

```

1 # ANOVA_tracking error.r - R Code for data analysis for CHI 2010 paper by Hornof et al.
2 # R stats scripts for the EPIC NRL dual task model by Hornof.
3 # R scripts originally written by Yunfeng Zhang - 1/5/2010
4 # Backed up in: Dual Task/Analysis/zArchived Analyses/
5 # 3_09 Fall R (no transformation and use VizFix data V1)/scripts/ANOVA_tracking error.r
6
7 # Updated by Hornof (ajh) in 2019.
8
9 # This script creates tables including:
10 #   TEdata: All tracking errors, recorded once ever 83ms.
11 #   MeanTE: means by Participant and condition.
12 # It no longer uses these tables:
13 #   allTEDataTrim2sd: allTEData, but without TES over 2 standard deviations from the mean.
14 #   allTEDataTrim200: allTEData, but with no TES over 200 pixels. Used for CHI 2010 graph.
15
16 # I could create a new .Rdata file such as by doing the following:
17 # save (TEdata, MeanTE, file="TE.Rdata")
18 # But it takes less disk space to just recreate tables above for every use. (ajh)
19
20 # -----
21 # Load packages and data.
22 # -----
23
24 dividing_line =
25 "-----\n"
26
27 # load dplyr for rename()
28 library(dplyr)
29 source("../scripts/graphFunctions.r") # For CI_half_width( )
30 # Load the function removeTwoPsDay3( ) to remove P14 & P15.
31 source("../scripts/importDataUtilities.r")
32
33 # -----
34
35 # Set the working directory (ajh)
36 # setwd("~/1. Synchronized Data/2. Work/1. Research/1. Active Projects/2. Kieras & Yunfeng
37 # Zhang Collaboration (DK YZ HZ)/2019 - Running ICCM 2010 models using EPIC 2019/CHI 2010
38 # analysis, revisited and updated")
39
40 # YZ 1/5/2010
41 # import blips data. Taken from importData.R.
42
43 # Read the table and remove bad participants.
44 blips <- removeTwoPsDay3(read.table("../data/classification time.txt", header = T))
45 blips$order <- ordered(as.character(blips$order), levels = c("1", "2", "3", "4"))
46 blips$day <- factor(as.character(blips$day))
47 blips$error.type <- factor(as.character(blips$error.type))
48
49 # ajh changed 2019
50 # levels(blips$gaze) <- c("g.off", "g.on")
51 # levels(blips$sound) <- c("s.off", "s.on")
52 levels(blips$gaze) <- c("no", "yes") # "Yes" means gaze contingency was on.
53 levels(blips$sound) <- c("no", "yes") # "Yes" means sounds played for blip events.
54
55 # import tracking error data.
56 TEdata <- removeTwoPsDay3(read.table("../data/all tracking error.txt", header = T))
57 TEdata <- merge(TEdata, unique(blips[c("id", "day", "order", "sound", "gaze")] ), all.x=T)
58
59 # aggregate TE
60 MeanTE <- as.data.frame(ftable(
61   with(TEdata, tapply(TE, list(id, day, sound, gaze), mean)))
62 names(MeanTE) <- c("id", "day", "sound", "gaze", "TE")
63
64 rm (blips, MeanTE) # Cleanup. (Remove temporary objects to avoid confusion.)
65
66 # -----
67 # Compute the Tracking Error (TE)
68 # -----
69
70 # Generate the tracking errors (a) for Day 3 only (b) without removing any "outliers"
71 # and (c) after first averaging by participant.

```

```

69
70 # Limit the analyses to Day 3.
71 TEdata <- filter(TEdata, day==3)
72 # Rename the "id" column to "Participant"
73 TEdata <- rename(TEdata, Participant=id)
74
75 #####
76 # TEMPORARILY REMOVE P04 & P14 & P15
77 #####
78
79 # TEdata <- filter(TEdata, Participant != "P04")
80 # TEdata <- filter(TEdata, Participant != "P14")
81 # TEdata <- filter(TEdata, Participant != "P15")
82
83
84 # First create a table of the participant means.
85 TE_P_means <- aggregate(x = list(TE = TEdata$TE), # "list(TE=...)" renames Column "x"
86                         by = list(
87                             Sound = TEdata$sound, # "Sound = " names the column.
88                             Gaze = TEdata$gaze,
89                             Participant = TEdata$Participant
90                         ),
91                         FUN = mean)
92
93
94 # Get the means of the participant means.
95 # (We cannot simply do "mean(TE_P_means$TE)" because we want the means of the
96 # four different sound and gaze conditions.)
97 output <- aggregate(x = list(TE = TE_P_means$TE),
98                    by = list(
99                        Sound = TE_P_means$Sound,
100                        Gaze = TE_P_means$Gaze
101                    ),
102                    FUN = mean)
103
104 # Get the 95% confidence intervals of the participant means.
105 ci95s <- aggregate(x = list(CI95 = TE_P_means$TE),
106                   by = list(
107                       Sound = TE_P_means$Sound,
108                       Gaze = TE_P_means$Gaze
109                   ),
110                   FUN = CI_half_width)
111
112 # Add the confidence intervals to the output table.
113 output <- merge(output, ci95s)
114
115 # Round the values in the table.
116 output <- transform(output, TE = round(TE, digits=2))
117 output <- transform(output, CI95 = round(CI95, digits=2))
118
119 # Output the results along with the grand mean.
120 cat(dividing_line, "Tracking Error, from the participant means, in pixels,\n",
121     "and the (+/-) 95% confidence intervals\n", dividing_line, sep="")
122 print(output)
123 cat(" Grand mean: ", round( mean( output$TE ), digits=2), "\n", sep="")
124
125 rm (output, ci95s) # Cleanup.
126
127 # -----
128 # ANOVA of Tracking Error
129 # -----
130
131 # It should be noted that all of these analyses are with Day 3 data only.
132 # It would be possible to rerun the analyses to with day as a factor.
133 # But I think it is okay to just work with Day 3.
134 # It does not need to be stated that these analyses were done on the 12 participant
135 # means because it is communicated in the numerator in the degrees of freedom.
136
137 # The current summary is as follows:
138 # "The tracking error from Day 3 were analyzed.
139 # A repeated measures ANOVA with two repeated measures (peripheral visibility

```

```

140 # and sound) found a significant effect of peripheral visibility
141 # (F(1,11)=35.0, p<.001, eta-squared=0.053) but not of sound on or off,
142 # and found no significant interaction."
143
144 # The calculation is done in two ways, showing the results are the same.
145 # Note that only the ezANOVA provides the generalized-eta-squared term.
146
147 # -----
148 # Version 1 - using aov()
149
150 # "The error term is represented by subject error divided by the within-subjects error."
151 # From https://ademos.people.uic.edu/Chapter21.html (though for afex::aov_car())
152 # The within-subjects error are the two repeated measures - sound and gaze
153 # Also see https://www.uvm.edu/~dhowell/StatPages/R/RepeatedMeasuresAnovaR.html.
154 # Note that we use have two repeated measures - sound and gaze - a 2x2 design.
155
156 model_1 <- aov(
157   formula = TE ~ (Sound*Gaze) + Error( Participant / (Sound*Gaze) ),
158   data = TE_P_means
159 )
160
161 # *** NOTE THAT THIS OUTPUT HAS BEEN SUPPRESSED, JUST TO CLEAN UP THE OUTPUT. ***
162 # cat(dividing_line, "Version 1 - using aov()\n", dividing_line, sep="") # Output for
... formatting
163 # print(summary (model_1))
164
165 # -----
166 # Version 2 - using ezANOVA()
167 # This follow's DK's example from "use_cell_mean_data CSF.R"
168 # ges = generalized eta-squared
169
170 library(ez)
171
172 model_2 <- ezANOVA(
173   data = TE_P_means,
174   dv = TE,
175   wid = Participant,
176   within = .(Sound, Gaze)
177 )
178
179 cat(dividing_line, "ANOVA of Tracking Error using ezANOVA()\n", dividing_line, sep="")
180 # cat(dividing_line, "Version 2 - using ezANOVA()\n", dividing_line, sep="")
181
182 print(model_2)
183
184 rm (TE_P_means, model_1, model_2) # Cleanup
185
186 # -----
187
188 # -----
189 # Compute the RMS (Root Mean Squared) Tracking Error
190 # -----
191
192 # Add the squared error to each row of all TE data. (Adds a "TESquared" column.)
193 TEdata <- transform(TEdata, TESquared = TE ^ 2)
194
195 # write.csv(TEdata, "TEdata-tmp.txt")
196
197 # Create a table of each participant's mean squared error for each condition.
198 MSE_P_means <- aggregate(x = list(MSE = TEdata$TESquared),
199   by = list(
200     Sound = TEdata$sound,
201     Gaze = TEdata$gaze,
202     Participant = TEdata$Participant
203   ),
204   FUN = mean)
205
206 # Compute the RMS error for each participant and condition. (Adds an "RMSE" column.)
207 MSE_P_means <- transform( MSE_P_means, RMSE = sqrt(MSE_P_means$MSE))
208
209 # Average those RMSEs to get the RMSE per condition.

```

```

210 output <- aggregate(x = list(RMSE = MSE_P_means$RMSE),
211                     by = list(
212                         Sound = MSE_P_means$Sound,
213                         Gaze = MSE_P_means$Gaze
214                     ),
215                     FUN = mean)
216
217 # Get the 95% confidence intervals of the participant means.
218 ci95s <- aggregate(x = list(CI95 = MSE_P_means$RMSE),
219                   by = list(
220                       Sound = MSE_P_means$Sound,
221                       Gaze = MSE_P_means$Gaze
222                   ),
223                   FUN = CI_half_width)
224
225 # Add the confidence intervals to the output table.
226 output <- merge(output, ci95s)
227
228 # Round the values in the table.
229 output <- transform(output, RMSE = round(RMSE, digits=2))
230 output <- transform(output, CI95 = round(CI95, digits=2))
231
232 # Output the results along with the grand mean.
233 cat(dividing_line, "RMS Error, starting with participant RMSEs,\n",
234     "and the (+/-) 95% confidence intervals\n", dividing_line, sep="")
235 print(output)
236 cat(" Grand RMSE: ", round( mean( output$RMSE ), digits=2), "\n", sep="")
237
238 # write.csv(MSE_P_means, "TEdata-tmp.txt")
239
240 # Cleanup.
241 rm (output, ci95s)
242
243 # -----
244 # ANOVA of RMSTracking Error
245 # -----
246
247 # The current summary is as follows:
248 # "The RMS tracking error from Day 3 were analyzed.
249 # A repeated measures ANOVA with two repeated measures (peripheral visibility
250 # and sound) found a significant effect of peripheral visibility
251 # (F(1,11)=32.1, p<.001, eta-squared=0.066) but not of sound on or off,
252 # and found no significant interaction."
253
254 # The calculation is done using ezANOVA (and not aov() also as was done above.)
255
256 library(ez)
257
258 model <- ezANOVA(
259     data = MSE_P_means,
260     dv = RMSE,
261     wid = Participant,
262     within = .(Sound, Gaze)
263 )
264
265 cat(dividing_line, "ANOVA of RMS Tracking Error using ezANOVA()\n", dividing_line, sep="")
266 print(model)
267
268 # Cleanup.
269 rm (MSE_P_means, model)
270
271 # -----
272
273 # The ezANOVA outputs (in August, 2019):
274
275 # Tracking Error
276
277 # Effect DFn DFd          F          p p<.05          ges
278 # 2      Sound    1  11  2.516716 0.1409520751      0.0052236707
279 # 3      Gaze     1  11 35.024741 0.0001004386      * 0.0530178146
280 # 4 Sound:Gaze    1  11  0.617168 0.4486927863      0.0006504402

```

```

281
282 # RMS Tracking Error
283
284 # Effect DFn DFd          F          p p<.05          ges
285 # 2      Sound    1   11  1.10238624 0.3162633380      0.0035086413
286 # 3      Gaze     1   11 32.12553900 0.0001449067      * 0.0655802585
287 # 4 Sound:Gaze    1   11  0.08935375 0.7705719527      0.0001600389
288
289 # -----
290 # Recompute TE and RMSE TE only based on gaze condition, not by sound.
291 # -----
292
293 cat(dividing_line, "Recompute the TE and RMSE separating by peripheral visible or not, but
... not by\n",
294      "sound on or off, because the ANOVAs above showed that sound has no effect.\n", sep="")
295
296 # Because sound has no effect on tracking performance, we again go through the
297 # steps above but this time ignoring sound as a factor.
298
299 # -----
300 # Compute the Tracking Error (TE).
301 # -----
302
303 # Generate the tracking errors (a) for Day 3 only (b) without removing any "outliers"
304 # and (c) after first averaging by participant.
305
306 # First create a table of the participant means.
307 TE_P_means <- aggregate(x = list(TE = TEdata$TE), # "list(TE=...)" renames Column "x"
308                          by = list(
309                              Gaze = TEdata$gaze,
310                              Participant = TEdata$Participant
311                          ),
312                          FUN = mean)
313
314 # Now get the means of the participant means.
315 output <- aggregate(x = list(TE = TE_P_means$TE),
316                     by = list(Gaze = TE_P_means$Gaze),
317                     FUN = mean)
318
319 # Round the TE. (ajh)
320 output <- transform(output, TE = round(TE, digits=1))
321 # Why not simply the following??? (ajh)
322 # output$TE <- round(output$TE, digits=1)
323
324 # Output the results along with the grand mean.
325 cat(dividing_line, "Tracking Error, from the participant means, in pixels\n", dividing_line,
... sep="")
326 print(output)
327 cat("Grand Mean: ", round( mean( output$TE ), digits=1), "\n", sep="")
328
329 rm (output, TE_P_means) # Cleanup.
330
331 # -----
332 # Compute the RMS (Root Mean Squared) Tracking Error
333 # -----
334
335 # Create a table of each participant's mean squared error for each condition.
336 MSE_P_means <- aggregate(x = list(MSE = TEdata$TESquared),
337                           by = list(
338                               Gaze = TEdata$gaze,
339                               Participant = TEdata$Participant
340                           ),
341                           FUN = mean)
342
343 # Compute the RMS error for each participant and condition. (Adds an "RMSE" column.)
344 MSE_P_means <- transform( MSE_P_means, RMSE = sqrt(MSE_P_means$MSE))
345
346 # Now average those RMSEs to get the RMSE per condition.
347 output <- aggregate(x = list(RMSE = MSE_P_means$RMSE),
348                     by = list(
349                         Gaze = MSE_P_means$Gaze

```

```

350         ),
351         FUN = mean)
352
353 # Round the RMSE.
354 output <- transform(output, RMSE = round(RMSE, digits=1))
355
356 # Output the results along with the grand mean.
357 cat(dividing_line, "RMS Error, starting with participant RMSEs, in pixels\n", dividing_line,
... sep="")
358 print(output)
359 cat("Grand Mean: ", round( mean( output$RMSE ), digits=1), "\n", sep="")
360
361 # write.csv(MSE_P_means, "TEdata-tmp.txt")
362
363 # Cleanup.
364 rm (output, MSE_P_means, TEdata)
365

```