

```

setwd('/home/allgoodguys/Documents/Studying/Lund_PhD/z_helping-out/
lima')

df = read.csv ('data_preprocessed_formatForMatch.csv')
df = df[df$condition=='deliberated',]
parts = unique(df$subject)

folder = '/home/allgoodguys/Documents/Studying/Lund_PhD/z_helping-
out/lima/deliberated_csv'
files = dir(folder)

# matching participant numbers
for (f in files){
  p <- read.csv(paste0(folder,'/',f)) # open file
  for (i in 1:length(parts)){
    temp = df[df$subject==parts[i],] # data for one participant at a
time
    propMatch = sum(p$ReactionTime %in% temp$time) / nrow(temp) #
proportion of matching times
    if (propMatch>.5){
      print (paste(f, parts[i])) # print which participant has >50%
of matches (should be just one). Rename .csv file starting with this
number for future matching
    }
  }
}

### PRETEST ***
df = read.csv('data/data_pretest1.csv')
df = df[df$Included == 'Y', ]

for (i in 1:nrow(df)) {
  s = as.character(df$Stimulus[i])
  s1 = strsplit(s, ' ')[[1]][3]
  if (substr(s1, 1, 4) == 'EXP_') {
    start = 5
  } else {
    start = 1
  }
  df$soundName_short[i] = substr(s1, start, nchar(s1)-5)
}

colnames(df)
a = df[, c(25, 1:2, 5:24)]

library(reshape2)
b = melt(a)
unique(b$Feature)
idx_val = which(b$Feature == 'Valence')
idx_ar = which(b$Feature == 'Arousal')
idx_int = which(b$Feature == 'Intensity')

```

```

val = data.frame(
  subject = b$variable[idx_val],
  soundName_short = b$soundName_short[idx_val],
  emotion = b$Emotion[idx_val],
  pretestVal = b$value[idx_val]
)
ar = data.frame(
  subject = b$variable[idx_ar],
  soundName_short = b$soundName_short[idx_ar],
  emotion = b$Emotion[idx_ar],
  pretestAr = b$value[idx_ar]
)
int = data.frame(
  subject = b$variable[idx_int],
  soundName_short = b$soundName_short[idx_int],
  emotion = b$Emotion[idx_int],
  pretestInt = b$value[idx_int]
)
which(paste(val$subject, val$soundName_short) != paste(ar$subject,
ar$soundName_short)) # same order
which(paste(val$subject, val$soundName_short) != paste(int$subject,
int$soundName_short)) # same order

out = val
out$pretestAr = ar[, 4]
out$pretestInt = int[, 4]
out$label = NA
out$correct = NA

## add accuracy in forced choice task by 20 DIFFERENT participants
df2 = read.csv('data/data_pretest2.csv')
df2 = df2[df2$Included == 'Y', ]

for (i in 1:nrow(df2)) {
  s = as.character(df2$Stimulus[i])
  s1 = strsplit(s, ' ')[[1]][3]
  if (substr(s1, 1, 4) == 'EXP_') {
    start = 5
  } else {
    start = 1
  }
  df2$soundName_short[i] = substr(s1, start, nchar(s1)-5)
}
colnames(df2)
df2 = df2[, c(25, 1, 4, 5:24)]

c = melt(df2, id.vars = c('soundName_short', 'Emotion',
'Correct.label'))
c$correct = c$Correct.label == c$value

out2 = data.frame(
  subject = c$variable,
  soundName_short = c$soundName_short,
  emotion = c$Emotion,

```

```

    pretestVal = NA,
    pretestAr = NA,
    pretestInt = NA,
    label = c$value,
    correct = c$correct
  )

out3 = rbind(out, out2)
# summary(out3)

## add duration
df3 = read.csv ('data/data_preprocessed.csv')[,-1]
any(is.na(match(out3$soundName_short, df3$soundName_short))) #
thank goodness

out3$duration = df3$duration[match(out3$soundName_short,
df3$soundName_short)]

# write.csv(out3, 'data/data_pretest_preprocessed.csv')

## aggregate per sound
ag = aggregate(duration ~ emotion + soundName_short, out3, mean)
ag$pretestVal = aggregate(pretestVal ~ soundName_short, out3, mean)
[, 2]
ag$pretestAr = aggregate(pretestAr ~ soundName_short, out3, mean) [,
2]
ag$pretestInt = aggregate(pretestInt ~ soundName_short, out3, mean)
[, 2]
ag$correct = aggregate(correct ~ soundName_short, out3, mean) [, 2]
ag = ag[order(ag$emotion, ag$soundName_short), ]
# write.csv(ag, 'data/data_pretest_aggreg.csv')

```