# ANOVA_tracking error.r - R Code for data analysis for CHI 2010 paper by Hornof et al.
# R scripts stats for the EPIC NRL dual task model by Hornof.
# R scripts originally written by Yunfeng Zhang - 1/5/2010
# Backed up in: Dual Task/Analysis/zArchived Analyses/
# 3_09 Fall R (no transformation and use VizFix data V1)/scripts/ANOVA_tracking error.r
# Updated by Hornof (ajh) in 2019.

# This script creates tables including:
# TEdata: All tracking errors, recorded once ever 83ms.
# MeanTE: means by Participant and condition.
# It no longer uses these tables:
# allTETdataTrim2sd: allTETdata, but without TEs over 2 standard deviations from the mean.
# allTETdataTrim200: allTETdata, but with no TEs over 200 pixels. Used for CHI 2010 graph.
# I could create a new .Rdata file such as by doing the following:
# save (TEdata, MeanTE, file="TE.Rdata")
# It takes less disk space to just recreate tables above for every use. (ajh)

# Load packages and data.
# ------------------------------------------------------------------

# Set the working directory (ajh)

# YZ 1/5/2010
# import blips data. Taken from importData.R.

# Read the table and remove bad participants.
blips <- removeTwoPsDay3(read.table("./data/classification time.txt", header = T))
blips$order <- ordered(as.character(blips$order), levels = c("1", "2", "3", "4"))
blips$day <- factor(as.character(blips$day))
blips$error.type <- factor(as.character(blips$error.type))

# ajh changed 2019
# levels(blips$gaze) <- c("g.off", "g.on")
# levels(blips$sound) <- c("s.off", "s.on")
levels(blips$gaze) <- c("no", "yes") # "Yes" means gaze contingency was on.
levels(blips$sound) <- c("no", "yes") # "Yes" means sounds played for blip events.

# import tracking error data.
TEdata <- removeTwoPsDay3(read.table("./data/all tracking error.txt", header = T))
TEdata <- merge(TEdata, unique(blips[c("id", "day", "order", "sound", "gaze")]), all.x=T)

# aggregate TE
MeanTE <- as.data.frame(ftable(
  with(TEdata, tapply(TE, list(id, day, sound, gaze), mean))))
names(MeanTE) <- c("id", "day", "sound", "gaze", "TE")

# Cleanup. (Remove temporary objects to avoid confusion.)
rm (blips, MeanTE)

# Compute the Tracking Error (TE)
# Computes the Tracking Error (TE)
# ------------------------------------------------------------------------------
# Load the function removeTwoPsDay3( ) to remove P14 & P15.
source("./scripts/removeTwoPsDay3.r")

# load dplyr for rename()
library(dplyr)
source("./scripts/graphFunctions.r") # For CI_half_width( )
# Load the function removeTwoPsDay3( ) to remove P14 & P15.
source("./scripts/importDataUtilities.r")
# ------------------------------------------------------------------

# Generate the tracking errors (a) for Day 3 only (b) without removing any "outliers" and (c) after first averaging by participant.
# Limit the analyses to Day 3.
TEdata <- filter(TEdata, day==3)
# Rename the "id" column to "Participant"
TEdata <- rename(TEdata, Participant=id)

# TEMPORARILY REMOVE P04 & P14 & P15
TEdata <- filter(TEdata, Participant != "P04")
TEdata <- filter(TEdata, Participant != "P14")
TEdata <- filter(TEdata, Participant != "P15")

# First create a table of the participant means.
TE_P_means <- aggregate(x = list(TE = TEdata$TE),
by = list(Sound = TEdata$sound, Gaze = TEdata$gaze, Participant = TEdata$Participant ),
  FUN = mean)

# Get the means of the participant means.
# (We cannot simply do "mean(TE_P_means$TE)" because we want the means of the
# four different sound and gaze conditions.)
output <- aggregate(x = list(TE = TE_P_means$TE),
by = list(Sound = TE_P_means$Sound, Gaze = TE_P_means$Gaze ),
  FUN = mean)

# Get the 95% confidence intervals of the participant means.
ci95s <- aggregate(x = list(CI95 = TE_P_means$TE),
by = list(Sound = TE_P_means$Sound, Gaze = TE_P_means$Gaze ),
  FUN = CI_half_width)

# Add the confidence intervals to the output table.
output <- merge(output, ci95s)

# Round the values in the table.
output <- transform(output, TE = round(TE, digits=2))
output <- transform(output, CI95 = round(CI95, digits=2))

# Output the results along with the grand mean.
cat(dividing_line, "Tracking Error, from the participant means, in pixels,\n", "and the (+/-) 95% confidence intervals\n", dividing_line, sep="")
print(output)
cat(" Grand mean: ", round( mean( output$TE ), digits=2), "\n", sep="")
rm (output, ci95s) # Cleanup.

# ANOVA of Tracking Error
# It should be noted that all of these analyses are with Day 3 data only.
# It would be possible to rerun the analyses to with day as a factor.
# But I think it is okay to just work with Day 3.
# It does not need to be stated that these analyses were done on the 12 participant
# means because it is communicated in the numerator in the degrees of freedom.
# The current summary is as follows:
# The tracking error from Day 3 were analyzed.
# A repeated measures ANOVA with two repeated measures (peripheral visibility
and sound) found a significant effect of peripheral visibility
(F(1,11)=35.0, p<.001, eta-squared=0.053) but not of sound on or off,
and found no significant interaction.

The calculation is done in two ways, showing the results are the same.
Note that only the ezANOVA provides the generalized-eta-squared term.

# Version 1 - using aov()

# "The error term is represented by subject error divided by the within-subjects error."
# From https://ademos.people.uic.edu/Chapter21.html (though for afex::aov_car())
# The within-subjects error are the two repeated measures - sound and gaze
# Also see https://www.uvm.edu/~dhowell/StatPages/R/RepeatedMeasuresAnovaR.html.
# Note that we use have two repeated measures - sound and gaze - a 2x2 design.

model_1 <- aov(
  formula = TE ~ (Sound*Gaze) + Error( Participant / (Sound*Gaze) ),
  data = TE_P_means
)

# *** NOTE THAT THIS OUTPUT HAS BEEN SUPPRESSED, JUST TO CLEAN UP THE OUTPUT. ***
# cat(dividing_line, "Version 1 - using aov()\n", dividing_line, sep="") # Output for formatting
# print(summary (model_1))

# Version 2 - using ezANOVA()
# This follow's DK's example from "use_cell_mean_data CSF.R"
# ges = generalized eta-squared

library(ez)

model_2 <- ezANOVA(
  data = TE_P_means,
  dv = TE,
  wid = Participant,
  within = .(Sound, Gaze)
)

cat(dividing_line, "ANOVA of Tracking Error using ezANOVA()\n", dividing_line, sep="")
# cat(dividing_line, "Version 2 - using ezANOVA()\n", dividing_line, sep="")

print(model_2)

rm (TE_P_means, model_1, model_2) # Cleanup

# Compute the RMS (Root Mean Squared) Tracking Error

# Add the squared error to each row of all TE data. (Adds a "TEsquared" column.)
TEdata <- transform(TEdata, TEsquared = TE ^ 2)

# Create a table of each participant’s mean squared error for each condition.
MSE_P_means <- aggregate(x = list(MSE = TEdata$TEsquared),
                          by = list(
                            Sound = TEdata$sound,
                            Gaze = TEdata$gaze,
                            Participant = TEdata$Participant
                          ),
                          FUN = mean)

# Compute the RMS error for each participant and condition. (Adds an "RMSE" column.)
MSE_P_means <- transform( MSE_P_means, RMSE = sqrt(MSE_P_means$MSE))

# Average those RMSEs to get the RMSE per condition.
output <- aggregate(x = list(RMSE = MSE_P_means$RMSE),
                     by = list(
                       Sound = MSE_P_means$Sound,
                       Gaze = MSE_P_means$Gaze
                     ),
                     FUN = mean)

# Get the 95% confidence intervals of the participant means.
ci95s <- aggregate(x = list(CI95 = MSE_P_means$RMSE),
                     by = list(
                       Sound = MSE_P_means$Sound,
                       Gaze = MSE_P_means$Gaze
                     ),
                     FUN = CI_half_width)

# Add the confidence intervals to the output table.
output <- merge(output, ci95s)

# Round the values in the table.
output <- transform(output, RMSE = round(RMSE, digits=2))
output <- transform(output, CI95 = round(CI95, digits=2))

# Output the results along with the grand mean.
cat(dividing_line, "RMS Error, starting with participant RMSEs,\nand the (+/-) 95% confidence intervals\n", dividing_line, sep="")
print(output)
cat(" Grand RMSE: ", round(mean(output$RMSE), digits=2), "\n", sep="")

# write.csv(MSE_P_means, "TEdata-tmp.txt")

# Cleanup.
rm(output, ci95s)

# ANOVA of RMSTracking Error
# The current summary is as follows:
# "The RMS tracking error from Day 3 were analyzed.
# A repeated measures ANOVA with two repeated measures (peripheral visibility
# and sound) found a significant effect of peripheral visibility
# (F(1,11)=32.1, p<.001, eta-squared=0.066) but not of sound on or off,
# and found no significant interaction."

# The calculation is done using ezANOVA (and not aov() also as was done above.)
library(ez)
model <- ezANOVA(
  data = MSE_P_means,
  dv = RMSE,
  wid = Participant,
  within = .(Sound, Gaze)
)

cat(dividing_line, "ANOVA of RMS Tracking Error using ezANOVA()\n", dividing_line, sep="")
print(model)

# Cleanup.
rm(MSE_P_means, model)

# The ezANOVA outputs (in August, 2019):
# Tracking Error
# Effect DFn DFd      F      p   p<.05    ges
# 2 Sound   1  11  2.516716 0.140952075 0.0052236707
# 3 Gaze    1  11 35.024741 0.0001004386 * 0.0530178146
# 4 Sound:Gaze 1  11 0.617168 0.4486927863 0.0006504402
# RMS Tracking Error

<table>
<thead>
<tr>
<th>Effect</th>
<th>DFn</th>
<th>DFd</th>
<th>F</th>
<th>p</th>
<th>p&lt;.05</th>
<th>ges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sound</td>
<td>1</td>
<td>11</td>
<td>1.10238624</td>
<td>0.3162633380</td>
<td>0.0035086413</td>
<td></td>
</tr>
<tr>
<td>Gaze</td>
<td>1</td>
<td>11</td>
<td>32.12553900</td>
<td>0.0001449067</td>
<td>* 0.0655802585</td>
<td></td>
</tr>
<tr>
<td>Sound:Gaze</td>
<td>1</td>
<td>11</td>
<td>0.08935375</td>
<td>0.7705719527</td>
<td>0.0001600389</td>
<td></td>
</tr>
</tbody>
</table>

---

Recompute TE and RMSE TE only based on gaze condition, not by sound.

---

cat(dividing_line, "Recompute the TE and RMSE separating by peripheral visible or not, but not by\n"   "sound on or off, because the ANOVAs above showed that sound has no effect.\n", sep="")

Because sound has no effect on tracking performance, we again go through the steps above but this time ignoring sound as a factor.

---

Compute the Tracking Error (TE).

---

# First create a table of the participant means.

TE_P_means <- aggregate(x = list(TE = TEdata$TE), # "list(TE=...)") renames Column "x"   by = list(   Gaze = TEdata$gaze,   Participant = TEdata$Participant ),   FUN = mean)

# Now get the means of the participant means.

output <- aggregate(x = list(TE = TE_P_means$TE),   by = list(Gaze = TE_P_means$Gaze),   FUN = mean)

# Round the TE. (ajh)
output <- transform(output, TE = round(TE, digits=1))

# Why not simply the following?? (ajh)
# output$TE <- round(output$TE, digits=1)

# Output the results along with the grand mean.

cat(dividing_line, "Tracking Error, from the participant means, in pixels\n", dividing_line,         print(output)            cat("Grand Mean: ", round( mean( output$TE ), digits=1 ), "\n", sep=""))

rm (output, TE_P_means) # Cleanup.

---

Compute the RMS (Root Mean Squared) Tracking Error

---

# Create a table of each participant's mean squared error for each condition.

MSE_P_means <- aggregate(x = list(MSE = TEdata$TEsquared),   by = list(   Gaze = TEdata$gaze,   Participant = TEdata$Participant ),   FUN = mean)

# Compute the RMS error for each participant and condition. (Adds an "RMSE" column.)
MSE_P_means <- transform( MSE_P_means, RMSE = sqrt(MSE_P_means$MSE))

# Now average those RMSEs to get the RMSE per condition.
output <- aggregate(x = list(RMSE = MSE_P_means$RMSE),   by = list(   Gaze = MSE_P_means$Gaze)


# Round the RMSE.
output <- transform(output, RMSE = round(RMSE, digits=1))

# Output the results along with the grand mean.
cat(dividing_line, "RMS Error, starting with participant RMSEs, in pixels\n", dividing_line, sep="")
print(output)
cat("Grand Mean: ", round( mean( output$RMSE ), digits=1), "\n", sep="")

# write.csv(MSE_P_means, "TEdata-tmp.txt")

# Cleanup.
rm (output, MSE_P_means, TEdata)