

acoustic_analysis

```
cur_exp = "exp2"
features = c("duration", "meanIntensity", "meanpit")
# info = c('participant', 'verb', 'condition', 'word', 'word_num')
info = c('participant', 'item_id', 'location_condition', 'word', 'word_num')
bRemove_outliers = 0
```

This the analysis for exp2. The parameters of all exps can be seen at https://github.com/Xinzhu-Fang/prosody_study_exp/blob/master/tAll_exps.csv.

The trial-by-trial design of this exp can be seen at https://github.com/Xinzhu-Fang/prosody_study_exp/blob/master/exp2/tAll_trials.csv

```
tAll_trials = read.csv(file.path '..', cur_exp, 'tAll_trials.csv'))

df0 = read.csv(paste0('measure_', cur_exp, '.csv'), header = T)
df0$location_condition = NA
df0$item_id = NA

for (iR in 1:nrow(df0)){
  df0$location_condition[iR] = as.character(tAll_trials[tAll_trials$trial_id == df0$trialId[iR], 'location_condition'])
  df0$item_id[iR] = as.character(tAll_trials[tAll_trials$trial_id == df0$trialId[iR], 'filler_or_item_id'])
}

df1 = df0[startsWith(df0$item_id, "item"),]

# df0 = read.csv("measure_nonrhyming_84total_60No_24Yes_20181210.csv", header = T)
# df0 = transform(df0, trialId=as.numeric(trialId))
# sort(df0$trialId, decreasing = FALSE)
# colnamesC(df1)

df2 = df1[df1$word != 'sp',]
# code for word_num
df2 <- df2 %>%
  dplyr::group_by(participant, trialId) %>%
  # dplyr::group_by(participant, question, trialId) %>%
  dplyr::mutate(word_num=1:n()) %>%
  dplyr::select(c(info, features))

## Adding missing grouping variables: `trialId`
```

29 workers and 779 trials are included in this analysis.

```
# write.csv(df2, 'newdf.csv')
# code for getting Nth instance of question
# nthdf <- df1 %>%
#   group_by(participant, Verb, question, condition, word_num) %>%
#   mutate(Appearance=1:n())
# write.csv(nthdf, 'nthdf.csv')
```

```

# subsetting it to relevant Nth appearance
# workingdf <- nthdf %>%
#   filter (Appearance == 2)
#
# write.csv(workingdf, 'workingdf2.csv')

normalize_data = function(df, remove_outliers){
  for(col_name in features){
    if(!is.numeric(df[[col_name]])){
      df[[col_name]] = as.numeric(df[[col_name]])
    }
    df[[col_name]] = scale(df[[col_name]])
    # there is surge of na after the first colling of the above line. tested by print(sum(is.na(df_Agent)))
    # print(sum(is.na(df_Agent)))
  }
  for(col_name in features){
    if(remove_outliers){
      df = df[df[[col_name]]>-2 & df[[col_name]]<2,]
      # print(sum(is.na(df_Agent)))
    }
  }
  return(df)
}

# process_data = function(file_name){
process_data = function(df){
  # df <- read.csv(file_name, header = TRUE, fileEncoding="UTF-8", na.strings=c("", "NA", "--undefined--"))
  # df <- na.omit(df)

  #df = df[df$wordlabel != 'sp']
  # df$verb = as.factor(df$verb)

  # df_Agent = df[(df$location_condition=='Agent' | df$location_condition=='Verb') & df$word_num=='3',]
  #
  # df_Verb = df[(df$location_condition=='Verb' | df$location_condition=='Patient') & df$word_num=='5',]
  #
  # df_Patient = df[(df$location_condition=='Patient' | df$location_condition=='Agent') & df$word_num=='1',]

  df_Agent = df[(df$location_condition=='Agent' | df$location_condition=='Control') & df$word_num=='2',]
  # df_Agent inheri row hum from df

  df_Verb = df[(df$location_condition=='Verb' | df$location_condition=='Control') & df$word_num=='4',]

  df_Patient = df[(df$location_condition=='Patient' | df$location_condition=='Control') & df$word_num=='1',]

  # print(sum(is.na(df_Agent)))

  # relevant_columns = c('participant', 'verb', 'condition', 'duration', 'meanIntensity', 'meanpit')
  # df_Agent = df_Agent[relevant_columns]

```

```

# df_Verb = df_Verb[relevant_columns]
# df_Patient = df_Patient[relevant_columns]
print(sum(is.na(df[df$word != 'sp',])))
# df1[(df1$meanpit == '--undefined--') && (df1$word != 'sp'),]
# it seems that the only undefined is meanpitch for sp

# print(df_Verb)

df_Verb = normalize_data(df_Verb, bRemove_outliers)
df_Agent = normalize_data(df_Agent, bRemove_outliers)
df_Patient = normalize_data(df_Patient, bRemove_outliers)
# print(sum(is.na(df_Agent)))

# return(list(df_Agent_duration, df_Agent_meanIntensity, df_Agent_meanpit, df_Patient_duration, df_Pa
return(list(df_Verb, df_Agent, df_Patient))
}

#
# file_name = 'newdf.csv'

# c(df_Agent_duration, df_Agent_meanIntensity, df_Agent_meanpit, df_Patient_duration, df_Patient_meanIn
# c(df_Verb, df_Agent, df_Patient) %<-% process_data(file_name)
c(df_Verb, df_Agent, df_Patient) %<-% process_data(df2)

## [1] 0

combine_datasets = function(Agent,Verb,Patient){
  Agent$condition = mapvalues(Agent$location_condition,c('Agent'),c('contrast'))
  Verb$condition = mapvalues(Verb$location_condition,c('Verb'),c('contrast'))
  Patient$condition = mapvalues(Patient$location_condition,c('Patient'),c('contrast'))

  Agent$Location = 'Agent'
  Verb$Location = 'Verb'
  Patient$Location = "Patient"

  return(rbind(Agent,Verb,Patient))
}

summarize_data = function(d, feature){
  # http://www.cookbook-r.com/Graphs/Plotting\_means\_and\_error\_bars\_\(ggplot2\)/
  return(summarySE(d,measurevar=feature ,groupvars=c('Location','condition'))
}

plot_data = function(d,feature, title){
  print(ggplot(d, aes(x=Location, y=get(feature), fill=condition)) +
    geom_bar(position=position_dodge(), stat="identity") +
    geom_errorbar(aes(ymin=get(feature)-ci, ymax=get(feature)+ci),
      width=.2,
      position=position_dodge(.9))+
    xlab("Location") +
    ylab(paste0("normalized ", feature)) +

```

```

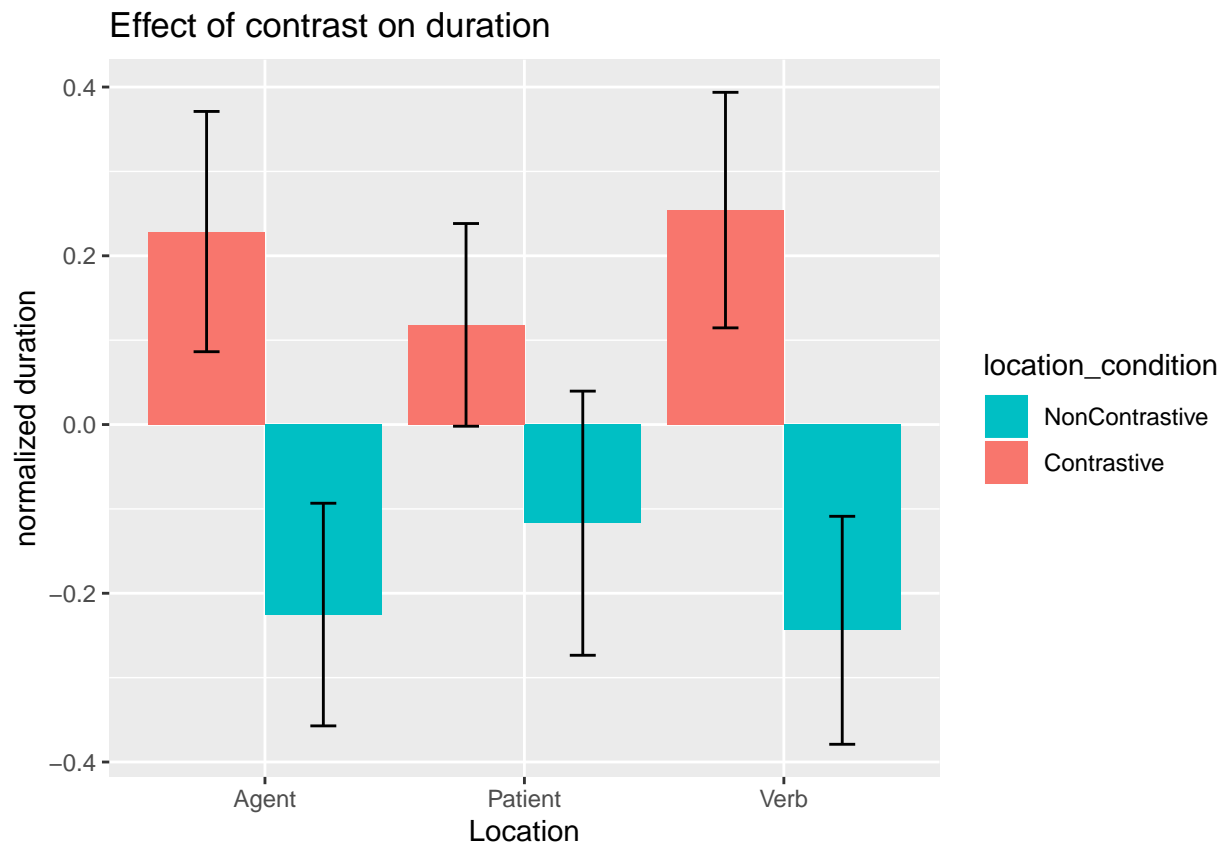
    scale_fill_hue(name="location_condition",
                   breaks=c("Control", "contrast"),
                   labels=c("NonContrastive", "Contrastive")) +
    ggtitle(title))
}
for (iF in features){
  print(iF)

  combined_dataset = combine_datasets(df_Agent, df_Verb, df_Patient)
  summarized_dataset= summarize_data(combined_dataset, iF)

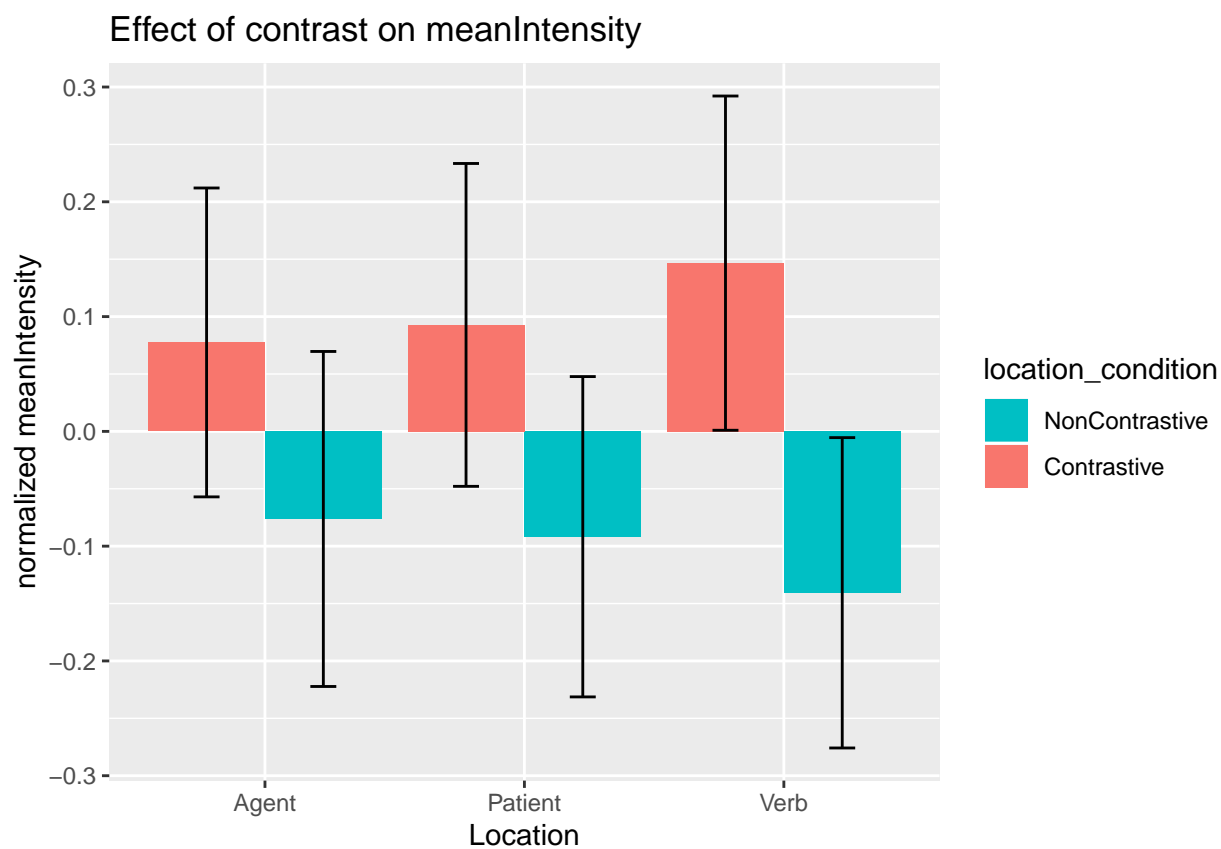
  plot_data(summarized_dataset,iF, title= paste0('Effect of contrast on ', iF))
}

```

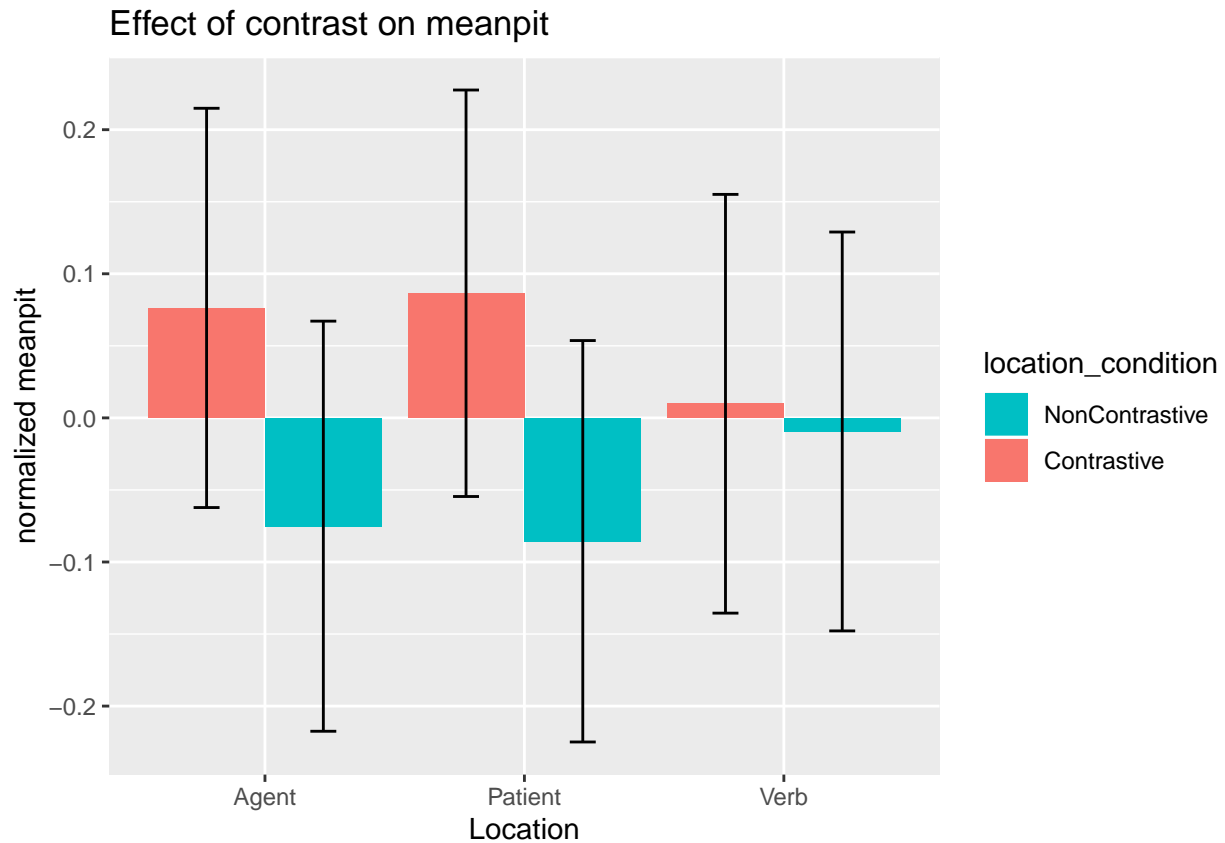
```
## [1] "duration"
```



```
## [1] "meanIntensity"
```



```
## [1] "meanpit"
```



```
run_regression = function(location, observation){
  cat(" \n###", observation, "of", location, " \n")
  r = lmer(get(observation) ~ location_condition + (1 + location_condition|participant) + (1 + location.
  # r = lmer(get(observation) ~ location_condition + (1 + location_condition | item_id), data=get(past
  print(summary(r))
  summary(r)
  cat(" \n")
}

for (iF in features){
  run_regression("Agent", iF)

  run_regression("Patient", iF)

  run_regression("Verb", iF)
}

##
## ### duration of Agent
## boundary (singular) fit: see ?isSingular
```

```

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 903.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.3525 -0.5951 -0.0921  0.4453  4.2244
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          1.08979  1.0439
##              location_conditionControl 0.13348  0.3653  -0.31
##   item_id     (Intercept)          0.06493  0.2548
##              location_conditionControl 0.03651  0.1911  -1.00
## Residual                        0.42043  0.6484
## Number of obs: 393, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      0.3377    0.2381   1.418
## location_conditionControl -0.4671    0.1357  -3.442
##
## Correlation of Fixed Effects:
##              (Intr)
## lctn_cndtnC -0.583
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
## ### duration of Patient
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 886.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.6146 -0.4160 -0.0692  0.3653  5.8347
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          1.428177  1.19506
##              location_conditionPatient 0.425094  0.65199  -0.81
##   item_id     (Intercept)          0.061866  0.24873
##              location_conditionPatient 0.004456  0.06676   1.00
## Residual                        0.386919  0.62203

```

```

## Number of obs: 394, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      -0.05519   0.25896  -0.213
## location_conditionPatient  0.20046   0.14175   1.414
##
## Correlation of Fixed Effects:
##              (Intr)
## lctn_cndtnP -0.539
##
##
## ### duration of Verb
##
## boundary (singular) fit: see ?isSingular
##
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 887.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.4628 -0.5260 -0.1106  0.3835  6.2027
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          1.64942  1.2843
##               location_conditionVerb 0.04913  0.2217  0.36
##   item_id     (Intercept)          0.06723  0.2593
##               location_conditionVerb 0.04568  0.2137  1.00
##   Residual                        0.40560  0.6369
## Number of obs: 388, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      -0.1168   0.2759  -0.423
## location_conditionVerb  0.5337   0.1323   4.034
##
## Correlation of Fixed Effects:
##              (Intr)
## lctn_cndtnV 0.423
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### meanIntensity of Agent
##
## boundary (singular) fit: see ?isSingular
##
## Linear mixed model fit by REML ['lmerMod']
## Formula:

```



```

## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 658.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -5.1884 -0.4670  0.0994  0.5150  2.6906
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          0.735290 0.8575
##             location_conditionControl 0.072319 0.2689  0.32
##   item_id     (Intercept)          0.106913 0.3270
##             location_conditionControl 0.003857 0.0621 -1.00
##   Residual                        0.215728 0.4645
## Number of obs: 393, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      -0.06586   0.23129  -0.285
## location_conditionControl -0.12112   0.07631  -1.587
##
## Correlation of Fixed Effects:
##              (Intr)
## lctn_cndtnC -0.214
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
## ### meanIntensity of Patient

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.00495708
## (tol = 0.002, component 1)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 784.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3309 -0.5457 -0.0692  0.6198  3.2553
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          0.833603 0.91302
##             location_conditionPatient 0.001246 0.03530 -1.00
##   item_id     (Intercept)          0.006643 0.08151

```

```

##           location_conditionPatient 0.002803 0.05294 -1.00
## Residual                        0.327253 0.57206
## Number of obs: 394, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)      -0.07399   0.17980  -0.412
## location_conditionPatient  0.22068   0.06392   3.453
##
## Correlation of Fixed Effects:
##           (Intr)
## lctn_cndtnP -0.333
## convergence code: 0
## Model failed to converge with max|grad| = 0.00495708 (tol = 0.002, component 1)
##
##
## ### meanIntensity of Verb
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.0103674
## (tol = 0.002, component 1)
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 680
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.6962 -0.5781  0.0468  0.6132  2.7703
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   participant (Intercept)          6.885e-01 0.829730
##               location_conditionVerb 9.070e-06 0.003012 -1.00
##   item_id     (Intercept)          8.669e-02 0.294428
##               location_conditionVerb 1.069e-02 0.103394 0.99
## Residual                        2.489e-01 0.498887
## Number of obs: 388, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)      -0.13482   0.21653  -0.623
## location_conditionVerb  0.26820   0.07264   3.692
##
## Correlation of Fixed Effects:
##           (Intr)
## lctn_cndtnV 0.397
## convergence code: 0
## Model failed to converge with max|grad| = 0.0103674 (tol = 0.002, component 1)
##

```

```

##
##
## ### meanpit of Agent

## boundary (singular) fit: see ?isSingular

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 770.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.7569 -0.2530  0.0166  0.2832  4.8033
##
## Random effects:
##   Groups             Name                Variance Std.Dev.  Corr
##   participant (Intercept)                7.541e-01 8.684e-01
##               location_conditionControl  2.543e-03 5.043e-02 -0.39
##   item_id      (Intercept)                1.252e-09 3.539e-05
##               location_conditionControl  7.561e-11 8.695e-06 1.00
## Residual                        3.198e-01 5.655e-01
## Number of obs: 393, groups:  participant, 29; item_id, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      0.02195   0.16736   0.131
## location_conditionControl -0.09930   0.05833  -1.702
##
## Correlation of Fixed Effects:
##              (Intr)
## lctn_cndtnC -0.237
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### meanpit of Patient

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
##   participant) + (1 + location_condition | item_id)
##   Data: get(paste0("df_", location))
##
## REML criterion at convergence: 919.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.1094 -0.2813  0.0395  0.4544  3.6230
##
## Random effects:
##   Groups             Name                Variance Std.Dev.  Corr

```

```

## participant (Intercept)                0.663418 0.81450
##          location_conditionPatient 0.092499 0.30414 -0.66
## item_id (Intercept)                0.001072 0.03273
##          location_conditionPatient 0.007932 0.08906 -1.00
## Residual                0.475585 0.68963
## Number of obs: 394, groups: participant, 29; item_id, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      -0.1127    0.1610  -0.700
## location_conditionPatient  0.1483    0.1009   1.471
##
## Correlation of Fixed Effects:
##          (Intr)
## lctn_cndtnP -0.547
##
## ### meanpit of Verb
##
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.00299067
## (tol = 0.002, component 1)
##
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## get(observation) ~ location_condition + (1 + location_condition |
## participant) + (1 + location_condition | item_id)
## Data: get(paste0("df_", location))
##
## REML criterion at convergence: 894.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.9524 -0.2666 -0.0045  0.3080  3.8443
##
## Random effects:
## Groups      Name                Variance Std.Dev.  Corr
## participant (Intercept)          5.263e-01 0.7254446
##          location_conditionVerb 3.517e-01 0.5930283 -0.21
## item_id (Intercept)              1.172e-07 0.0003423
##          location_conditionVerb 1.285e-07 0.0003585 0.71
## Residual                        4.324e-01 0.6575756
## Number of obs: 388, groups: participant, 29; item_id, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      -0.07068    0.14375  -0.492
## location_conditionVerb  0.01873    0.13190   0.142
##
## Correlation of Fixed Effects:
##          (Intr)
## lctn_cndtnV -0.287
## convergence code: 0
## Model failed to converge with max|grad| = 0.00299067 (tol = 0.002, component 1)
##

```

```
##
```

```
# r = lmer(get(observation) ~ condition + (1 | participant) + (1 | verb), data=df)
```