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_id
  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.

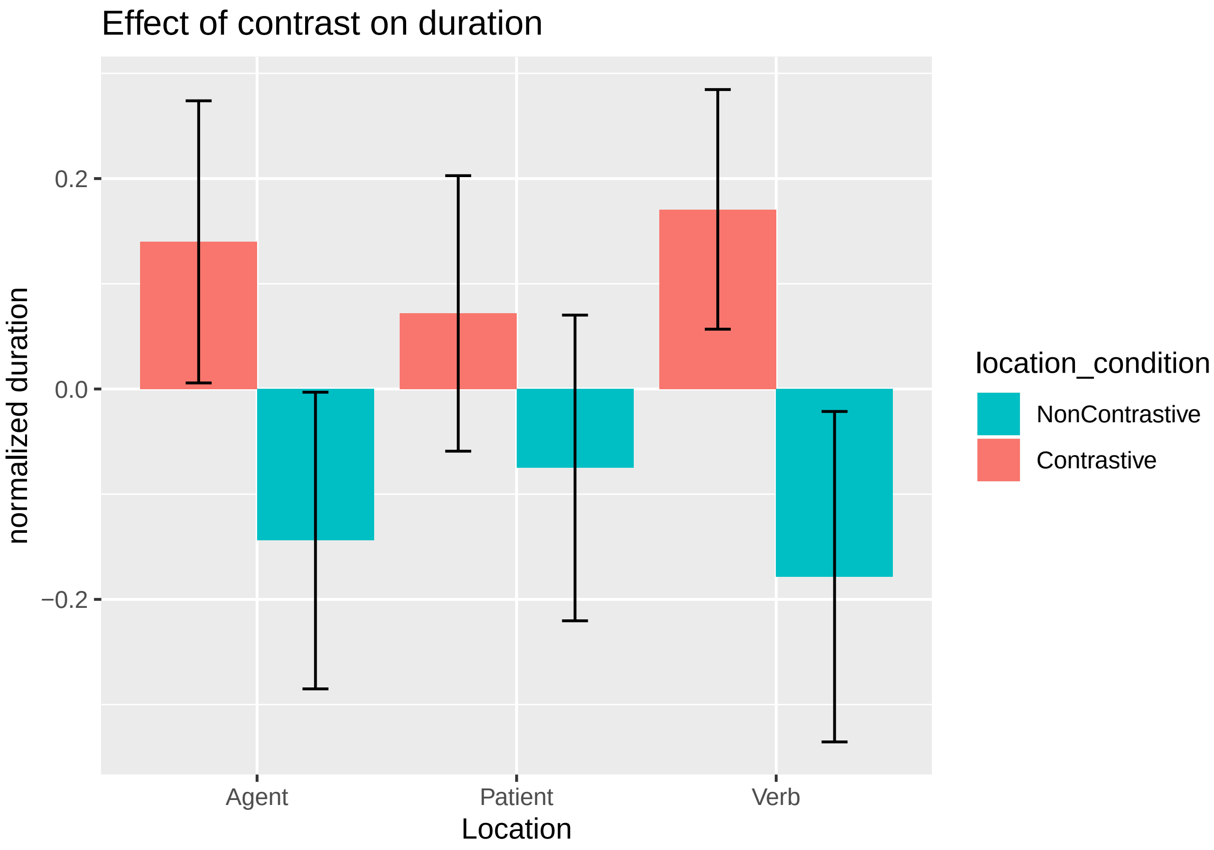
```
# write.csv(df2,'newdf.csv')
# code for getting Nth instance of question
# nthdf <- df1 %>%
#   group_by(participant,Verb, question, condition, word_num) %>%
```

```r
#   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_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)
```

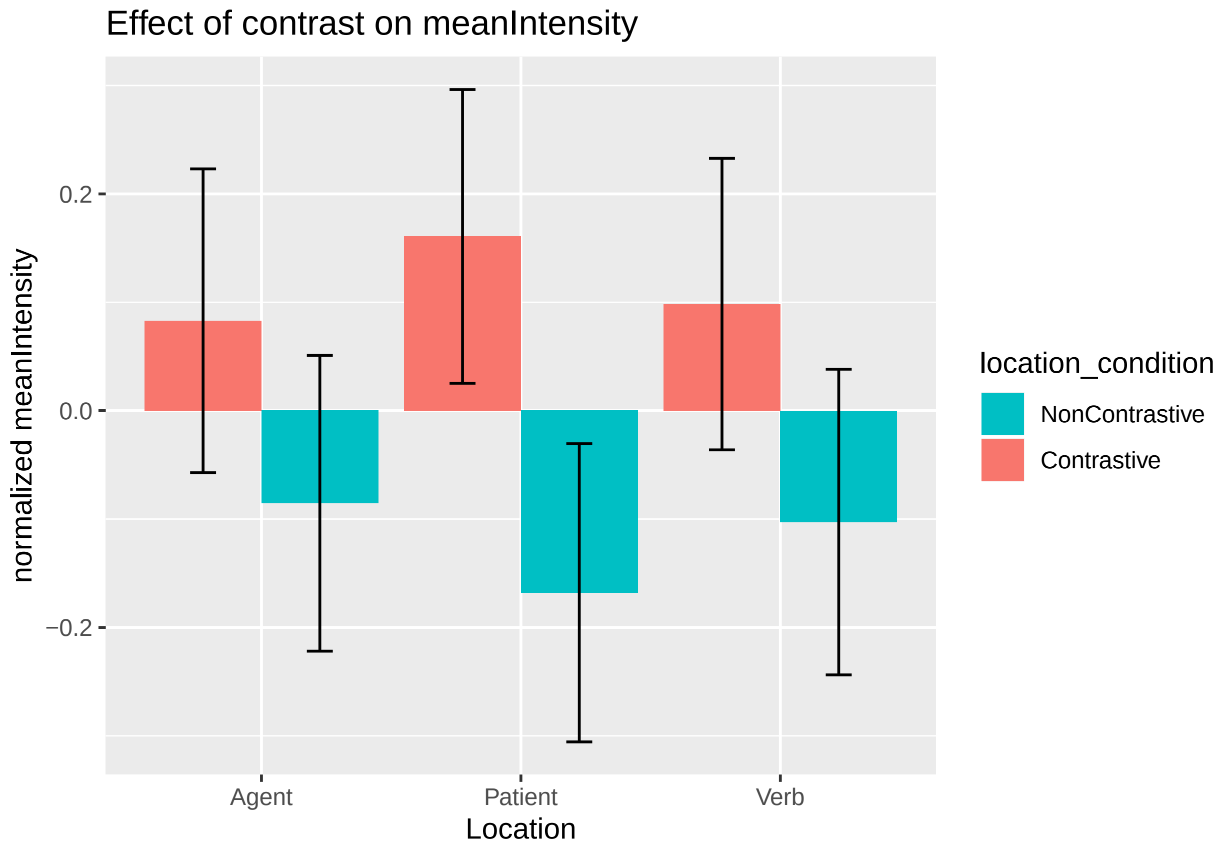
```
## [1] 0
```

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



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))
}
```

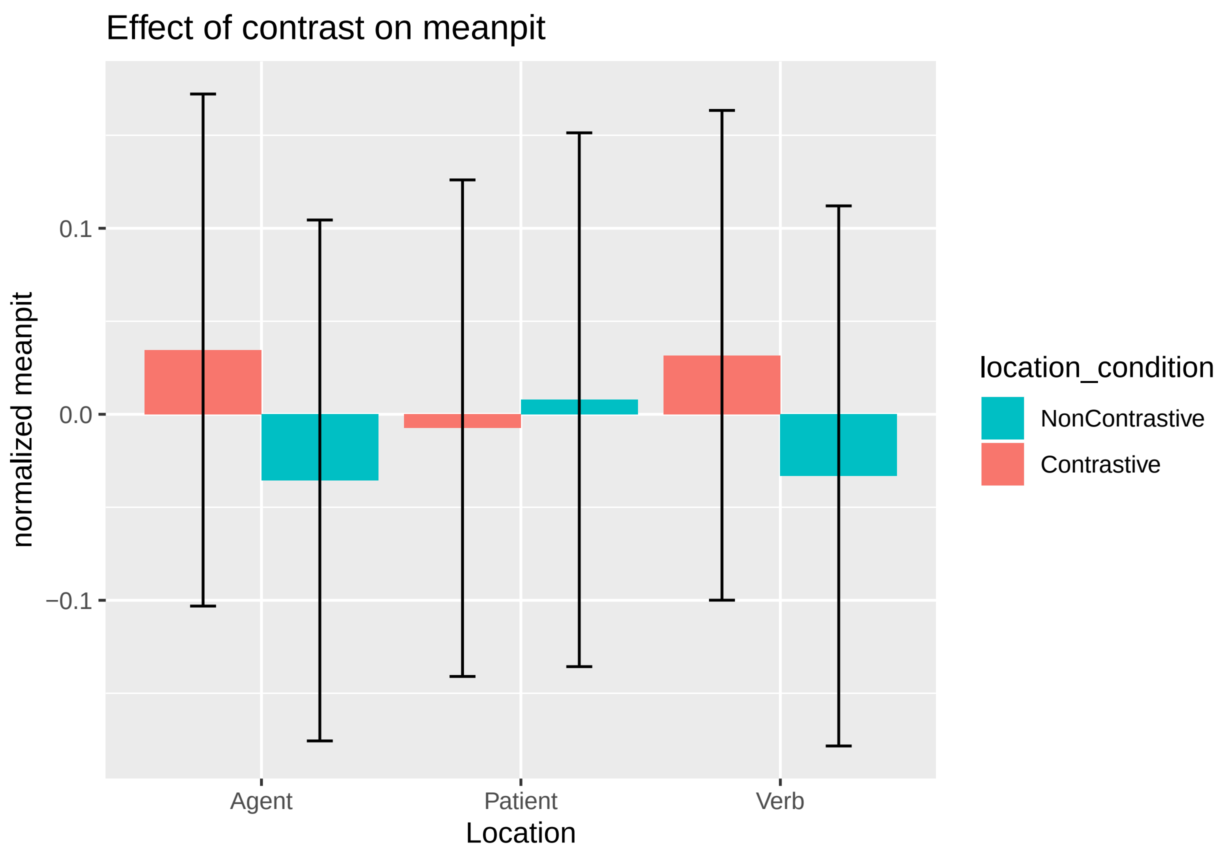
```
## [1] "duration"
```



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


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

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



```r
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: 1082.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3249 -0.5305 -0.0902  0.3598 10.0210
##
## Random effects:
##  Groups      Name                      Variance  Std.Dev. Corr
##  participant (Intercept)               3.073e-01 0.554365
##              location_conditionControl 1.149e-01 0.339019 -0.50
##  item_id     (Intercept)               3.132e-04 0.017699
##              location_conditionControl 7.901e-05 0.008889 -1.00
##  Residual                              7.278e-01 0.853089
## Number of obs: 404, groups:  participant, 30; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                 0.1340     0.1205   1.111
## location_conditionControl  -0.2669     0.1079  -2.473
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.537
## 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: 921.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4236 -0.3636 -0.0689  0.2324  8.5993
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.60282  0.77641
##              location_conditionPatient 0.00122  0.03493  -1.00
##  item_id     (Intercept)               0.02170  0.14730
##              location_conditionPatient 0.02784  0.16687  1.00
##  Residual                              0.44421  0.66649
## Number of obs: 407, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                -0.1366     0.1677  -0.814
## location_conditionPatient   0.1466     0.1074   1.365
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP 0.156
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### 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: 987.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8951 -0.4599 -0.1130  0.2885 11.9039
##
## Random effects:
##  Groups      Name                   Variance Std.Dev. Corr
##  participant (Intercept)            0.492102 0.70150
##              location_conditionVerb 0.044807 0.21168  -0.56
##  item_id     (Intercept)            0.049624 0.22276
##              location_conditionVerb 0.000537 0.02317   1.00
##  Residual                           0.534102 0.73082
## Number of obs: 407, groups:  participant, 30; item_id, 4
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)            -0.19095    0.17991  -1.061
## location_conditionVerb  0.34526    0.08487   4.068
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV -0.299
## 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: 862.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.8155 -0.3832  0.1032  0.5756  2.5364
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.43328  0.6582
##              location_conditionControl 0.01201  0.1096   0.45
##  item_id     (Intercept)               0.20921  0.4574
##              location_conditionControl 0.02267  0.1506   -1.00
##  Residual                              0.38692  0.6220
## Number of obs: 404, groups:  participant, 30; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                0.08184    0.26291   0.311
## location_conditionControl -0.14093    0.10039  -1.404
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.681
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
##
## ### 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: 854.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8866 -0.4456 -0.0292  0.4933  6.6147
##
## Random effects:
##  Groups      Name                      Variance  Std.Dev. Corr
##  participant (Intercept)               5.974e-01 0.772933
##              location_conditionPatient 2.949e-02 0.171738 0.27
##  item_id     (Intercept)               1.377e-02 0.117326
##              location_conditionPatient 1.769e-05 0.004206 1.00
##  Residual                              3.667e-01 0.605577
## Number of obs: 407, groups:  participant, 31; item_id, 4
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               -0.09148    0.15943  -0.574
## location_conditionPatient  0.27882    0.06944   4.015
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP -0.066
## 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: 663.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8602 -0.5391  0.0109  0.5239  4.6048
##
## Random effects:
##  Groups      Name                   Variance Std.Dev. Corr
##  participant (Intercept)            0.76267  0.8733
##              location_conditionVerb 0.06679  0.2584   -0.37
##  item_id     (Intercept)            0.02914  0.1707
##              location_conditionVerb 0.01965  0.1402   1.00
##  Residual                           0.20902  0.4572
## Number of obs: 407, groups:  participant, 30; item_id, 4
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)            -0.05164    0.18544  -0.278
## location_conditionVerb  0.16183    0.09764   1.657
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV 0.100
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

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

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.0026524
## (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: 812.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4063 -0.4398  0.0281  0.3220  4.4421
##
## Random effects:
##  Groups      Name                      Variance Std.Dev. Corr
##  participant (Intercept)               0.783961 0.88542
##              location_conditionControl 0.078860 0.28082  -0.51
##  item_id     (Intercept)               0.002487 0.04987
##              location_conditionControl 0.011505 0.10726  -0.29
##  Residual                              0.325105 0.57018
## Number of obs: 404, groups:  participant, 30; item_id, 4
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)                0.002474   0.170173   0.015
## location_conditionControl -0.048885   0.095600  -0.511
##
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnC -0.393
## convergence code: 0
## Model failed to converge with max|grad| = 0.0026524 (tol = 0.002, component 1)
##
##
##
## ### 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: 1041.9
##
## Scaled residuals:
```

```
##     Min      1Q  Median      3Q     Max 
## -2.8432 -0.4220 -0.0387  0.4021  3.2232 
## 
## Random effects:
##  Groups      Name                      Variance  Std.Dev.  Corr 
##  participant (Intercept)               3.735e-01 0.6111849      
##              location_conditionPatient 1.069e-01 0.3269094 -0.17
##  item_id     (Intercept)               8.634e-08 0.0002938      
##              location_conditionPatient 4.488e-07 0.0006699 -1.00
##  Residual                              6.223e-01 0.7888465      
## Number of obs: 407, groups:  participant, 31; item_id, 4
## 
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                0.02756    0.12778   0.216
## location_conditionPatient -0.03035    0.10144  -0.299
## 
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnP -0.364
## 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: 953.5
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -2.8928 -0.3874 -0.0026  0.3081  3.9211 
## 
## Random effects:
##  Groups      Name                   Variance Std.Dev. Corr 
##  participant (Intercept)            0.5736   0.7574        
##              location_conditionVerb 0.1017   0.3188   -0.41
##  item_id     (Intercept)            0.0000   0.0000        
##              location_conditionVerb 0.0000   0.0000    NaN 
##  Residual                           0.4805   0.6932        
## Number of obs: 407, groups:  participant, 30; item_id, 4
## 
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)            -0.04444    0.15007  -0.296
## location_conditionVerb  0.07480    0.09301   0.804
## 
```

```
## Correlation of Fixed Effects:
##             (Intr)
## lctn_cndtnV -0.450
## convergence code: 0
## boundary (singular) fit: see ?isSingular
##
##
```

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