acoustic_analysis

```
cur_exp = "exp3"
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 exp3. 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/exp3/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$trialId[iR],'locati
  df0$item_id[iR] =  as.character(tAll_trials[tAll_trials$trial_id == df0$trialId[iR],'filler_or_item_i
  df0$present_num[iR] = as.numeric(rownames(tAll_trials[tAll_trials$trial_id == df0$trialId[iR],]))
}

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:dplyr::n()) %>%
  dplyr::select(c(info, features))
```

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

31 workers and 820 trials are included in this analysis.

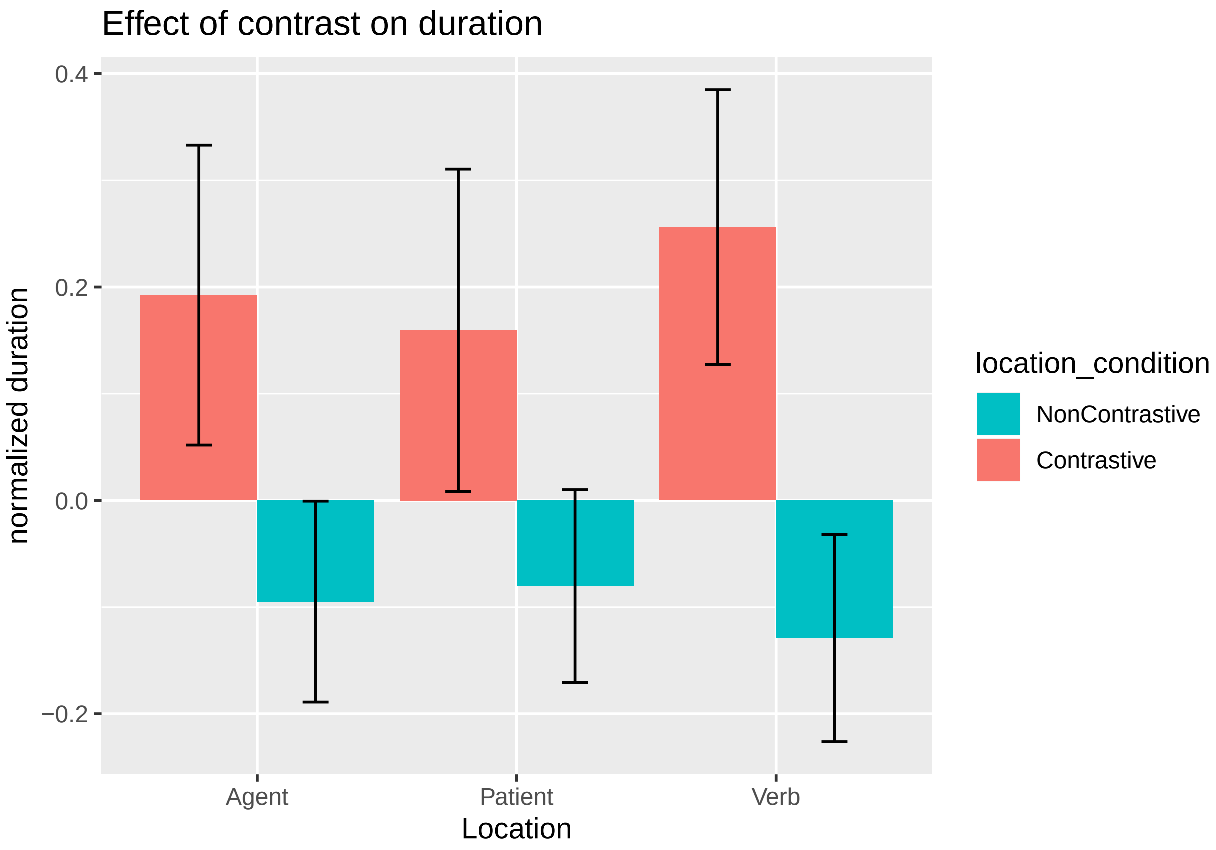
```
# write.csv(df2,'newdf.csv')
# code for getting Nth instance of question
# nthdf <- df1 %>%
#   group_by(participant,Verb, question, condition, word_num) %>%
#   mutate(Appearance=1:n())
#write.csv(nthdf,'nthdf.csv')
```

```r
# 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_Agen
    # 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_with_yes = function(df){


  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=='5

  # 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))
}

process_data_without_yes = function(df){
  df_Agent = df[ df$location_condition!='Control' & df$word_num=='2',]
  # df_Agent inheri row hum from df

  df_Verb = df[ df$location_condition!='Control' & df$word_num=='4',]


  df_Patient = df[ df$location_condition!='Control' & df$word_num=='5',]

  df_Agent$location_condition = mapvalues(df_Agent$location_condition, from=c("Patient", "Verb"), to=c(

  df_Verb$location_condition = mapvalues(df_Verb$location_condition, from=c("Agent", "Patient"), to=c('

  df_Patient$location_condition = mapvalues(df_Patient$location_condition, from=c("Agent", "Verb"), to=

  # 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))
}



# c(df_Verb, df_Agent, df_Patient) %<-% process_data_with_yes(df2)
c(df_Verb, df_Agent, df_Patient) %<-% process_data_without_yes(df2)
```

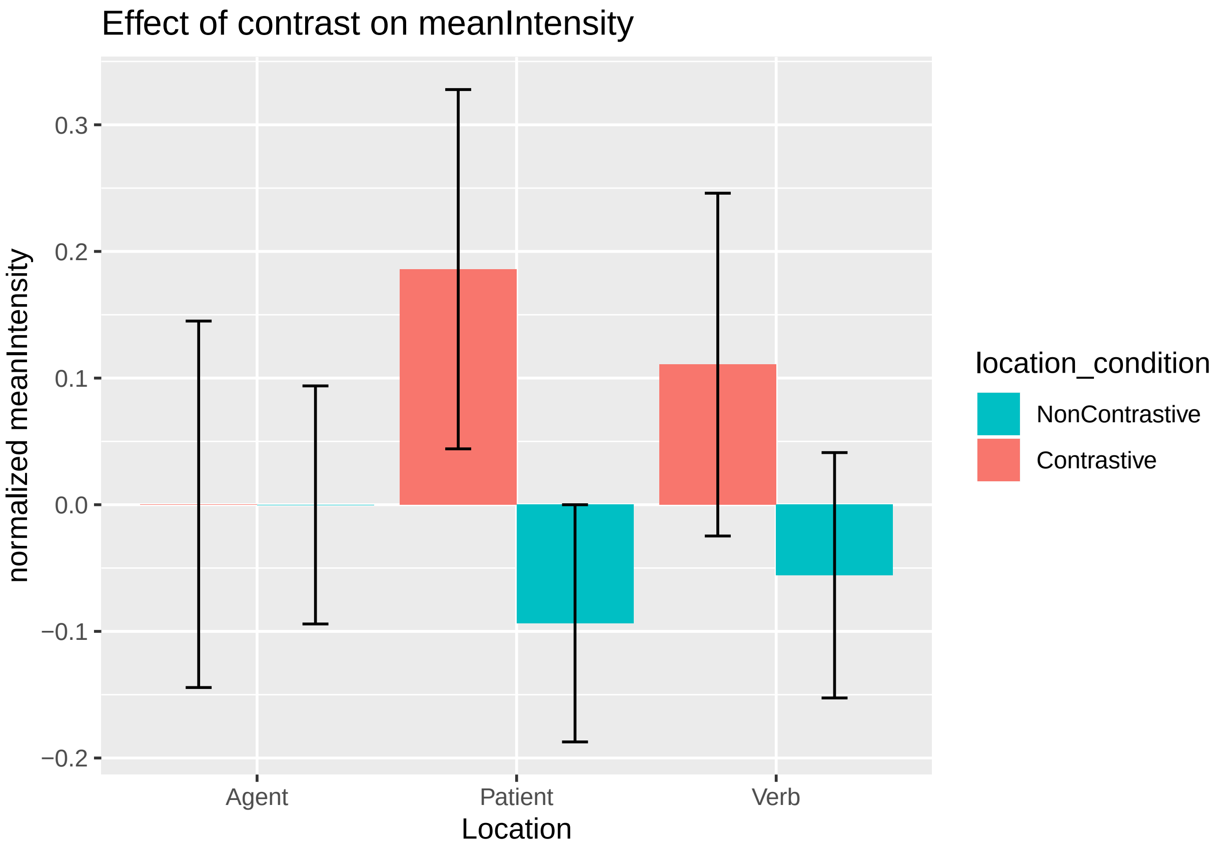
```
## [1] 0
```

```
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))
}
```

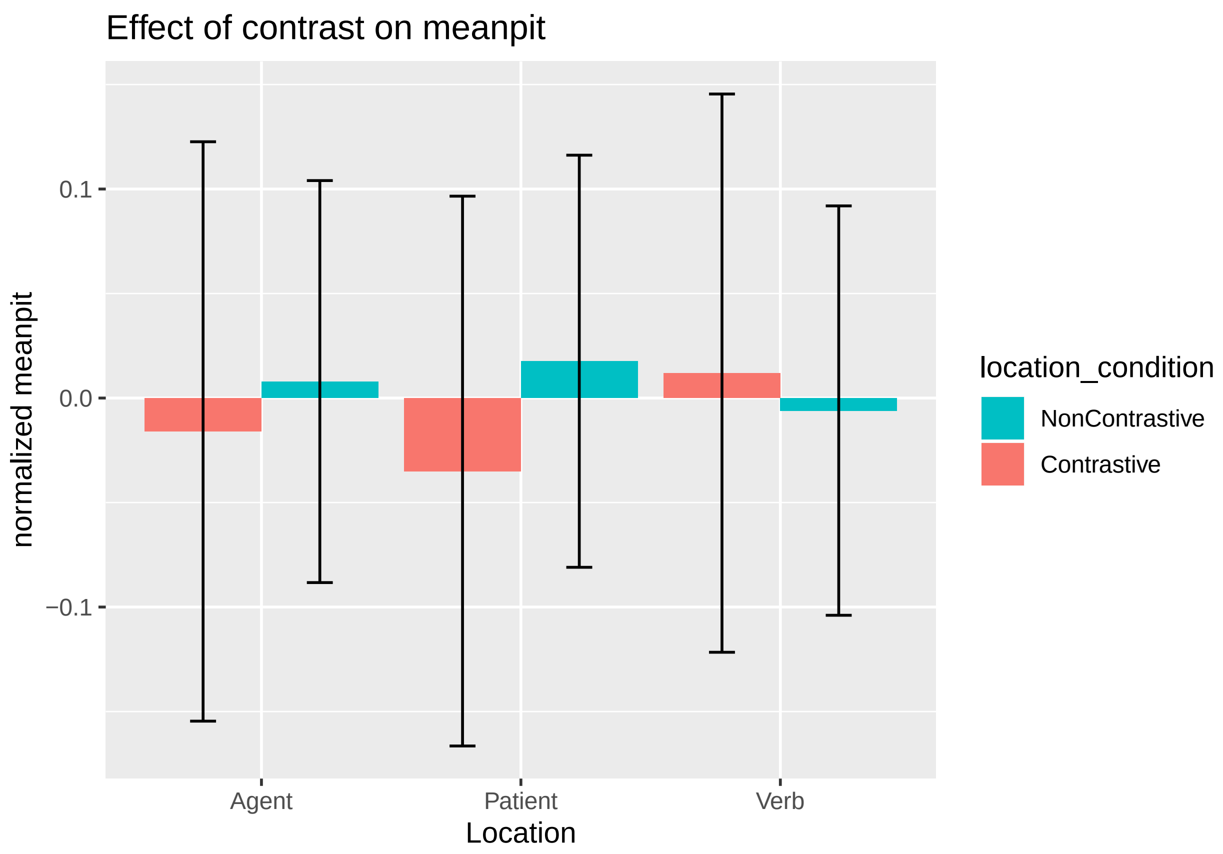
```
## [1] "duration"
```



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


Effect of contrast on meanIntensity
normalized meanIntensity
0.3
0.2
0.1
0.0
−0.1
−0.2
Agent
Patient
Verb
Location
location_condition
NonContrastive
Contrastive

```
## [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: 1619.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0931 -0.5736 -0.1315  0.3678  7.7156
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.380273 0.61666
##              location_conditionControl 0.069092 0.26285  -0.49
##  item_id     (Intercept)               0.006305 0.07941
##              location_conditionControl 0.001382 0.03718  1.00
##  Residual                              0.692996 0.83246
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                0.18065    0.13461   1.342
## location_conditionControl -0.29178    0.08902  -3.278
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.464
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### duration of Patient
```

```
## 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: 1229.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9219 -0.4482 -0.0644  0.2836 12.2532
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.633968 0.79622
##              location_conditionPatient 0.009895 0.09947  1.00
##  item_id     (Intercept)               0.076097 0.27586
##              location_conditionPatient 0.010018 0.10009  1.00
##  Residual                              0.346379 0.58854
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               -0.14899    0.20186  -0.738
## location_conditionPatient  0.23204    0.07314   3.173
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP 0.587
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### duration of Verb
## 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: 1366.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2672 -0.5359 -0.1090  0.4165 10.1165
##
## Random effects:
##  Groups      Name                   Variance Std.Dev. Corr
##  participant (Intercept)            0.57464  0.7580
##              location_conditionVerb 0.01818  0.1348   -0.16
##  item_id     (Intercept)            0.02671  0.1634
##              location_conditionVerb 0.01600  0.1265   1.00
##  Residual                           0.43930  0.6628
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)             -0.1784     0.1640  -1.088
## location_conditionVerb   0.3730     0.0887   4.205
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV 0.240
##
##
## ### meanIntensity of Agent
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.00410871
## (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: 1215.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.7843 -0.4639  0.0891  0.5825  3.2341
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.46128  0.67918
##              location_conditionControl 0.00434  0.06588  0.86
##  item_id     (Intercept)               0.22235  0.47154
##              location_conditionControl 0.01975  0.14052  -1.00
##  Residual                              0.34196  0.58477
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                0.01264    0.26940   0.047
## location_conditionControl  0.01460    0.08706   0.168
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.725
## convergence code: 0
## Model failed to converge with max|grad| = 0.00410871 (tol = 0.002, component 1)
##
##
##
## ### meanIntensity of Patient
```

```
## 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: 1251.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.7806 -0.4832 -0.0263  0.4619  7.0095
##
## Random effects:
##  Groups      Name                      Variance  Std.Dev. Corr
##  participant (Intercept)               0.6196223 0.78716
##              location_conditionPatient 0.0378048 0.19443  0.44
##  item_id     (Intercept)               0.0083835 0.09156
##              location_conditionPatient 0.0007176 0.02679  1.00
##  Residual                              0.3583744 0.59864
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               -0.03181    0.15303  -0.208
## location_conditionPatient  0.25667    0.06422   3.996
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP 0.194
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### meanIntensity 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: 867.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.9570 -0.5707  0.0237  0.5539  3.1277
##
## Random effects:
##  Groups      Name                   Variance Std.Dev. Corr
##  participant (Intercept)            0.859050 0.9268
##              location_conditionVerb 0.003882 0.0623   -0.92
##  item_id     (Intercept)            0.041598 0.2040
##              location_conditionVerb 0.012475 0.1117   1.00
##  Residual                           0.185744 0.4310
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)            0.008589   0.197201   0.044
## location_conditionVerb 0.173751   0.067882   2.560
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV 0.261
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
```

```
## ### 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: 1289.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.0153 -0.3702 -0.0149  0.3128  4.1280
##
## Random effects:
##  Groups      Name                      Variance  Std.Dev. Corr
##  participant (Intercept)               0.7742050 0.87989
##              location_conditionControl 0.0820859 0.28651  -0.55
##  item_id     (Intercept)               0.0042882 0.06548
##              location_conditionControl 0.0007666 0.02769  1.00
##  Residual                              0.3787303 0.61541
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               -0.04420    0.16970  -0.260
## location_conditionControl  0.02438    0.07695   0.317
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.478
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### meanpit of Patient

## 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: 1540.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9107 -0.4385 -0.0301  0.3057  3.3798
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.486538 0.69752
##              location_conditionPatient 0.252681 0.50267  -0.44
##  item_id     (Intercept)               0.005742 0.07577
##              location_conditionPatient 0.001302 0.03608  -1.00
##  Residual                              0.579860 0.76149
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                0.06140    0.13903   0.442
## location_conditionPatient -0.06218    0.11628  -0.535
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP -0.459
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### meanpit 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: 1532.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.8570 -0.3582 -0.0246  0.3600  3.4718
##
## Random effects:
##  Groups      Name                   Variance  Std.Dev. Corr
##  participant (Intercept)            0.4400268 0.66335
##              location_conditionVerb 0.2305780 0.48019  -0.28
##  item_id     (Intercept)            0.0023997 0.04899
##              location_conditionVerb 0.0006456 0.02541  -1.00
##  Residual                           0.5735677 0.75734
## Number of obs: 621, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)             0.01068    0.13033   0.082
## location_conditionVerb  0.01107    0.11190   0.099
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV -0.339
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

```
##
##
```

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