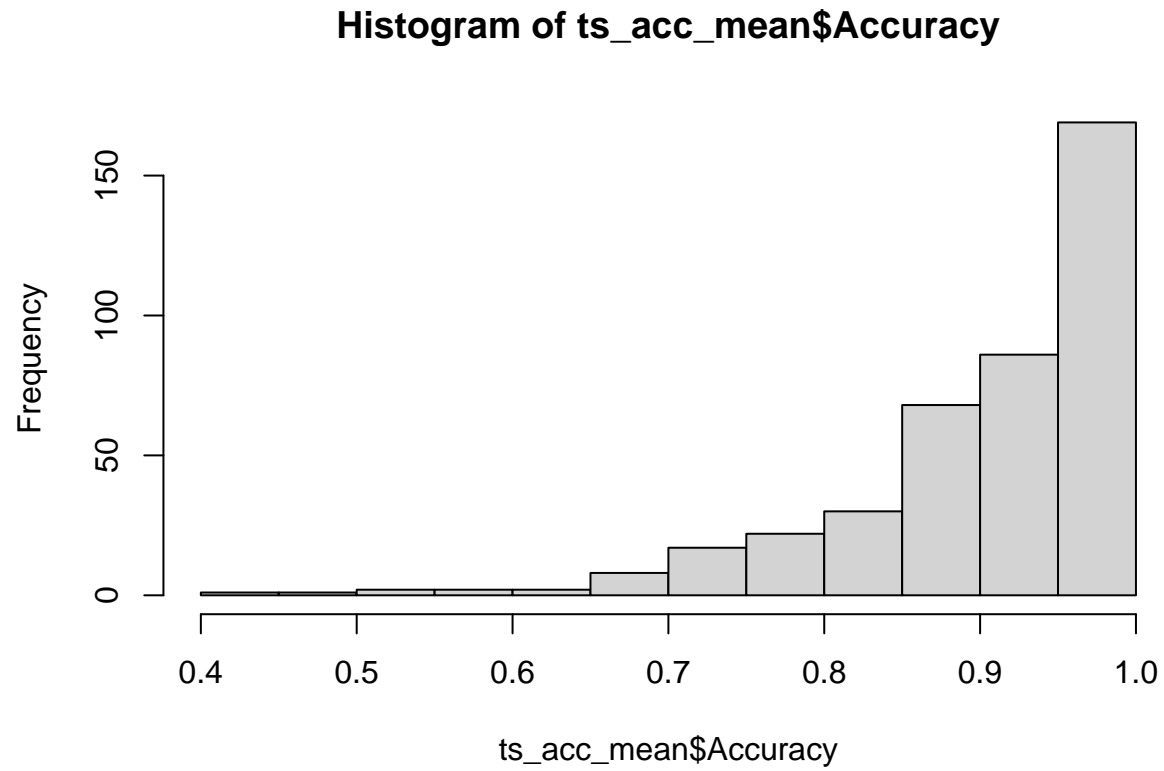
```
## # A tibble: 6 x 2
##   Race.Pick          n
##   <chr>          <int>
## 1 White /European American    22
## 2 Black / African American    11
## 3 Hispanic/Latino/Latina/Latinx 11
## 4 Asian /South Pacific Islander   3
## 5 Central Asian /Indian /Pakistani 3
## 6 Native American / American Indian 1
```

```
#language breakdown
lang_table <- demo_df %>%
  group_by(Eng.First) %>%
  summarize(n=n())
lang_table
```

```
## # A tibble: 2 x 2
##   Eng.First    n
##   <chr>    <int>
## 1 NO        8
## 2 YES       43
```

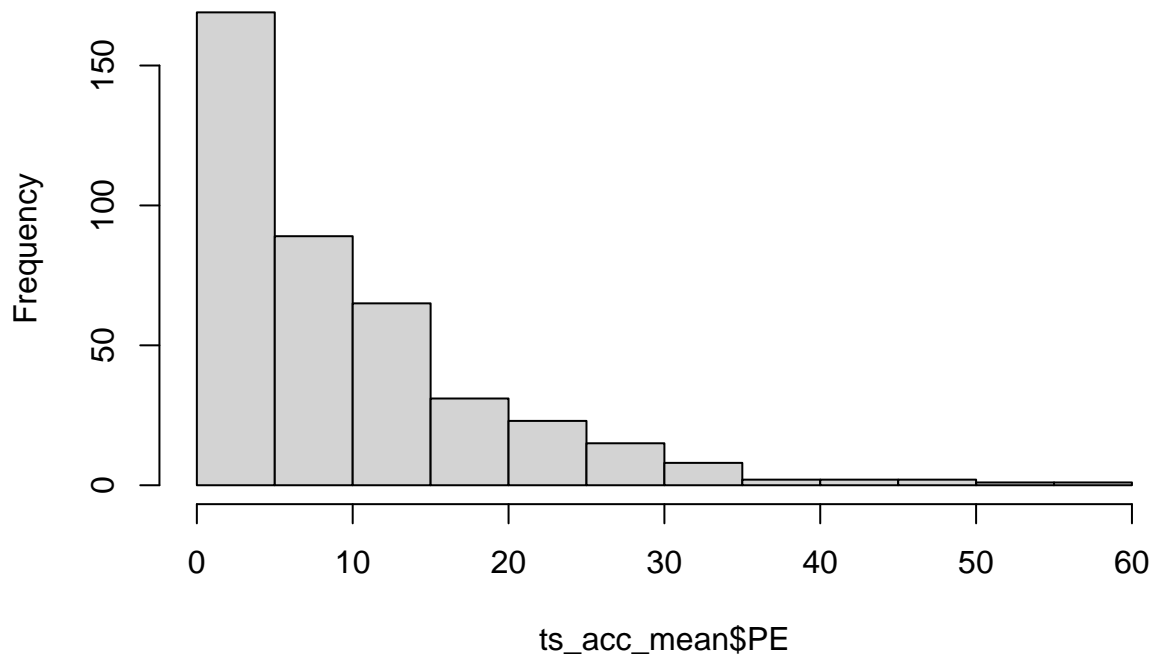
## Accuracy results

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



```
hist(ts_acc_mean$PE)
```

## Histogram of ts\_acc\_mean\$PE



```
accModelTS <- aov_ez(data = ts_acc_mean,
                     dv = "PE",
                     id = "participant",
                     within = c("posture", "congruentTrialType", "switchTrialType"),
                     type = 3,
                     anova_table = list(es = "pes")
)

acc.stats.TS <- ezStats(ts_acc_mean,
                       dv = PE,
                       wid = participant,
                       within = .(posture, congruentTrialType, switchTrialType),
                       type = 3
)
```

## Warning: Converting "participant" to factor for ANOVA.

```
write.csv(acc.stats.TS[, -7], file = "results/exp2_Descriptives_ACC.csv",
          row.names = F)
```

```
write.csv(accModelTS$anova_table, "results/exp2_ANOVA_PE_uncon.csv")
accModelTS
```

## Anova Table (Type 3 tests)

```
##
## Response: PE
##
##          Effect    df    MSE      F    pes p.value
## 1          posture 1, 50 88.41    1.06 .021    .308
## 2      congruentTrialType 1, 50 86.07 99.66 *** .666    <.001
## 3      switchTrialType 1, 50 28.61 92.04 *** .648    <.001
## 4 posture:congruentTrialType 1, 50 41.84    0.02 <.001    .875
## 5 posture:switchTrialType 1, 50 24.00    0.74 .015    .395
## 6 congruentTrialType:switchTrialType 1, 50 15.15 58.43 *** .539    <.001
## 7 posture:congruentTrialType:switchTrialType 1, 50 18.80    1.26 .024    .268
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Calculate confidence interval: PES for posture x switch/condition interaction
#using ANOVA results (partial eta-squared)
interaction_effect_CI <- get.ci.partial.eta.squared(accModelTS$anova_table$F[5],
                                                    accModelTS$anova_table$num Df[5],
                                                    accModelTS$anova_table$den Df[5],
                                                    conf.level = 0.90)
                                                    #90% CI is the convention for PES

interaction_effect_CI
```

```
## $LL
## [1] 0
##
## $UL
## [1] 0.1073579
```

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

#make plot like Smith et al's
acc_plot <-
  superbPlot(ts_acc_mean_wide,
             WSFactors = c("Condition(2)", "Congruent(2)", "Posture(2)"),
             variables = colnames(ts_acc_mean_wide)[2:9],
             errorbar = "SE", #Tempted to change to CI, should stay SE to be consistent with SMith
             plotStyle = "line",
             factorOrder = c("Condition", "Posture", "Congruent"),
             adjustments = list(purpose = "difference")) +
  theme_classic() +
  theme(axis.text=element_text(size=14),
        panel.background = element_rect(fill = "white", colour = "white", size = 1),
        axis.title = element_text(size=14, face="bold"),
        strip.text = element_text(size = 14, face = "bold", colour = "black", angle = 0),
        legend.text = element_text(size = 13),
        legend.title = element_text(size= 14, face = "bold")) +
  ylim(0, 22) + #Trying to make ylim same as the Smith w/o cutting off error bars
  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 = "Percent Error")
```

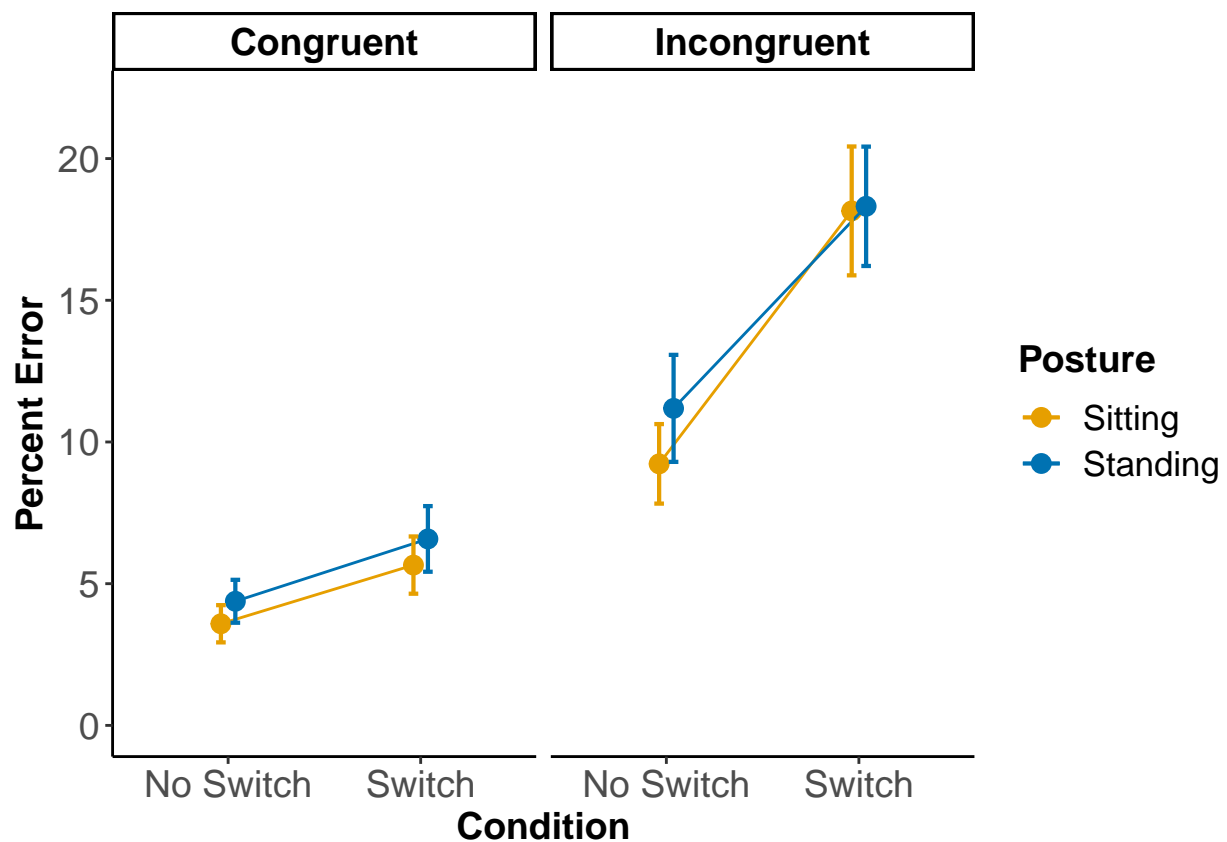
```
## superb::FYI: Here is how the within-subject variables are understood:
```

```
## Condition Congruent Posture variable
##          1          1      1 sitting_congruent_noswitch
##          2          1      1 sitting_congruent_switch
##          1          2      1 sitting_incongruent_noswitch
##          2          2      1 sitting_incongruent_switch
##          1          1      2 standing_congruent_noswitch
##          2          1      2 standing_congruent_switch
##          1          2      2 standing_incongruent_noswitch
##          2          2      2 standing_incongruent_switch
```

```
#Note this is Figure 3, not figure 2 (drawn below)
```

```
ggsave(acc_plot,
  file = "results/plots/fig3_exp2_ts_acc_plot.pdf",
  units = "in",
  width = 6.62,
  height = 5.50,
  dpi = 600)
```

```
acc_plot
```



```
#...CALCULATE THE BAYES FACTORS FOR THE ACC ANALYSIS
```

```
taskswitchBF = data.frame(ts_acc_mean)
```



```

taskswitchBF$participant = factor(ts_acc_mean$participant)
taskswitchBF$posture = factor(ts_acc_mean$posture)
taskswitchBF$congruentTrialType = factor(taskswitchBF$congruentTrialType)
taskswitchBF$switchTrialType = factor(taskswitchBF$switchTrialType)
#str(taskswitchBF)

bfValues2 = anovaBF(Accuracy~ posture*switchTrialType+participant,
                    data = taskswitchBF,
                    whichRandom = "participant",
                    method="laplace")
bfValues2

## Bayes factor analysis
## -----
## [1] posture + participant : 0.1836549 ±NA%
## [2] switchTrialType + participant : 5429475 ±NA%
## [3] posture + switchTrialType + participant : 1064550 ±NA%
## [4] posture + switchTrialType + posture:switchTrialType + participant : 168897.1 ±NA%
##
## Against denominator:
## Accuracy ~ participant
## ---
## Bayes factor type: BFlinearModel, JZS

#...get the Bayes factor for the Null Interaction (vs. model w/no interaction)
bfValues2[3]/bfValues2[4]

## Bayes factor analysis
## -----
## [1] posture + switchTrialType + participant : 6.302952 ±NA%
##
## Against denominator:
## Accuracy ~ posture + switchTrialType + posture:switchTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS

1/(bfValues2[3]/bfValues2[4])

## Bayes factor analysis
## -----
## [1] posture + switchTrialType + posture:switchTrialType + participant : 0.1586558 ±NA%
##
## Against denominator:
## Accuracy ~ posture + switchTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS

```

## Reaction time results

```

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

#Second exclusion criteria
#How many trials faster than 100 ms? Only a single one
sum(ts_correct_only$reactionTime < 100)

## [1] 1

dim(ts_correct_only)

## [1] 17699    20

ts_correct_only2 <- ts_correct_only %>% filter(reactionTime >= 100)
#Sanity check, one trial is dropped. Now have 17,698 trials
dim(ts_correct_only2)

## [1] 17698    20

trimOutputTS = pjRecursiveTrim2(dataSet = ts_correct_only2,
                                dv = "reactionTime",
                                splitvars = c("participant",
                                                "posture",
                                                "switchTrialType",
                                                "congruentTrialType"))

trimmedTSData=trimOutputTS[[1]]
totalN.TS = trimOutputTS[[2]]
rejectedTS = trimOutputTS[[3]]
percentTrimmedTS = trimOutputTS[[4]] #this is very close to the percentage trimmed for stroop
#2.14% of trials
percentTrimmedTS

## [1] 2.141485

NcellsTS = trimOutputTS[[5]] # 51 participants * 8 conditions

trimmed_rt_mean_TS <- trimmedTSData %>%
  group_by(participant,
            posture,
            congruentTrialType,
            switchTrialType) %>%
  summarize(mean_rt = mean(reactionTime))

## 'summarise()' has grouped output by 'participant', 'posture', 'congruentTrialType'. You can
## override using the '.groups' argument.

```

```

#Convert data to wide format
trimmed_rt_mean_TS_wide <- trimmed_rt_mean_TS %>%
  pivot_wider(names_from = c(posture,
                             congruentTrialType,
                             switchTrialType),
              values_from = mean_rt)

trimmed_RT_plot <-
  superbPlot(trimmed_rt_mean_TS_wide,
             WSFactors = c("Condition(2)", "Congruent(2)", "Posture(2)"),
             variables = colnames(trimmed_rt_mean_TS_wide)[2:9],
             errorbar = "SE",
             plotStyle = "line",
             factorOrder = c("Condition", "Posture", "Congruent"),
             adjustments = list(purpose = "difference"))+
  theme_classic() +
  theme(axis.text=element_text(size=14),
        panel.background = element_rect(fill = "white", colour = "white", size = 1),
        axis.title = element_text(size=14, face="bold"),
        strip.text = element_text(size = 14, face = "bold", colour = "black", angle = 0),
        legend.text = element_text(size = 13),
        legend.title = element_text(size= 14, face = "bold")) +
  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")) +
  ylim(500, 700) +
  labs(y = "Response Time (ms)")

```

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

```
## Condition Congruent Posture variable
##      1      1      1 sitting_congruent_noswitch
##      2      1      1 sitting_congruent_switch
##      1      2      1 sitting_incongruent_noswitch
##      2      2      1 sitting_incongruent_switch
##      1      1      2 standing_congruent_noswitch
##      2      1      2 standing_congruent_switch
##      1      2      2 standing_incongruent_noswitch
##      2      2      2 standing_incongruent_switch

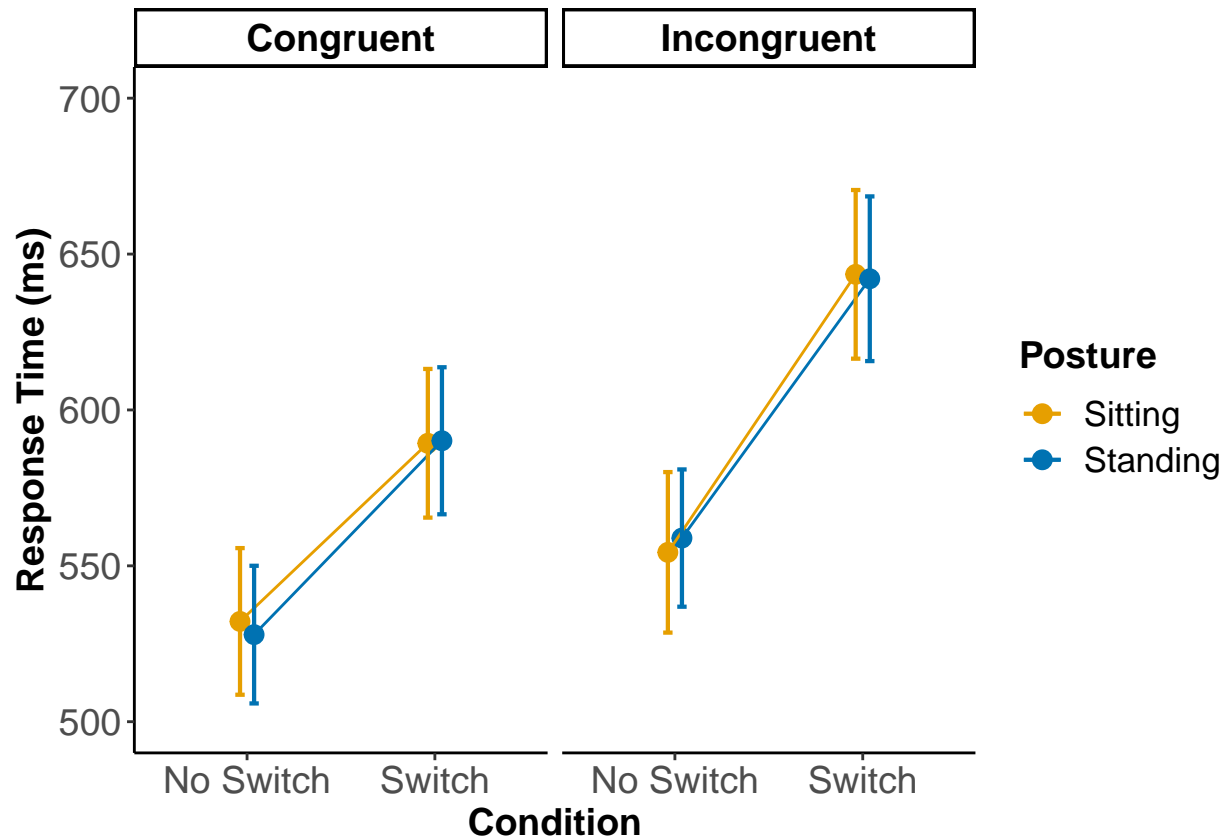
```

```

ggsave(trimmed_RT_plot,
       file = "results/plots/fig2_exp2_ts_trimmed_RT_plot.pdf",
       units = "in",
       width = 6.62,
       height = 5.50,
       dpi = 600)

trimmed_RT_plot

```



```
rtModelTS <- aov_ez(data = trimmed_rt_mean_TS,
  dv = 'mean_rt',
  id = 'participant',
  within = c('posture',
    'congruentTrialType',
    'switchTrialType'),
  type = 3,
  anova_table = list(es = "pes"))

rt.stats.TS <- ezStats(trimmed_rt_mean_TS,
  dv = mean_rt,
  wid = participant,
  within = .(posture, congruentTrialType, switchTrialType),
  type = 3
)
```

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

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

```
## Warning: Converting "congruentTrialType" to factor for ANOVA.
```

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

```

write.csv(rt.stats.TS[ , -7],
          file = "results/exp2_Descriptives_trimmed_RT.csv",
          row.names = F)

write.csv(rtModelTS$anova_table, file = "results/exp2_ANOVA_trimmed_RT.csv")
rtModelTS

## Anova Table (Type 3 tests)
##
## Response: mean_rt
##
##          Effect    df    MSE          F    pes p.value
## 1          posture 1, 50 9922.05      0.00 <.001    .995
## 2 congruentTrialType 1, 50 3302.21  48.98 ***    .495    <.001
## 3 switchTrialType 1, 50 4166.22 130.17 ***    .722    <.001
## 4 posture:congruentTrialType 1, 50 1592.27      0.17    .003    .679
## 5 posture:switchTrialType 1, 50 1519.08      0.00 <.001    .951
## 6 congruentTrialType:switchTrialType 1, 50 1252.33  14.32 ***    .223    <.001
## 7 posture:congruentTrialType:switchTrialType 1, 50 1552.36      0.50    .010    .483
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

trimmed_rt_mean_TS2 <- data.frame(trimmed_rt_mean_TS)
trimmed_rt_mean_TS2$participant = factor(trimmed_rt_mean_TS2$participant)
trimmed_rt_mean_TS2$posture = factor(trimmed_rt_mean_TS2$posture)
trimmed_rt_mean_TS2$congruentTrialType = factor(trimmed_rt_mean_TS2$congruentTrialType)
trimmed_rt_mean_TS2$switchTrialType = factor(trimmed_rt_mean_TS2$switchTrialType)

bfValues2.RT = anovaBF(mean_rt ~ posture*switchTrialType+participant,
                      data = trimmed_rt_mean_TS2,
                      whichRandom = "participant",
                      method="laplace")
bfValues2.RT

## Bayes factor analysis
## -----
## [1] posture + participant : 0.1007388 ±NA%
## [2] switchTrialType + participant : 4.227658e+24 ±NA%
## [3] posture + switchTrialType + participant : 4.265145e+23 ±NA%
## [4] posture + switchTrialType + posture:switchTrialType + participant : 6.012989e+22 ±NA%
##
## Against denominator:
## mean_rt ~ participant
## ---
## Bayes factor type: BFlinearModel, JZS

#...get the Bayes factor for the Null Interaction (vs. model w/no interaction)
bfValues2.RT[3]/bfValues2.RT[4]

## Bayes factor analysis
## -----
## [1] posture + switchTrialType + participant : 7.093219 ±NA%
##

```

```
## Against denominator:
##   mean_rt ~ posture + switchTrialType + posture:switchTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS

1/(bfValues2.RT[3]/bfValues2.RT[4])

## Bayes factor analysis
## -----
## [1] posture + switchTrialType + posture:switchTrialType + participant : 0.1409797 ±NA%
##
## Against denominator:
##   mean_rt ~ posture + switchTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS
```

## Experiment 3 - Visual search

### Import and clean data

```
vs_files = list.files(path = "data/Experiment 3 Data/", full.names = T)
vs_files = vs_files[str_detect(vs_files, pattern="(?.*SJ)(?.*.txt)")]

merged.VS.data <- ldply(vs_files,
                        read.delim,
                        header=FALSE,
                        stringsAsFactors = FALSE,
                        sep = "") #for each item in the list apply the function read.delim

##..ADD HEADERS
names(merged.VS.data) = c("sj",
                          "cb",
                          "blockNumber",
                          "blockType",
                          "trialNum",
                          "target",
                          "targetImage",
                          "distractor",
                          "distractorImage",
                          "posture",
                          "setSize",
                          "rt",
                          "resp",
                          "cresp",
                          "ac")

##...look at unique values from both columns
unique(merged.VS.data[c('sj')])
```

```
##      sj
```

```
## 1      1
## 265    10
## 529    11
## 793    12
## 1057   13
## 1321   14
## 1585   15
## 1849   16
## 2113   17
## 2377   18
## 2641   19
## 2905    2
## 3169   20
## 3433   21
## 3697   22
## 3961   23
## 4225   24
## 4489   25
## 4753   26
## 5017   27
## 5281   28
## 5545   29
## 5809    3
## 6073   30
## 6337   31
## 6601   32
## 6865   33
## 7129   34
## 7393   35
## 7657   36
## 7921   37
## 8185   38
## 8449   39
## 8713    4
## 8977   40
## 9241   41
## 9505   42
## 9769   43
## 10033  44
## 10297  45
## 10561  46
## 10825  47
## 11089  48
## 11353  49
## 11617   5
## 11881  50
## 12145   6
## 12409   7
## 12673   8
## 12937   9
```

```
unique(merged.VS.data[c('blockType')])
```

```
##      blockType
```

```
## 1      practice
## 9 experimental
```

```
## .DOES EACH SUBJECT HAVE THE SAME NUMBER OF TRIALS
ftable(blockType~sj, merged.VS.data)
```

```
##      blockType experimental practice
## sj
## 1              256           8
## 2              256           8
## 3              256           8
## 4              256           8
## 5              256           8
## 6              256           8
## 7              256           8
## 8              256           8
## 9              256           8
## 10             256           8
## 11             256           8
## 12             256           8
## 13             256           8
## 14             256           8
## 15             256           8
## 16             256           8
## 17             256           8
## 18             256           8
## 19             256           8
## 20             256           8
## 21             256           8
## 22             256           8
## 23             256           8
## 24             256           8
## 25             256           8
## 26             256           8
## 27             256           8
## 28             256           8
## 29             256           8
## 30             256           8
## 31             256           8
## 32             256           8
## 33             256           8
## 34             256           8
## 35             256           8
## 36             256           8
## 37             256           8
## 38             256           8
## 39             256           8
## 40             256           8
## 41             256           8
## 42             256           8
## 43             256           8
## 44             256           8
## 45             256           8
## 46             256           8
```



```
## 47          256      8
## 48          256      8
## 49          256      8
## 50          256      8
```

```
###DO WE HAVE EQUAL OBSERVATIONS FOR EACH COUNTERBALANCE
```

```
fable(blockType~cb, merged.VS.data)
```

```
##      blockType experimental practice
## cb
## 1          6400      200
## 2          6400      200
```

```
###LOOK FOR MISSING DATA
```

```
merged.VS.data[!complete.cases(merged.VS.data),]
```

```
## [1] sj          cb          blockNumber    blockType      trialNum       target
## [7] targetImage    distractor     distractorImage posture        setSize        rt
## [13] resp          cresp         ac
## <0 rows> (or 0-length row.names)
```

```
### GET RID OF PRACTICE TRIALS
```

```
merged.VS.data <- merged.VS.data[!merged.VS.data$blockType=="practice",]
```

```
### CHECK TRIALS PER CONDITION
```

```
fable(posture+target+distractor+setSize~sj, merged.VS.data)
```

```
##      posture      SITTING                                STANDING
##      target      h          s          h          s
##      distractor    e      u      e      u      e      u      e      u
##      setSize      4  8  4  8  4  8  4  8      4  8  4  8  4  8  4  8
## sj
## 1          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 2          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 3          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 4          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 5          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 6          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 7          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 8          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 9          16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 10         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 11         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 12         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 13         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 14         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 15         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 16         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 17         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 18         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 19         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 20         16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
```

```
## 21      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 22      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 23      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 24      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 25      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 26      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 27      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 28      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 29      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 30      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 31      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 32      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 33      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 34      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 35      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 36      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 37      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 38      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 39      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 40      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 41      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 42      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 43      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 44      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 45      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 46      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 47      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 48      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 49      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
## 50      16 16 16 16 16 16 16 16      16 16 16 16 16 16 16 16
```

```
##... UNLIKE THE STROOP, PARTICIPANTS WERE ALLOWED TO TAKE LONGER THAN 1500MS BUT WERE GIVEN A WARNING
##... TRIALS LONGER THAN 1500 MS will be considered errors (i.e., they will be dropped in RT but kept in
##... Set values in the ac column to 0 on trials where a response is > = 1500
```

```
##...check that only experimental trials are left
unique(merged.VS.data$blockType)
```

```
## [1] "experimental"
```

```
write.table(merged.VS.data, file = "results/exp3_merged_vs_data.txt", row.names = F)
```

```
##...count trials
```

```
totalTrialsVS = dim(merged.VS.data)[1]
```

```
observationDataVS = data.frame(ftable(blockType~sj, merged.VS.data))[,c(1,3)]
```

```
##...get the number of extreme trials <100 - anticipatory or fast responses
```

```
merged.VS.data= merged.VS.data[!merged.VS.data$rt<=100,]
```

```
validRTTrialsVS = dim(merged.VS.data)[1]
```

```
observationDataVS$validTrials = data.frame(ftable(blockType~sj, merged.VS.data))[,c(3)]
```

```
print(paste("percent invalid trials = ", ((totalTrialsVS-validRTTrialsVS)/totalTrialsVS)*100))
```

```
## [1] "percent invalid trials = 0"
```

```
#...get the number of time outs
timeOutVS = merged.VS.data %>% filter(rt >= 1500)
dim(timeOutVS)[1]
```

```
## [1] 168
```

```
ftable(posture~setSize, timeOutVS)
```

```
##      posture SITTING STANDING
## setSize
## 4          32          13
## 8          56          67
```

```
ftable(posture~sj, timeOutVS)
```

```
##      posture SITTING STANDING
## sj
## 2          9          12
## 3          2          0
## 4          6          2
## 5          3          1
## 6          0          1
## 7          1          2
## 9          4          6
## 10         7          3
## 11         3          2
## 12         2          7
## 13         4          1
## 15         1          2
## 17         2          0
## 18         0          1
## 19         3          2
## 20         1          2
## 22         2          2
## 24         0          2
## 25         2          4
## 27         0          2
## 28         0          1
## 29         5          6
## 30         2          0
## 31         3          0
## 33         0          1
## 35         3          2
## 36         0          1
## 37         0          1
## 39         2          3
## 40         3          3
## 41         0          2
## 42         2          1
## 44         4          2
## 45         3          0
## 46         3          0
```

```
## 47          0          1
## 48          1          0
## 49          3          2
## 50          2          0
```

```
fable(blockType~setSize, timeOutVS)
```

```
##          blockType experimental
## setSize
## 4          45
## 8          123
```

```
merged.VS.data.conventional <- merged.VS.data %>% filter(rt < 1500)
write.csv(merged.VS.data.conventional,
          file = "results/exp3_merged_vs_data_conv.csv",
          row.names = F)
```

```
#...this code changes the 1550ms+ trials into errors
```

```
merged.VS.data.orig <- merged.VS.data
merged.VS.data$ac[merged.VS.data$rt>=1500] = 0
```

```
vsCorrect = merged.VS.data[merged.VS.data$ac ==1,]
```

```
errorsRemovedVS = dim(vsCorrect)[1]
observationDataVS$correctTrials = data.frame(fable(blockType~sj, vsCorrect))[c(3)]
```

```
trimInfo = data.frame(totalTrialsVS, validRTTrialsVS, errorsRemovedVS)
head(trimInfo)
```

```
## totalTrialsVS validRTTrialsVS errorsRemovedVS
## 1          12800          12800          12397
```

```
#####
```

```
#...CHECK 20% CRITERION
```

```
#####
```

```
observationDataVS$percentLoss = ((observationDataVS$Freq-observationDataVS$correctTrials)/observationDataVS$Freq)*100
sum(observationDataVS$percentLoss>20)
```

```
## [1] 0
```

```
#...None!
```

```
#...RUN TRIMMING PROCEDURE
```

```
tempList = pjRecursiveTrim2(vsCorrect, #...dataset
                             "rt", #...dependent variables
                             c("sj",
                               "cb",
                               "setSize",
                               "posture")) #.independent variables
```

```
trimmedData=tempList[[1]]
```

```
totalN = tempList[[2]]
rejected = tempList[[3]]
percentTrimmed = tempList[[4]]
Ncells = tempList[[5]]

print(paste("Percent of outliers removed: ",round(percentTrimmed,3)))
```

```
## [1] "Percent of outliers removed: 1.339"
```

```
###...get the trimming info
output.out= data.frame(totalN, rejected,percentTrimmed,Ncells)
head(output.out)
```

```
## totalN rejected percentTrimmed Ncells
## 1 12397 166 1.339034 200
```

```
###...get mean error data
vsPE = plyr::ddply(merged.VS.data,
                  .(sj,cb,setSize, posture),
                  summarise,
                  meanPE = 100 - (mean(ac)*100))
head(vsPE)
```

```
## sj cb setSize posture meanPE
## 1 1 1 4 SITTING 0.0000
## 2 1 1 4 STANDING 0.0000
## 3 1 1 8 SITTING 0.0000
## 4 1 1 8 STANDING 0.0000
## 5 2 1 4 SITTING 4.6875
## 6 2 1 4 STANDING 4.6875
```

```
vsRT = plyr::ddply(trimmedData,
                  .(sj, cb, setSize,posture),
                  summarise,
                  meanRT = mean(rt))
```

```
###...combine the RT and error data
vsCombined = cbind(vsRT,meanPE =vsPE$meanPE)
str(vsCombined)
```

```
## 'data.frame': 200 obs. of 6 variables:
## $ sj : int 1 1 1 1 2 2 2 2 3 3 ...
## $ cb : int 1 1 1 1 1 1 1 1 1 1 ...
## $ setSize: int 4 4 8 8 4 4 8 8 4 4 ...
## $ posture: chr "SITTING" "STANDING" "SITTING" "STANDING" ...
## $ meanRT : num 677 593 736 620 792 ...
## $ meanPE : num 0 0 0 0 4.69 ...
```

```
###...set as factors
vsCombined$sj = factor(vsCombined$sj)
vsCombined$cb = factor(vsCombined$cb)
```

```
vsCombined$setSize = factor(vsCombined$setSize)
vsCombined$postureFactor = factor(vsCombined$posture)
summary(vsCombined$cb)
```

```
## 1 2
## 100 100
```

## Reaction time results

```
rtModelVS <- ezANOVA(vsCombined,
  dv = .(meanRT),
  wid = .(sj),
  within = .(postureFactor, setSize),
  detailed = TRUE,
  type = 3,
  return_aov = TRUE)
```

```
rtModelVS$ANOVA
```

##	Effect	DFn	DFd	SSn	SSd	F	p	p<.05
## 1	(Intercept)	1	49	1.084958e+08	1189588.17	4.469020e+03	8.326740e-50	*
## 2	postureFactor	1	49	2.052064e+04	153738.29	6.540411e+00	1.369090e-02	*
## 3	setSize	1	49	3.574624e+05	46863.03	3.737628e+02	1.414816e-24	*
## 4	postureFactor:setSize	1	49	2.246613e+01	35654.35	3.087534e-02	8.612429e-01	
##	ges							
## 1							9.870285e-01	
## 2							1.418774e-02	
## 3							2.004492e-01	
## 4							1.575613e-05	

```
rt.VS.MSE <- rtModelVS$ANOVA$SSd/rtModelVS$ANOVA$DFd
```

```
##...print ANOVA in nice format
paste(rtModelVS$ANOVA$Effect,": F(",
  rtModelVS$ANOVA$DFn,
  ", ",
  rtModelVS$ANOVA$DFd,
  ") = ",
  round(rtModelVS$ANOVA$F,3),
  ", MSE = ",
  round(rt.VS.MSE,3),
  ", p = ",
  round(rtModelVS$ANOVA$p,3),
  ", partialEtaSq = ",
  round(rtModelVS$ANOVA$SSn/(rtModelVS$ANOVA$SSn+rtModelVS$ANOVA$SSd),4),sep="")
```

```
## [1] "(Intercept): F(1, 49) = 4469.02, MSE = 24277.31, p = 0, partialEtaSq = 0.9892"
## [2] "postureFactor: F(1, 49) = 6.54, MSE = 3137.516, p = 0.014, partialEtaSq = 0.1178"
## [3] "setSize: F(1, 49) = 373.763, MSE = 956.388, p = 0, partialEtaSq = 0.8841"
## [4] "postureFactor:setSize: F(1, 49) = 0.031, MSE = 727.64, p = 0.861, partialEtaSq = 6e-04"
```

```
#...CALCULATE THE BAYES FACTORS FOR THE RT ANALYSIS
bfValues3 = anovaBF(meanRT~setSize*postureFactor+sj,
                    data = vsCombined,
                    whichRandom = "sj",
                    method="laplace")
bfValues3
```

```
## Bayes factor analysis
## -----
## [1] setSize + sj : 2.916459e+26 ±NA%
## [2] postureFactor + sj : 1.51507 ±NA%
## [3] setSize + postureFactor + sj : 1.321058e+28 ±NA%
## [4] setSize + postureFactor + setSize:postureFactor + sj : 2.585184e+27 ±NA%
##
## Against denominator:
## meanRT ~ sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
#...get the Bayes factor for the Null Interaction
bfValues3[3]/bfValues3[4]
```

```
## Bayes factor analysis
## -----
## [1] setSize + postureFactor + sj : 5.110113 ±NA%
##
## Against denominator:
## meanRT ~ setSize + postureFactor + setSize:postureFactor + sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
1/(bfValues3[3]/bfValues3[4])
```

```
## Bayes factor analysis
## -----
## [1] setSize + postureFactor + setSize:postureFactor + sj : 0.1956904 ±NA%
##
## Against denominator:
## meanRT ~ setSize + postureFactor + sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
#####
# GET DIFFERENCE SCORES - SEARCH RATE
#####

wideData = dcast(vsCombined, #the name of the dataframe you want to reshape
                 sj+cb #row variables
                 ~posture+setSize, #row variables ~ column variables
                 value.var = "meanRT")
head(wideData)
```

```
##   sj cb SITTING_4 SITTING_8 STANDING_4 STANDING_8
## 1  1  1  676.5238  735.5397   593.1129   619.6406
## 2  2  1  792.4590  931.9474   815.7213   993.5000
## 3  3  1  721.2787  827.2222   654.1639   774.5238
## 4  4  1  695.7119  741.9298   660.2632   653.0172
## 5  5  1  693.6034  839.2903   705.1967   759.7419
## 6  6  1  625.3750  694.0484   592.4531   687.5645
```

```
wideData$sittingEffect = (wideData$SITTING_8-wideData$SITTING_4)/4
wideData$standingEffect = (wideData$STANDING_8-wideData$STANDING_4)/4
wideData$interaction = wideData$sittingEffect - wideData$standingEffect
```

```
searchratestand = mean(wideData$standingEffect) #...search rate in standing condition
searchratesit = mean(wideData$sittingEffect) #...search rate in the sitting condition

searchratestand
```

```
## [1] 21.30589
```

```
searchratesit
```

```
## [1] 20.97073
```

```
#One-sample t-tests
t.test(wideData$standingEffect)
```

```
##
## One Sample t-test
##
## data: wideData$standingEffect
## t = 16.69, df = 49, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 18.74050 23.87127
## sample estimates:
## mean of x
## 21.30589
```

```
t.test(wideData$sittingEffect)
```

```
##
## One Sample t-test
##
## data: wideData$sittingEffect
## t = 13.055, df = 49, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 17.74261 24.19884
## sample estimates:
## mean of x
## 20.97073
```



```

##...Exact Binomial SIGN TEST
binom.test(length(wideData$interaction[wideData$interaction>=0]),
            length(unique(vsCombined$sj)))

##
## Exact binomial test
##
## data: length(wideData$interaction[wideData$interaction >= 0]) and length(unique(vsCombined$sj))
## number of successes = 25, number of trials = 50, p-value = 1
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.355273 0.644727
## sample estimates:
## probability of success
##                                0.5

```

## Percent error results

```

errModelVS <- ezANOVA(vsCombined,
                      dv = .(meanPE),
                      wid=.(sj),
                      within=.(postureFactor,setSize),
                      detailed=TRUE,
                      type=3,
                      return_aov = TRUE)

```

```
errModelVS
```

```

## $ANOVA
##           Effect DFn DFD          SSn          SSd          F          p p<.05          ges
## 1      (Intercept)    1   49 1982.531738 1158.9478 83.8209098 3.463466e-12      * 0.504342884
## 2      postureFactor    1   49   3.527832  227.7954  0.7588554 3.879351e-01      0.001807368
## 3           setSize    1   49 129.504395  343.5181 18.4727266 8.162026e-05      * 0.062324860
## 4 postureFactor:setSize    1   49  20.520020  218.1274  4.6096032 3.676850e-02      * 0.010422027
##
## $aov
##
## Call:
## aov(formula = formula(aov_formula), data = data)
##
## Grand Mean: 3.148438
##
## Stratum 1: sj
##
## Terms:
##           Residuals
## Sum of Squares 1158.948
## Deg. of Freedom    49
##
## Residual standard error: 4.863332
##

```

```
## Stratum 2: sj:postureFactor
##
## Terms:
##           postureFactor Residuals
## Sum of Squares      3.52783 227.79541
## Deg. of Freedom      1      49
##
## Residual standard error: 2.156128
## 1 out of 2 effects not estimable
## Estimated effects are balanced
##
## Stratum 3: sj:setSize
##
## Terms:
##           setSize Residuals
## Sum of Squares 129.5044 343.5181
## Deg. of Freedom      1      49
##
## Residual standard error: 2.647749
## 1 out of 2 effects not estimable
## Estimated effects are balanced
##
## Stratum 4: sj:postureFactor:setSize
##
## Terms:
##           postureFactor:setSize Residuals
## Sum of Squares      20.52002 218.12744
## Deg. of Freedom      1      49
##
## Residual standard error: 2.109877
## Estimated effects are balanced
```

```
err.VS.MSE <- errModelVS$ANOVA$SSd/errModelVS$ANOVA$DFd

exp3_ANOVA_PE_uncon <-
paste(errModelVS$ANOVA$Effect,": F(",
      errModelVS$ANOVA$DFn,
      ", ",
      errModelVS$ANOVA$DFd,
      ") = ",
      round(errModelVS$ANOVA$F,3),
      ", MSE = ",
      round(err.VS.MSE,3),
      ", p = ",
      round(errModelVS$ANOVA$p,3),
      ", partialEtaSq = ",
      round(errModelVS$ANOVA$SSn/(errModelVS$ANOVA$SSn+errModelVS$ANOVA$SSd),4),sep="")

write.csv(exp3_ANOVA_PE_uncon, "results/exp3_ANOVA_PE_uncon.csv")

wideData = dcast(vsCombined, #the name of the dataframe you want to reshape
  sj+cb #row variables
  ~posture+setSize, #row variables ~ column variables
  value.var = "meanPE")
```

```
head(wideData)
```

```
##   sj cb SITTING_4 SITTING_8 STANDING_4 STANDING_8
## 1  1  1   0.0000   0.0000   0.0000   0.0000
## 2  2  1   4.6875  10.9375   4.6875  15.6250
## 3  3  1   1.5625   1.5625   0.0000   0.0000
## 4  4  1   6.2500  10.9375   4.6875   6.2500
## 5  5  1   3.1250   3.1250   1.5625   1.5625
## 6  6  1   0.0000   1.5625   0.0000   1.5625
```

```
wideData$SittingEffect = (wideData$SITTING_8-wideData$SITTING_4)/4
wideData$standingEffect = (wideData$STANDING_8-wideData$STANDING_4)/4
wideData$interaction = wideData$SittingEffect - wideData$standingEffect
```

```
searchratestand = mean(wideData$standingEffect) #...search rate in standing condition
searchratesit = mean(wideData$SittingEffect) #...search rate in the sitting condition
```

```
searchratestand
```

```
## [1] 0.5625
```

```
searchratesit
```

```
## [1] 0.2421875
```

```
#One-sample t-tests
```

```
t.test(wideData$standingEffect)
```

```
##
## One Sample t-test
##
## data: wideData$standingEffect
## t = 4.0858, df = 49, p-value = 0.0001623
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 0.2858399 0.8391601
## sample estimates:
## mean of x
## 0.5625
```

```
t.test(wideData$SittingEffect)
```

```
##
## One Sample t-test
##
## data: wideData$SittingEffect
## t = 2.4588, df = 49, p-value = 0.01752
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 0.04424588 0.44012912
## sample estimates:
## mean of x
## 0.2421875
```

```

#...CALCULATE THE BAYES FACTORS FOR THE RT ANALYSIS
bfValues3.error = anovaBF(meanPE~setSize*postureFactor+sj,
                          data = vsCombined,
                          whichRandom = "sj",
                          method="laplace")
bfValues3.error

## Bayes factor analysis
## -----
## [1] setSize + sj : 4888.005 ±NA%
## [2] postureFactor + sj : 0.1823325 ±NA%
## [3] setSize + postureFactor + sj : 937.7872 ±NA%
## [4] setSize + postureFactor + setSize:postureFactor + sj : 997.3734 ±NA%
##
## Against denominator:
## meanPE ~ sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

```

#...get the Bayes factor for the Null Interaction
bfValues3.error[3]/bfValues3.error[4]

```

```

## Bayes factor analysis
## -----
## [1] setSize + postureFactor + sj : 0.9402568 ±NA%
##
## Against denominator:
## meanPE ~ setSize + postureFactor + setSize:postureFactor + sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

```

1/(bfValues3.error[3]/bfValues3.error[4])

```

```

## Bayes factor analysis
## -----
## [1] setSize + postureFactor + setSize:postureFactor + sj : 1.063539 ±NA%
##
## Against denominator:
## meanPE ~ setSize + postureFactor + sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

## Make plots for visual search

```

graphRT3 = describeBy(vsCombined$meanRT,
                      list(vsCombined$posture,vsCombined$setSize),
                      mat=TRUE,
                      digits = 1)
graphPE3 = describeBy(vsCombined$meanPE,
                      list(vsCombined$posture,vsCombined$setSize),

```

```

mat=TRUE,
digits = 1)

graphRT3 = graphRT3[,c("group1","group2","mean","se")]
graphPE3 = graphPE3[,c("group1","group2","mean","se")]

names(graphRT3) = c("posture","setSize","mean","se")
names(graphPE3) = c("posture","setSize","mean","se")

graphRT3$posture = str_to_upper(graphRT3$posture)

#####
#..calculate the within subjects confidence intervals based on loftus and masson
#..the confidence intervals are based on the interaction term.
#####

graphRT3$se = sqrt((rt.VS.MSE[4])/length(unique(vsCombined$sj)))
graphPE3$se= sqrt((err.VS.MSE[4])/length(unique(vsCombined$sj)))

#####
#..calculate the within subjects confidence intervals based on loftus and masson
#..the confidence intervals are based on the interaction term.
#####

critT3 = qt(p=.025,df=length(unique(vsCombined$sj))-2,lower.tail =FALSE)

#---add the min and max for the confidence intervals
graphRT3$min = graphRT3$mean - (graphRT3$se*critT3)
graphRT3$max = graphRT3$mean + (graphRT3$se*critT3)

####GET AC DATA FROM twoAnimalWordsPRPac.R
graphRT3$ac = paste("(",format(round(graphPE3$mean,digits=1),nsmall = 1),"),",sep="")
head(graphRT3)

```

```

##      posture setSize  mean      se      min      max      ac
## X11  SITTING      4 704.7 3.814813 697.0298 712.3702 (2.5)
## X12 STANDING      4 683.8 3.814813 676.1298 691.4702 (2.2)
## X13 SITTING      8 788.6 3.814813 780.9298 796.2702 (3.5)
## X14 STANDING      8 769.0 3.814813 761.3298 776.6702 (4.4)

```

```

graphRT3$vAdj = 35 #down
graphRT3$vAdj[graphRT3$setSize=="incongruent"]=35 #up
graphRT3$hAdj = 0 #right
#graphRT3$hAdj[graphRT3$posture=="SITTING"]=-60 #left

graphRT3$congruency = factor(graphRT3$setSize,labels = c("4","8"))

interactionPlot3 <- ggplot(graphRT3, aes(setSize, mean, group=posture)) +
  theme(legend.position = "none")+
  scale_fill_manual(values=c("#FFFFFF","999999","FFFFFF","999999"))+
  coord_cartesian(ylim=c(500,950),expand=TRUE)+
  scale_y_continuous(breaks = round(seq(500, 950, by = 100),0))+

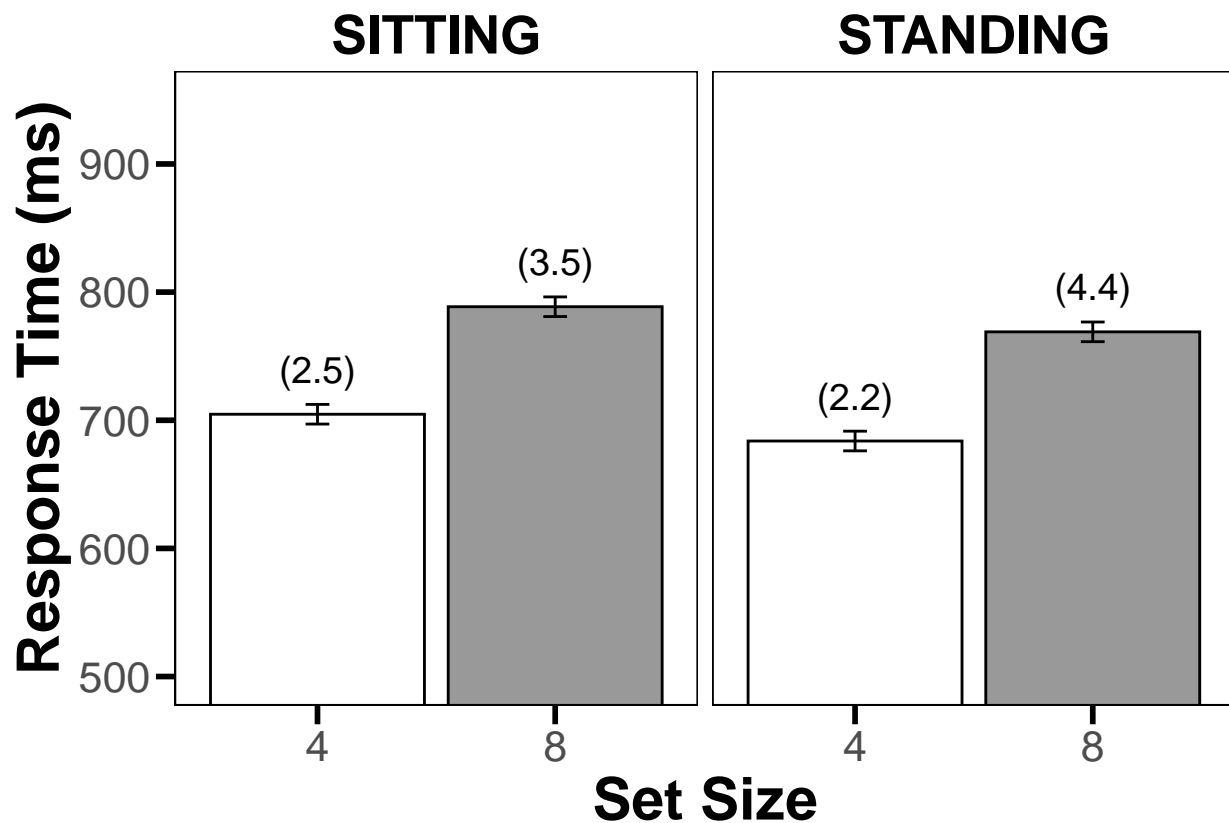
```

```

geom_text(aes(label=ac),nudge_x=graphRT3$hAdj,nudge_y =graphRT3$vAdj,size=5)+
geom_bar(stat="identity", aes(fill=interaction(setSize)),colour="black")+
geom_errorbar(aes(ymin=min,ymax=max,group=interaction(posture,setSize)), width=.1)+
labs(x = "Set Size", y = "Response Time (ms)") +
theme(axis.ticks = element_line(size = 1, colour = "black", linetype = "solid"),
      axis.ticks.length = unit(.25,"cm"),
      #axis.line = element_line(size = 1, colour = "black", linetype = "solid"),
      panel.background = element_rect(fill = "white", colour = "white", size = 1),
      axis.text=element_text(size=16),
      axis.title=element_text(size=22,face="bold"),
      strip.text = element_text(size = 20, face = "bold",colour = "black", angle = 0),
      panel.border = element_rect(colour = "black", fill = NA, size = 0.50),
      strip.background = element_rect(fill=NA,colour="NA",size = 2))+
facet_grid(~posture)

```

interactionPlot3



```

ggsave(interactionPlot3,
      file = "results/plots/fig4_exp3_visual_search_interaction_plot.pdf",
      units = "in",
      width = 8.5,
      height = 5,
      dpi = 600)

```

## Reproduce results from Smith et al.

```
# Data source: The data from all the experiments are available at:
# http://rabrams.net under the Resources tab.

### Experiment 1 (Stroop)

#load acc data
Smith_Exp1_acc <- read_excel("data/smith_data.xlsx",
                             sheet = "Exp1Acc",
                             n_max = 14) #Sample size in Smith

#load rt data
Smith_Exp1_rt <- read_excel("data/smith_data.xlsx",
                             sheet = "Exp1RT",
                             n_max = 14)

#Restructure from wide to narrow, using tidyr
Smith_Exp1_acc_narrow <- Smith_Exp1_acc %>%
  pivot_longer(cols = sit_neut:sta_con, names_to = "condition", values_to = "acc") %>%
  separate(col = condition, into = c("posture", "con"))

Smith_Exp1_rt_narrow <- Smith_Exp1_rt %>%
  pivot_longer(cols = sit_neut:sta_con, names_to = "condition", values_to = "rt") %>%
  separate(col = condition, into = c("posture", "con"))

Smith_Exp1 <- merge(Smith_Exp1_acc_narrow, Smith_Exp1_rt_narrow)

Smith_exp1_anova_acc <- aov_ez(data = Smith_Exp1,
                               dv = 'acc',
                               id = 'subj',
                               within = c('posture', 'con'),
                               anova_table = list(es = "pes", correction = "none"),
                               type = 3)
kable(nice(Smith_exp1_anova_acc), caption = "ANOVA results for Smith Exp 1 - accuracy")
```

Table 2: ANOVA results for Smith Exp 1 - accuracy

Effect	df	MSE	F	pes	p.value
posture	1, 13	4.59	0.51	.038	.488
con	2, 26	3.19	3.76 *	.224	.037
posture:con	2, 26	2.18	1.47	.101	.250

```
Smith_exp1_anova_rt <- aov_ez(data = Smith_Exp1,
                               dv = 'rt',
                               id = 'subj',
                               within = c('posture', 'con'),
                               anova_table = list(es = "pes", correction = "none"),
                               type = 3)
kable(nice(Smith_exp1_anova_rt), caption = "ANOVA results for Smith Exp 1 - RT")
```

Table 3: ANOVA results for Smith Exp 1 - RT

Effect	df	MSE	F	pes	p.value
posture	1, 13	816.34	0.09	.007	.768
con	2, 26	150.32	3.45 *	.210	.047
posture:con	2, 26	128.10	4.73 *	.267	.018

```
### Experiment 2 (Task-switching)

#load acc data
Smith_Exp2_acc <- read_excel("data/smith_data.xlsx",
                             sheet = "Exp2Acc",
                             n_max = 30)

#load rt data
Smith_Exp2_rt <- read_excel("data/smith_data.xlsx",
                             sheet = "Exp2RT",
                             n_max = 30)

#Restructure from wide to narrow, using tidyr
Smith_Exp2_acc_narrow <- Smith_Exp2_acc %>%
  pivot_longer(cols = sit_congruent_noswitch:stand_incongruent_switch,
               names_to = "condition", values_to = "acc") %>%
  separate(col = condition, into = c("posture", "con", "switch"))

Smith_Exp2_rt_narrow <- Smith_Exp2_rt %>%
  pivot_longer(cols = sit_congruent_noswitch:stand_incongruent_switch,
               names_to = "condition", values_to = "rt") %>%
  separate(col = condition, into = c("posture", "con", "switch"))

Smith_Exp2 <- merge(Smith_Exp2_acc_narrow, Smith_Exp2_rt_narrow)

Smith_exp2_anova_acc <- aov_ez(data = Smith_Exp2,
                               dv = 'acc',
                               id = 'subj',
                               within = c('posture', 'con', 'switch'),
                               anova_table = list(es = "pes", correction = "none"),
                               type = 3)

kable(nice(Smith_exp2_anova_acc), caption = "ANOVA results for Smith Exp 2 - accuracy")
```

Table 4: ANOVA results for Smith Exp 2 - accuracy

Effect	df	MSE	F	pes	p.value
posture	1, 29	0.00	2.86	.090	.101
con	1, 29	0.00	67.40 ***	.699	<.001
switch	1, 29	0.00	62.94 ***	.685	<.001
posture:con	1, 29	0.00	1.68	.055	.205
posture:switch	1, 29	0.00	5.54 *	.160	.026
con:switch	1, 29	0.00	23.34 ***	.446	<.001
posture:con:switch	1, 29	0.00	0.50	.017	.484



```
Smith_exp2_anova_rt <- aov_ez(data = Smith_Exp2,
  dv = 'rt',
  id = 'subj',
  within = c('posture', 'con', 'switch'),
  anova_table = list(es = "pes", correction = "none"),
  type = 3)
kable(nice(Smith_exp2_anova_rt), caption = "ANOVA results for Smith Exp 2 - RT")
```

Table 5: ANOVA results for Smith Exp 2 - RT

Effect	df	MSE	F	pes	p.value
posture	1, 29	0.02	0.03	.001	.856
con	1, 29	0.00	40.95 ***	.585	<.001
switch	1, 29	0.00	115.10 ***	.799	<.001
posture:con	1, 29	0.00	0.49	.017	.489
posture:switch	1, 29	0.00	0.10	.004	.751
con:switch	1, 29	0.00	4.77 *	.141	.037
posture:con:switch	1, 29	0.00	0.67	.023	.420

```
### Experiment 3 (Visual Search)

#load acc data
Smith_Exp3_acc <- read_excel("data/smith_data.xlsx",
  sheet = "Exp3Acc",
  n_max = 12) %>%
  select(subj:sit8)

#load rt data
Smith_Exp3_rt <- read_excel("data/smith_data.xlsx",
  sheet = "Exp3RT",
  n_max = 12)%>%
  select(subj:sit8)

#Restructure from wide to narrow, using tidyr
Smith_Exp3_acc_narrow <- Smith_Exp3_acc %>%
  pivot_longer(cols = stand4:sit8, names_to = "condition", values_to = "acc") %>%
  separate(col = condition, into = c("posture", "set.size"), sep = -1)

Smith_Exp3_rt_narrow <- Smith_Exp3_rt %>%
  pivot_longer(cols = stand4:sit8, names_to = "condition", values_to = "rt") %>%
  separate(col = condition, into = c("posture", "set.size"), sep = -1)

Smith_Exp3 <- merge(Smith_Exp3_acc_narrow, Smith_Exp3_rt_narrow)

Smith_exp3_anova_acc <- aov_ez(data = Smith_Exp3,
  dv = 'acc',
  id = 'subj',
  within = c('posture', 'set.size'),
  anova_table = list(es = "pes", correction = "none"),
  type = 3)
kable(nice(Smith_exp3_anova_acc), caption = "ANOVA results for Smith Exp 3 - accuracy")
```

Table 6: ANOVA results for Smith Exp 3 - accuracy

Effect	df	MSE	F	pes	p.value
posture	1, 11	4.61	0.76	.065	.401
set.size	1, 11	1.75	3.44 +	.238	.090
posture:set.size	1, 11	1.38	7.96 *	.420	.017

```
Smith_exp3_anova_rt <- aov_ez(data = Smith_Exp3,
                             dv = 'rt',
                             id = 'subj',
                             within = c('posture', 'set.size'),
                             anova_table = list(es = "pes", correction = "none"),
                             type = 3)
kable(nice(Smith_exp3_anova_rt), caption = "ANOVA results for Smith Exp 3 - RT")
```

Table 7: ANOVA results for Smith Exp 3 - RT

Effect	df	MSE	F	pes	p.value
posture	1, 11	2323.81	0.23	.021	.639
set.size	1, 11	473.24	81.88 ***	.882	<.001
posture:set.size	1, 11	298.96	5.91 *	.350	.033

## Overall summary plots: Smith and replication

```
smith_anovas <- lst(Smith_exp1_anova_acc$anova_table,
                   Smith_exp1_anova_rt$anova_table,
                   Smith_exp2_anova_acc$anova_table,
                   Smith_exp2_anova_rt$anova_table,
                   Smith_exp3_anova_acc$anova_table,
                   Smith_exp3_anova_rt$anova_table)

repl_anovas <- lst(aov_ez(data = stroopCombined,
                          dv = "meanPE",
                          id = "sj",
                          within = c("posture", "congruency"),
                          type = 3,
                          anova_table = list(es = "pes")),
                  aov_ez(data = stroopCombined,
                          dv = "meanRT",
                          id = "sj",
                          within = c("posture", "congruency"),
                          type = 3,
                          anova_table = list(es = "pes")),
                  accModelTS,
                  rtModelTS,
                  aov_ez(data = vsCombined,
                          dv = "meanPE",
                          id = "sj",
```

```

        within = c("postureFactor", "setSize"),
        type = 3,
        anova_table = list(es = "pes")),
    aov_ez(data = vsCombined,
        dv = "meanRT",
        id = "sj",
        within = c("postureFactor", "setSize"),
        type = 3,
        anova_table = list(es = "pes")))

for (i in 1:6){

  smith_anovas[[i]] <- smith_anovas[[i]] %>%
    rownames_to_column() %>%
    as.data.frame() %>%
    rowwise() %>%
    mutate(LL = get.ci.partial.eta.squared(F, `num Df`, `den Df`, conf.level = 0.9)$LL,
        UL = get.ci.partial.eta.squared(F, `num Df`, `den Df`, conf.level = 0.9)$UL)

  repl_anovas[[i]] <- repl_anovas[[i]]$anova_table %>%
    rownames_to_column() %>%
    as.data.frame() %>%
    rowwise() %>%
    mutate(LL = get.ci.partial.eta.squared(F, `num Df`, `den Df`, conf.level = 0.9)$LL,
        UL = get.ci.partial.eta.squared(F, `num Df`, `den Df`, conf.level = 0.9)$UL)
}

###Exp1 (Stroop)
smith.stroop <- smith_anovas[[1]] %>%
  ungroup() %>%
  bind_rows(smith_anovas[[2]]) %>%
  select(Effect = rowname, pes, LL, UL) %>%
  mutate(dv = rep(c("acc", "rt"), each = 3), col = rep(c("black", "black", "red"), 2))

repl.stroop <- repl_anovas[[1]] %>%
  ungroup() %>%
  bind_rows(repl_anovas[[2]]) %>%
  select(Effect = rowname, pes, LL, UL) %>%
  mutate(dv = rep(c("acc", "rt"), each = 3), col = rep(c("black", "black", "red"), 2),
        Effect = smith.stroop$Effect)

stroop.effects <- merge(smith.stroop, repl.stroop,
  by = c("Effect", "dv"), suffixes = c("Smith", "Replication"))

stroop.plot <- ggplot(data = stroop.effects, aes(x = pesSmith, y = pesReplication, shape = dv)) +
  geom_point(size = 2.5, col = stroop.effects$colSmith) +
  xlim(0, 1.00) +
  ylim(0, 1.00) +
  geom_abline(slope = 1, intercept = 0, col = "blue") +
  theme_classic() +
  theme(legend.position = c(0.2, 0.85),

```

```

    legend.background = element_rect(colour = "black",
                                     linetype = "solid",
                                     fill = "lightgray"),

    legend.title = element_blank(),
    legend.margin=margin(-3,5,0,0)) +
labs(y = "Replication", x = "Smith et al.", title = "Stroop")

###Exp2 (Task-switching)
smith.ts <- smith_anovas[[3]] %>%
  ungroup() %>%
  bind_rows(smith_anovas[[4]]) %>%
  select(Effect = rowname, pes, LL, UL) %>%
  mutate(dv = rep(c("acc","rt"), each = 7),
         col = rep(c("black","black","black","black","red","black","black"),2))

repl.ts <- repl_anovas[[3]] %>%
  ungroup() %>%
  bind_rows(repl_anovas[[4]]) %>%
  select(Effect = rowname, pes, LL, UL) %>%
  mutate(dv = rep(c("acc","rt"), each = 7),
         col = rep(c("black","black","black","black","red","black","black"),2),
         Effect = smith.ts$Effect)

ts.effects <- merge(smith.ts, repl.ts,
                   by = c("Effect","dv"), suffixes = c("Smith","Replication"))

ts.plot <- ggplot(data = ts.effects, aes(x = pesSmith, y = pesReplication, shape = dv)) +
  geom_point(size = 2.5, col = ts.effects$colSmith) +
  xlim(0, 1.00) +
  ylim(0, 1.00) +
  geom_abline(slope = 1, intercept = 0, col = "blue") +
  theme_classic() +
  theme(legend.position = c(0.2, 0.85),
        legend.background = element_rect(colour = "black",
                                           linetype = "solid",
                                           fill = "lightgray"),

        legend.title = element_blank(),
        legend.margin=margin(-3,5,0,0)) +
labs(y = "Replication", x = "Smith et al.", title = "Task-Switching")

###Exp3 (Visual Search)
smith.vs <- smith_anovas[[5]] %>%
  ungroup() %>%
  bind_rows(smith_anovas[[6]]) %>%
  select(Effect = rowname, pes, LL, UL) %>%
  mutate(dv = rep(c("acc","rt"), each = 3),
         col = rep(c("black","black","red"),2))

repl.vs <- repl_anovas[[5]] %>%
  ungroup() %>%
  bind_rows(repl_anovas[[6]]) %>%

```

```

select(Effect = rowname, pes, LL, UL) %>%
mutate(dv = rep(c("acc","rt"), each = 3),
       col = rep(c("black","black","red"),2),
       Effect = smith.vs$Effect)

vs.effects <- merge(smith.vs, repl.vs,
                   by = c("Effect","dv"), suffixes = c("Smith",
                                                         "Replication"))

vs.plot <- ggplot(data = vs.effects, aes(x = pesSmith, y = pesReplication, shape = dv)) +
  geom_point(size = 2.5, col = vs.effects$colSmith) +
  xlim(0, 1) +
  ylim(0, 1) +
  geom_abline(slope = 1, intercept = 0, col = "blue") +
  theme_classic() +
  theme(legend.position = c(0.2, 0.85),
        legend.background = element_rect(colour = "black",
                                           linetype = "solid",
                                           fill = "lightgray"),
        legend.title = element_blank(),
        legend.margin=margin(-3,5,0,0)) +
  labs(y = "Replication", x = "Smith et al.", title = "Visual Search")

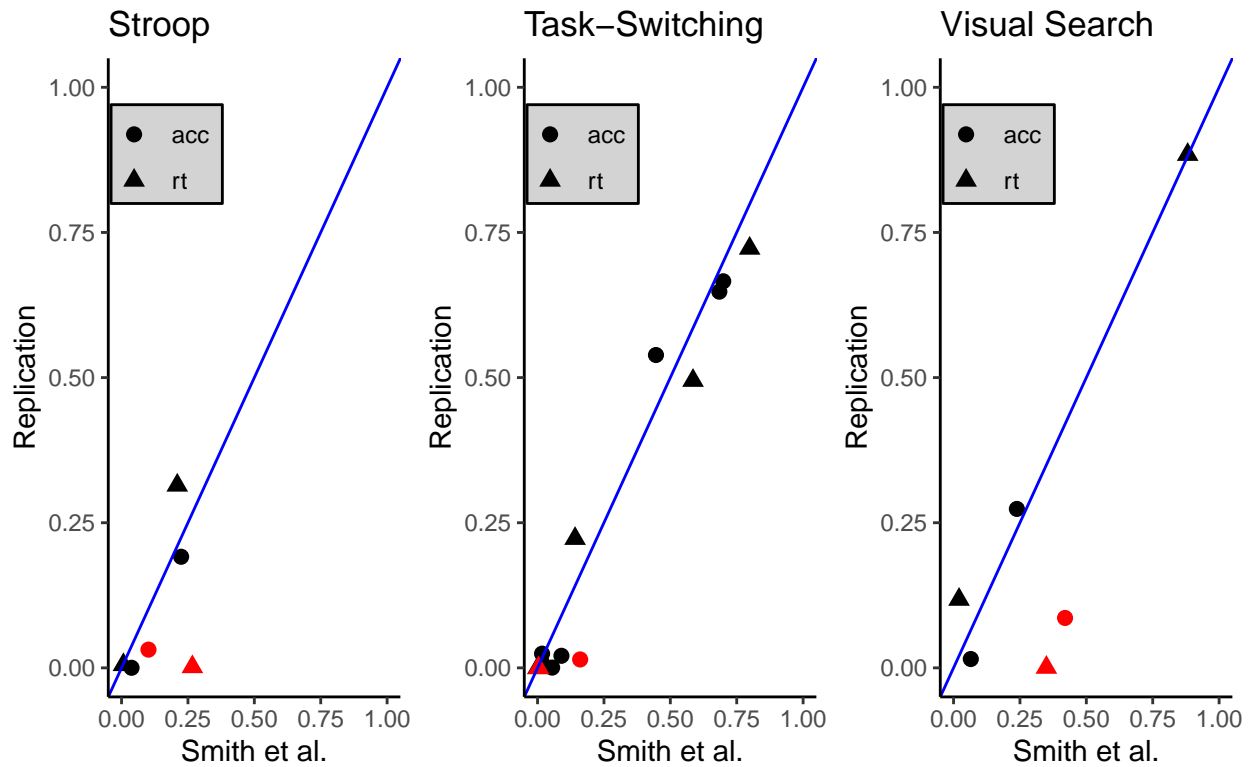
all.plot <- plot_grid(stroop.plot, ts.plot, vs.plot, ncol = 3)

title <- ggdraw() +
  draw_label(
    "Effect Size Comparisons",
    fontface = 'bold',
    x = 0,
    hjust = 0
  ) +
  theme(
    # add margin on the left of the drawing canvas,
    # so title is aligned with left edge of first plot
    plot.margin = margin(0, 0, 0, 7)
  )
all.plot <- plot_grid(
  title, all.plot,
  ncol = 1,
  # rel_heights values control vertical title margins
  rel_heights = c(0.1, 1)
)

all.plot

```

## Effect Size Comparisons



```
ggsave(all.plot,
  file = "results/plots/supp_all_effects_plot.pdf",
  units = "in",
  width = 9.5,
  height = 4.50,
  dpi = 600)

### Forest plot
#Graph comparison of key effects for all three experiments
forest.colors <- c("black", "red")

forest.data <- data.frame(Experiment = rep(c("Smith", "Replication"), 3),
  name = rep(c("Stroop", "Task-switching", "Visual Search"), each = 2),
  dv = rep(c("rt", "acc", "rt"), each = 2),
  pes = numeric(6),
  LL = numeric(6),
  UL = numeric(6))

forest.data[1,4:6] <- smith_anovas[[2]][3,c(6,8,9)]
forest.data[2,4:6] <- repl_anovas[[2]][3,c(6,8,9)]

forest.data[3,4:6] <- smith_anovas[[3]][5,c(6,8,9)]
forest.data[4,4:6] <- repl_anovas[[3]][5,c(6,8,9)]

forest.data[5,4:6] <- smith_anovas[[6]][3,c(6,8,9)]
forest.data[6,4:6] <- repl_anovas[[6]][3,c(6,8,9)]
```

```

forest.comp <- mod.forestplot(df = forest.data,
                             estimate = pes,
                             ci.lower = LL,
                             ci.upper = UL,
                             colour = Experiment,
                             xlab = "Partial Eta-Squared"
) +
  scale_color_manual(values = forest.colors) +
  scale_x_continuous(labels = label_number(accuracy = 0.01), breaks = seq(0.00, 0.60, 0.10)) +
  #Too busy w/numbers for effects?
  geom_label(data = subset(forest.data, Experiment == "Smith"),
             aes(label = round(pes, digits = 2))) +
  geom_label(data = subset(forest.data, Experiment == "Replication"),
             aes(label = round(pes, digits = 3)),
             vjust = 2.50) +
  theme(axis.text=element_text(size=14),
        panel.background = element_rect(fill = "white", colour = "white", size = 1),
        axis.title = element_text(size=12, face="bold"),
        strip.text = element_text(size = 12, face = "bold", colour = "black", angle = 0),
        legend.text = element_text(size = 14),
        legend.title = element_text(size= 14, face = "bold")) +
  coord_cartesian(clip="off") #Disable clipping to draw outside plot area

```

```

## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.

```

```

## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.

```

```

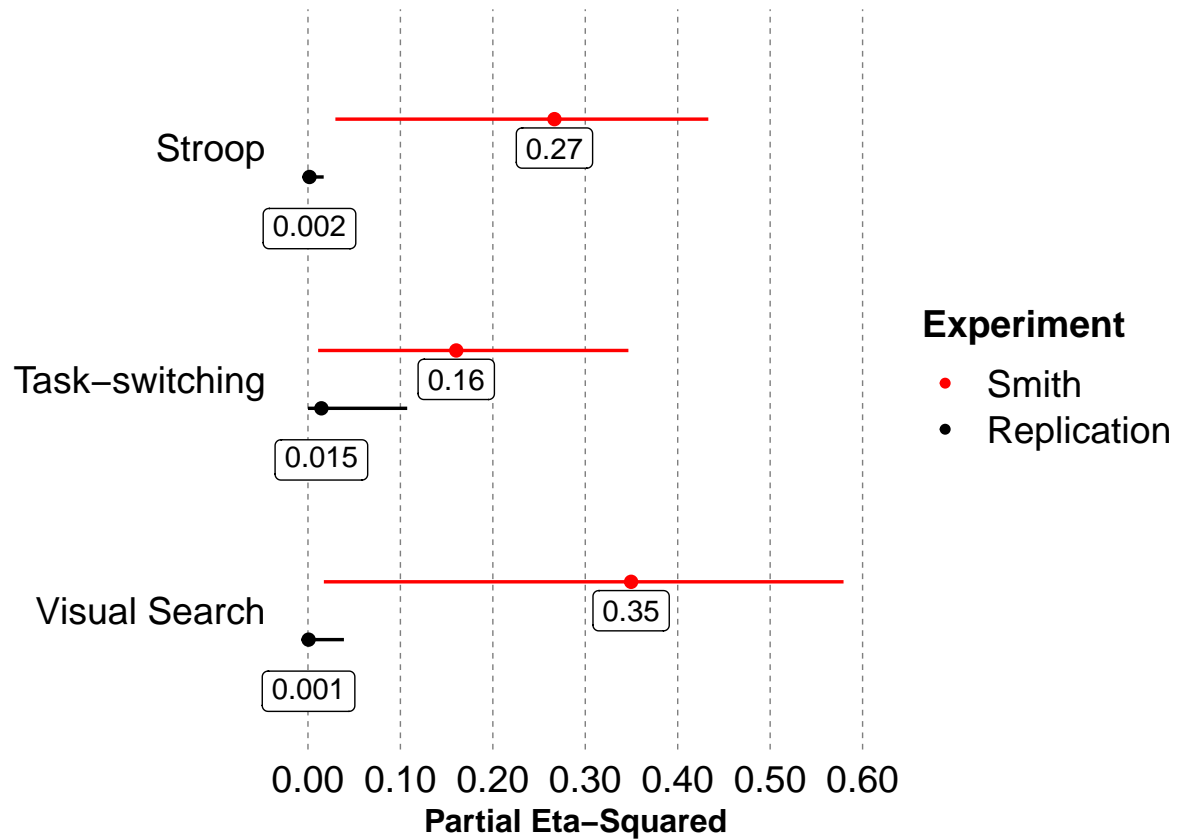
forest.comp

```

```

## Warning: Using the 'size' aesthetic with geom_segment was deprecated in ggplot2 3.4.0.
## i Please use the 'linewidth' aesthetic instead.

```



```
ggsave(forest.comp,
  file = "results/plots/fig5_forest_plot.pdf",
  units = "in",
  width = 7,
  height = 5,
  dpi = 600)

#Compare proportions: replication divided by original effect sizes
replication.effects <- subset(forest.data, Experiment == "Replication")
original.effects    <- subset(forest.data, Experiment == "Smith")

#As a percentage
prop.effects <- (replication.effects$pes/original.effects$pes)*100
#< 1%, ~9%, and <1%
prop.effects
```

```
## [1] 0.6181838 9.0496059 0.1801278
```

```
#Average proportion is 3.28%
mean(prop.effects)
```

```
## [1] 3.282639
```



## Appendix: Analysis with conventional dropping of timeout trials (only affects PE analysis)

### Experiment 1: Stroop

```
e1.conv.PE = plyr::ddply(mergedStroopData.conventional,
                        .(sj, cb, congruency, posture),
                        summarise,
                        meanPE = 100 - (mean(ac)*100))

e1.conv.PE$posture = factor(e1.conv.PE$posture)
e1.conv.PE$congruency = factor(e1.conv.PE$congruency)
e1.conv.PE$sj = factor(e1.conv.PE$sj)

e1.conv.errmodel <- ezANOVA(e1.conv.PE,
                           dv = .(meanPE),
                           wid=.(sj),
                           within=.(posture, congruency),
                           detailed=TRUE,
                           type=3,
                           return_aov = TRUE)

e1.conv.MSE = e1.conv.errmodel$ANOVA$SSd/e1.conv.errmodel$ANOVA$DFd

exp1_ANOVA_PE_con <-
paste(e1.conv.errmodel$ANOVA$Effect, ": F(",
      e1.conv.errmodel$ANOVA$DFn,
      ", ",
      e1.conv.errmodel$ANOVA$DFd,
      ") = ",
      round(e1.conv.errmodel$ANOVA$F, 3),
      ", MSE = ",
      round(e1.conv.MSE, 3),
      ", p = ",
      round(e1.conv.errmodel$ANOVA$p, 3),
      ", partialEtaSq = ",
      round(e1.conv.errmodel$ANOVA$SSn/(e1.conv.errmodel$ANOVA$SSn+e1.conv.errmodel$ANOVA$SSd), 4),
      sep="")
exp1_ANOVA_PE_con

## [1] "(Intercept): F(1, 49) = 54.682, MSE = 69.145, p = 0, partialEtaSq = 0.5274"
## [2] "posture: F(1, 49) = 0.028, MSE = 14.823, p = 0.868, partialEtaSq = 6e-04"
## [3] "congruency: F(2, 98) = 8.813, MSE = 9.641, p = 0, partialEtaSq = 0.1524"
## [4] "posture:congruency: F(2, 98) = 1.533, MSE = 5.865, p = 0.221, partialEtaSq = 0.0303"

write.csv(exp1_ANOVA_PE_con, "results/exp1_ANOVA_PE_con.csv")

#BF for errors
e1.conv.BF = anovaBF(meanPE~congruency*posture+sj,
```

```

                                data = e1.conv.PE,
                                whichRandom = "sj",
                                method="laplace")
e1.conv.BF

## Bayes factor analysis
## -----
## [1] congruency + sj           : 136.8983 ±NA%
## [2] posture + sj            : 0.1188895 ±NA%
## [3] congruency + posture + sj : 16.39088 ±NA%
## [4] congruency + posture + congruency:posture + sj : 2.363896 ±NA%
##
## Against denominator:
##   meanPE ~ sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

```

##...get the Bayes factor for the Null Interaction
e1.conv.BF[3]/e1.conv.BF[4]

```

```

## Bayes factor analysis
## -----
## [1] congruency + posture + sj : 6.933838 ±NA%
##
## Against denominator:
##   meanPE ~ congruency + posture + congruency:posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

```

1/(e1.conv.BF[3]/e1.conv.BF[4])

```

```

## Bayes factor analysis
## -----
## [1] congruency + posture + congruency:posture + sj : 0.1442203 ±NA%
##
## Against denominator:
##   meanPE ~ congruency + posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS

```

## Experiment 2: Stroop

```

e2.conv.PE <- task_switching_raw.conventional %>%
  group_by(participant,
            posture,
            congruentTrialType,
            switchTrialType) %>%
  summarize(meanPE = (1 - correct_bin) * 100)

```

```

## 'summarise()' has grouped output by 'participant', 'posture', 'congruentTrialType',
## 'switchTrialType'. You can override using the '.groups' argument.

```

```
e2.conv.PE$posture = factor(e2.conv.PE$posture)
e2.conv.PE$congruentTrialType = factor(e2.conv.PE$congruentTrialType)
e2.conv.PE$participant = factor(e2.conv.PE$participant)
e2.conv.PE$switchTrialType = factor(e2.conv.PE$switchTrialType)

e2.conv.errmodel <- ezANOVA(e2.conv.PE,
  dv = .(meanPE),
  wid=.(participant),
  within=.(posture,congruentTrialType, switchTrialType),
  detailed=TRUE,
  type=3,
  return_aov = TRUE)
```

## Warning: Collapsing data to cell means. \*IF\* the requested effects are a subset of the full design,  
## you must use the "within\_full" argument, else results may be inaccurate.

```
e2.conv.MSE = e2.conv.errmodel$ANOVA$SSd/e2.conv.errmodel$ANOVA$DFd

exp2_ANOVA_PE_con <-
paste(e2.conv.errmodel$ANOVA$Effect,": F(",
  e2.conv.errmodel$ANOVA$DFn,
  ", ",
  e2.conv.errmodel$ANOVA$DFd,
  ") = ",
  round(e1.conv.errmodel$ANOVA$F,3),
  ", MSE = ",
  round(e2.conv.MSE,3),
  ", p = ",
  round(e2.conv.errmodel$ANOVA$p,3),
  ", partialEtaSq = ",
  round(e2.conv.errmodel$ANOVA$SSn/(e2.conv.errmodel$ANOVA$SSn+e2.conv.errmodel$ANOVA$SSd),4),
  sep="")
exp2_ANOVA_PE_con
```

```
## [1] "(Intercept): F(1, 50) = 54.682, MSE = 124.788, p = 0, partialEtaSq = 0.8018"
## [2] "posture: F(1, 50) = 0.028, MSE = 77.648, p = 0.425, partialEtaSq = 0.0128"
## [3] "congruentTrialType: F(1, 50) = 8.813, MSE = 76.694, p = 0, partialEtaSq = 0.6854"
## [4] "switchTrialType: F(1, 50) = 1.533, MSE = 21.856, p = 0, partialEtaSq = 0.6665"
## [5] "posture:congruentTrialType: F(1, 50) = 54.682, MSE = 38.811, p = 0.939, partialEtaSq = 1e-04"
## [6] "posture:switchTrialType: F(1, 50) = 0.028, MSE = 18.608, p = 0.243, partialEtaSq = 0.0272"
## [7] "congruentTrialType:switchTrialType: F(1, 50) = 8.813, MSE = 13.78, p = 0, partialEtaSq = 0.560"
## [8] "posture:congruentTrialType:switchTrialType: F(1, 50) = 1.533, MSE = 13.555, p = 0.491, partial"
```

```
write.csv(exp2_ANOVA_PE_con, "results/exp2_ANOVA_PE_con.csv")
```

*#BF for errors*

```
e2.conv.BF = anovaBF(meanPE~congruentTrialType*posture+participant,
  data = e2.conv.PE,
  whichRandom = "participant",
  method="laplace")
```

## Warning: data coerced from tibble to data frame

```
e2.conv.BF
```

```
## Bayes factor analysis
## -----
## [1] posture + participant : 0.06825202 ±NA%
## [2] congruentTrialType + participant : 7.659849e+119 ±NA%
## [3] posture + congruentTrialType + participant : 5.611788e+118 ±NA%
## [4] posture + congruentTrialType + posture:congruentTrialType + participant : 1.187856e+117 ±NA%
##
## Against denominator:
## meanPE ~ participant
## ---
## Bayes factor type: BFlinearModel, JZS

##...get the Bayes factor for the Null Interaction
e2.conv.BF[3]/e2.conv.BF[4]
```

```
## Bayes factor analysis
## -----
## [1] posture + congruentTrialType + participant : 47.24301 ±NA%
##
## Against denominator:
## meanPE ~ posture + congruentTrialType + posture:congruentTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
1/(e2.conv.BF[3]/e2.conv.BF[4])
```

```
## Bayes factor analysis
## -----
## [1] posture + congruentTrialType + posture:congruentTrialType + participant : 0.02116715 ±NA%
##
## Against denominator:
## meanPE ~ posture + congruentTrialType + participant
## ---
## Bayes factor type: BFlinearModel, JZS
```

## Experiment 3: Visual Search

```
e3.conv.PE = plyr::ddply(merged.VS.data.conventional,
                        .(sj, setSize, posture),
                        summarise,
                        meanPE = 100 - (mean(ac)*100))

e3.conv.PE$posture = factor(e3.conv.PE$posture)
e3.conv.PE$setSize = factor(e3.conv.PE$setSize)
e3.conv.PE$sj = factor(e3.conv.PE$sj)

e3.conv.errmodel <- ezANOVA(e3.conv.PE,
```

```

      dv = .(meanPE),
      wid=.(sj),
      within=.(posture, setSize),
      detailed=TRUE,
      type=3,
      return_aov = TRUE)

e3.conv.MSE = e3.conv.errmodel$ANOVA$SSd/e3.conv.errmodel$ANOVA$DFd

exp3_ANOVA_PE_con <-
paste(e3.conv.errmodel$ANOVA$Effect,": F(",
      e3.conv.errmodel$ANOVA$DFn,
      ", ",
      e3.conv.errmodel$ANOVA$DFd,
      ") = ",
      round(e3.conv.errmodel$ANOVA$F,3),
      ", MSE = ",
      round(e3.conv.MSE,3),
      ", p = ",
      round(e3.conv.errmodel$ANOVA$p,3),
      ", partialEtaSq = ",
      round(e3.conv.errmodel$ANOVA$SSn/(e3.conv.errmodel$ANOVA$SSn+e3.conv.errmodel$ANOVA$SSd),4),
      sep="")

exp3_ANOVA_PE_con

## [1] "(Intercept): F(1, 49) = 55.236, MSE = 12.588, p = 0, partialEtaSq = 0.5299"
## [2] "posture: F(1, 49) = 3.311, MSE = 2.43, p = 0.075, partialEtaSq = 0.0633"
## [3] "setSize: F(1, 49) = 2.312, MSE = 3.628, p = 0.135, partialEtaSq = 0.0451"
## [4] "posture:setSize: F(1, 49) = 0.623, MSE = 2.794, p = 0.434, partialEtaSq = 0.0125"

write.csv(exp3_ANOVA_PE_con, "results/exp3_ANOVA_PE_con.csv")

graphPE3.con = describeBy(e3.conv.PE$meanPE,
                          list(e3.conv.PE$posture, e3.conv.PE$setSize),
                          mat=TRUE,
                          digits = 1)

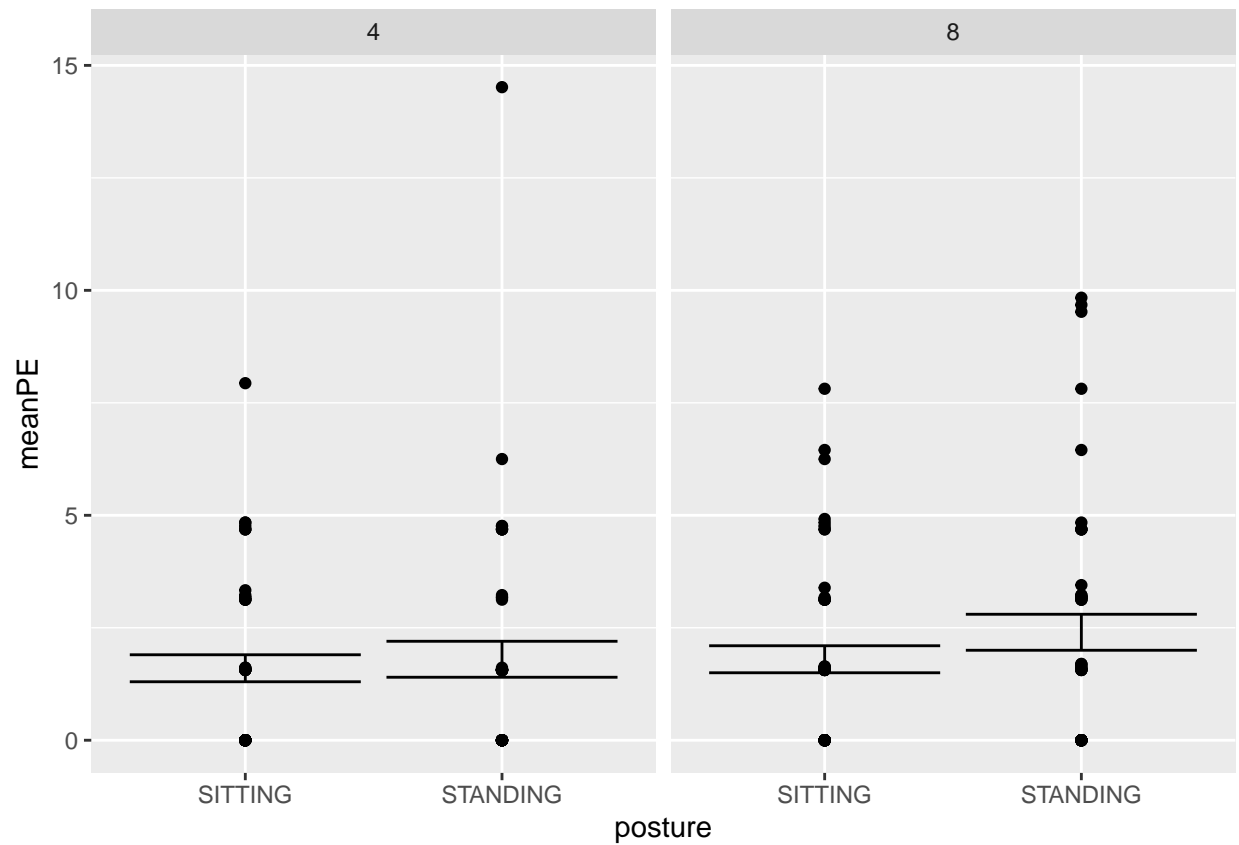
graphPE3.con = graphPE3.con[,c("group1","group2","mean", "sd", "se")]

names(graphPE3.con) = c("posture","setSize","meanPE", "sd", "se")

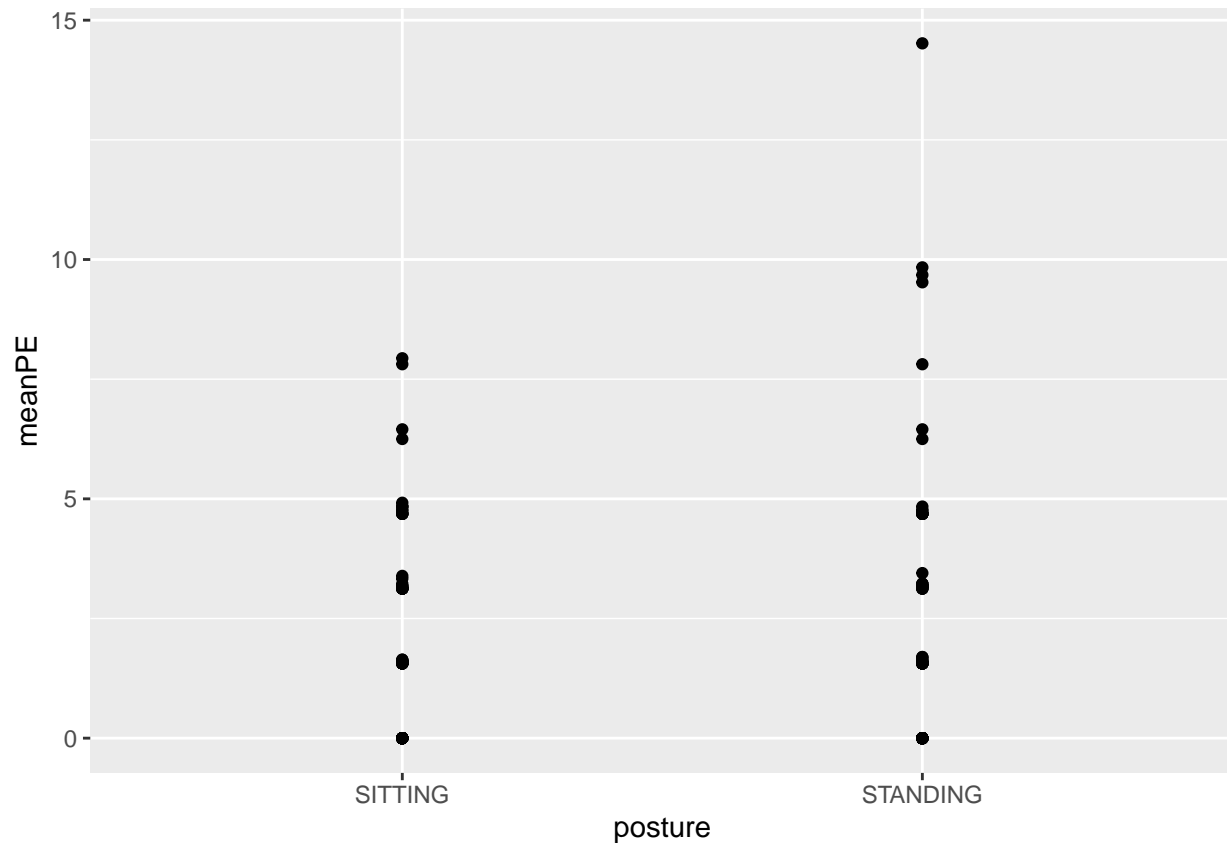
e3.PE.conv.plot <-
ggplot(data = e3.conv.PE, aes(x = posture,
                             y = meanPE)) +
  geom_point() +
  facet_wrap(~setSize) +
  geom_errorbar(data = graphPE3.con, aes(ymin = meanPE - se, ymax = meanPE + se))

e3.PE.conv.plot

```



```
#Standing has a single cell mean, set size = 4, nearly 15 percent error
e3.PE.conv.plot2 <-
ggplot(data = e3.conv.PE, aes(x = posture,
                             y = meanPE)) +
  geom_point()
e3.PE.conv.plot2
```



```
#hist(e3.conv.PE$meanPE)
```

```
#BF for errors
```

```
e3.conv.BF = anovaBF(meanPE~setSize*posture+sj,
                      data = e3.conv.PE,
                      whichRandom = "sj",
                      method="laplace")
```

```
e3.conv.BF
```

```
## Bayes factor analysis
```

```
## -----
```

```
## [1] setSize + sj : 0.5212586 ±NA%
```

```
## [2] posture + sj : 0.4937759 ±NA%
```

```
## [3] setSize + posture + sj : 0.2659863 ±NA%
```

```
## [4] setSize + posture + setSize:posture + sj : 0.06740099 ±NA%
```

```
##
```

```
## Against denominator:
```

```
## meanPE ~ sj
```

```
## ---
```

```
## Bayes factor type: BFlinearModel, JZS
```

```
##...get the Bayes factor for the Null Interaction
```

```
e3.conv.BF[3]/e3.conv.BF[4]
```

```
## Bayes factor analysis
```

```
## -----
## [1] setSize + posture + sj : 3.946327 ±NA%
##
## Against denominator:
##   meanPE ~ setSize + posture + setSize:posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS

1/(e3.conv.BF[3]/e3.conv.BF[4])

## Bayes factor analysis
## -----
## [1] setSize + posture + setSize:posture + sj : 0.2534002 ±NA%
##
## Against denominator:
##   meanPE ~ setSize + posture + sj
## ---
## Bayes factor type: BFlinearModel, JZS
```

## Effect of different RT cutoffs on PE interaction

```
#compare different criteria for removing long trials
criteria <- seq(1000, 2000, by = 100)
ncriteria <- length(criteria)
criteria.dat <- merged.VS.data.orig

for (i in criteria){
  criteria.dat <- criteria.dat %>%
    mutate("drop{i}" := ifelse(rt >= i | ac == 0, 0, 1))
}

vsPEcriteria = criteria.dat %>%
  group_by(sj, cb, setSize, posture) %>%
  summarize(across(starts_with("drop"), ~ 100 - (mean(.x)*100)))
```

## 'summarise()' has grouped output by 'sj', 'cb', 'setSize'. You can override using the '.groups' argument.

```
#...set as factors
vsPEcriteria$sj = factor(vsPEcriteria$sj)
vsPEcriteria$cb = factor(vsPEcriteria$cb)
vsPEcriteria$setSize = factor(vsPEcriteria$setSize)
vsPEcriteria$postureFactor = factor(vsPEcriteria$posture)

criteria.results <- data.frame(cutoff = criteria,
                              es = numeric(ncriteria),
                              p.vals = numeric(ncriteria),
                              ci.LL = numeric(ncriteria),
                              ci.UL = numeric(ncriteria))

for (i in 1:length(criteria)){
```



```

tempcol <- paste0("drop",criteria[i])
tempmod <- aov_ez(data = vsPEcriteria,
                  dv = tempcol,
                  id = 'sj',
                  within = c('posture', 'setSize'),
                  anova_table = list(es = "pes", correction = "none"),
                  type = 3)

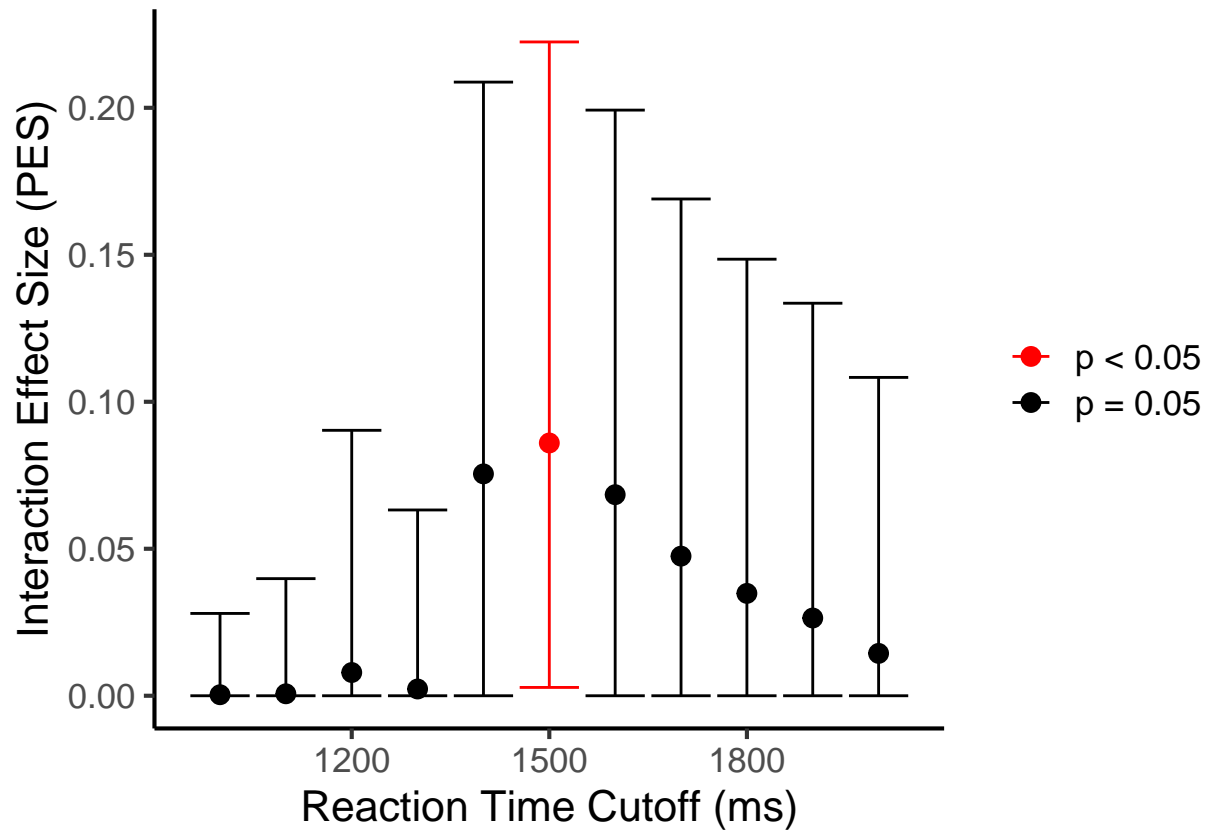
criteria.results$es[i] <- tempmod$anova_table$pes[3]
criteria.results$p.vals[i] <- tempmod$anova_table$`Pr(>F)`[3]
criteria.results$ci.LL[i] <- get.ci.partial.eta.squared(F.value=tempmod$anova_table$F[3],
                                                       df1=tempmod$anova_table$num Df`[3],
                                                       df2 = tempmod$anova_table$den Df`[3])$LL
criteria.results$ci.UL[i] <- get.ci.partial.eta.squared(F.value=tempmod$anova_table$F[3],
                                                       df1=tempmod$anova_table$num Df`[3],
                                                       df2 = tempmod$anova_table$den Df`[3])$UL
}

criteria.results$sig <- ifelse(criteria.results$p.vals < 0.05, "p < 0.05", "p \u2265 0.05")

exp3_rt_cutoff <-
ggplot(criteria.results, aes(criteria, es, col = sig)) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = ci.LL, ymax = ci.UL)) +
  scale_color_manual(values = c("red","black")) +
  theme_classic(base_size = 16) +
  theme(legend.title = element_blank()) +
  labs(y = "Interaction Effect Size (PES)", x = "Reaction Time Cutoff (ms)")

exp3_rt_cutoff

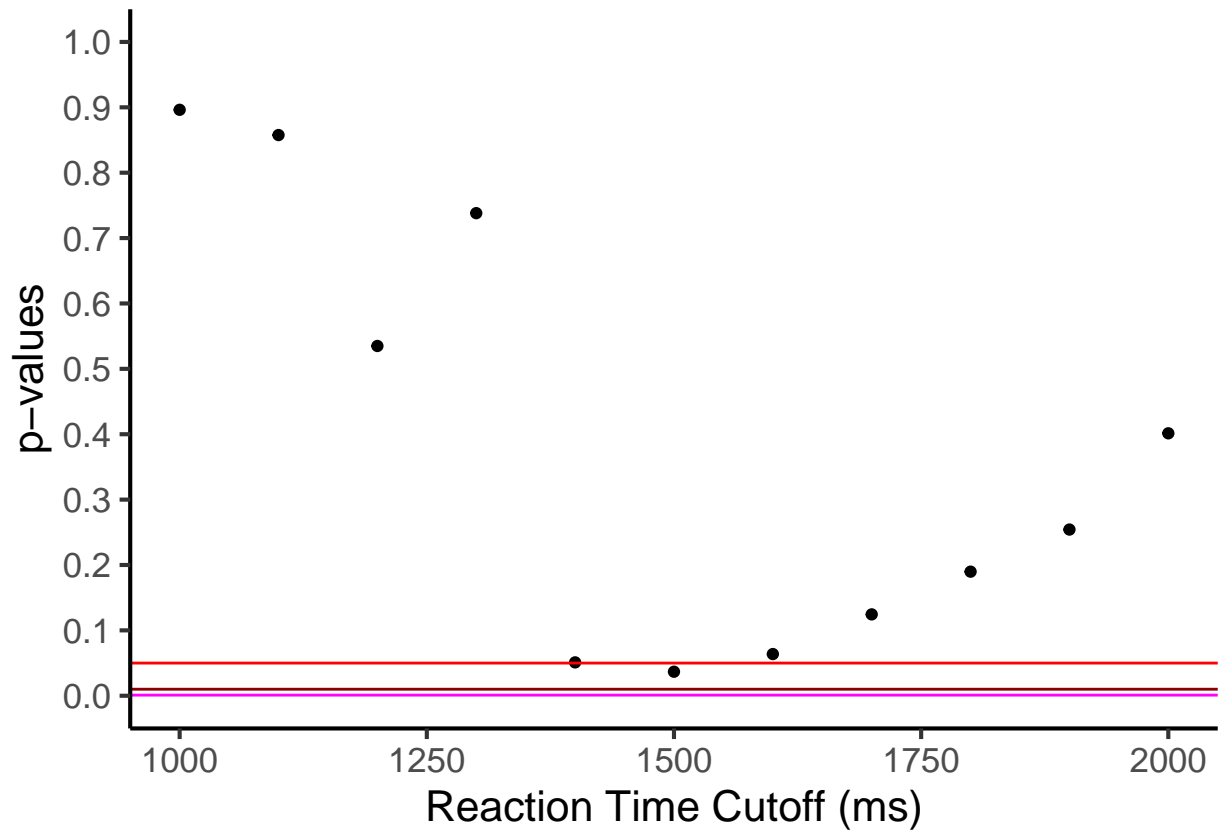
```



```
ggsave(exp3_rt_cutoff,
  file = "results/plots/supp_exp3_RT_cutoff.pdf",
  units = "in",
  width = 9,
  height = 6,
  dpi = 600)

#Not in the supp: Distribution of p-values
exp3_rt_cutoff.pvals <-
ggplot(criteria.results, aes(criteria, p.vals)) +
  geom_point(size = 1.5) +
  scale_color_manual(values = c("red", "black")) +
  theme_classic(base_size = 16) +
  theme(legend.title = element_blank()) +
  scale_y_continuous(limits = c(0, 1), breaks = seq(0, 1, 0.1)) +
  labs(y = "p-values", x = "Reaction Time Cutoff (ms)")

exp3_rt_cutoff.pvals + geom_hline(yintercept = 0.05, color = "red") +
  geom_hline(yintercept = 0.01, color = "darkred") +
  geom_hline(yintercept = 0.001, color = "magenta")
```



```
#plot(criteria.results$p.vals, criteria.results$es)
```