

Nested Subject-Verb Dependencies

12/18/2019

Contents

Preparing the data	1
PLOT - error rates	3
STATS - objrel-SR: violation_position * subjects-congruency	3
STATS - objrel-LR: violation_position * subjects-congruency	5
STATS: objrel (SR vs. LR) - long * subjects-congruency (inner verb only)	7
STATS: objrel (SR vs. LR) - long * violation-position * subjects-congruency (inner verb only)	8

Preparing the data

Variables:

- nested: 1 if center embedding (objrel or objrel-nounpp), 0 mental embedding (SR or LR)
- long: 1 if LR (objrel-nounpp or mental embedding LR), 0 SR (objrel, mental embedding SR)
- violation_position: “inner” or “outer” - verb on which the violation occurred (mental embedding has only “inner” - verb in embedded clause)
- congruent_subjects: 1 if the two first subject nouns agree on number, else 0.
- number_v2: “singular” or “plural”, based on the number of the second noun.
- congruent_attractor: 1 if the two last nouns agree on number, else 0.

For example, the last three variables defines: ‘SSS’ = (1, ‘singular’, 1), ‘SSP’ = (1, ‘singular’, 0), ‘SPS’ = (0, ‘plural’, 0), etc

Note that when long=0 (i.e., no attractor noun), then there are only four conditions (SS, SP, PS, PP), and therefore congruent_attractor=‘NA’: ‘SS’=(1, ‘singular’, ‘NA’), etc.

```
Data <- read.csv(file="../Paradigm/Results/dataframe_results_all_trials.csv", header=TRUE, sep=",")
Data <- subset(Data, trial_type == "Violation") # Take only trials in which there was a violation
Data <- subset(Data, violation_position != "other") # Take only trials in which the violation was on the
Data <- subset(Data, valid_answer != "REJECTED") # remove from it rejected trials (in which subject did

# Remove the following columns (which will not be analyzed):
Data$violation_type <- NULL
Data$correct_wrong <- NULL
Data$block <- NULL
Data$trial_num <- NULL
Data$RT <- NULL
Data$slide_num_of_viol <- NULL
```

```

# Define 3 new binary columns that will classify all conditions ('SSS', 'SSP', 'SPS'...) based on whether
Data$congruent_subjects <- ifelse(Data$condition == 'SSS' | Data$condition == 'SSP' | Data$condition ==
Data$number_v2 <- ifelse(Data$condition == 'SSS' | Data$condition == 'SSP' | Data$condition == 'PSS' |
Data$congruent_attractor <- ifelse(Data$condition == 'SSS' | Data$condition == 'PSS' | Data$condition ==
Data$correct_wrong <- NULL
Data$condition <- NULL

# Define the main fixed variables (nested and long), based on the type of syntactic structure.
Data$nested <- ifelse(Data$sentence_type == "objrel" | Data$sentence_type == "objrel_nounpp", 1, 0)
Data$long <- ifelse(Data$sentence_type == "embedding_mental_LR" | Data$sentence_type == "objrel_nounpp"
Data$sentence_type <- NULL
Data$trial_type <- NULL
Data$valid_answer <- ifelse(Data$valid_answer == "CORRECT", 1, 0)
Data[] <- lapply(Data, function(x) if(is.factor(x)) factor(x) else x) # Remove all empty level in dataf
Data[] <- lapply(Data, factor) # change all variables to 'factor' type.
Data$valid_answer <- as.numeric(Data$valid_answer)-1

str(Data)

```

```

## 'data.frame':    9643 obs. of  10 variables:
## $ subject          : Factor w/ 55 levels "21","22","23",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ violation_position : Factor w/ 2 levels "inner","outer": 1 1 1 1 1 1 1 1 1 1 ...
## $ congruent_subjects : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ congruent_attractor : Factor w/ 3 levels "0","1","NA": 1 1 1 1 1 1 1 1 1 1 ...
## $ congruent_subjects_attractor: Factor w/ 5 levels "", "False_False",...: 2 2 2 2 2 2 2 2 2 2 ...
## $ valid_answer      : num  1 1 1 1 1 1 1 1 1 1 ...
## $ is_outlier        : Factor w/ 2 levels "False","True": 1 1 1 1 1 1 1 1 1 1 ...
## $ number_v2         : Factor w/ 2 levels "plural","singular": 2 2 2 2 2 1 1 1 1 1 ...
## $ nested           : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ long              : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...

```

```
summary(Data)
```

```

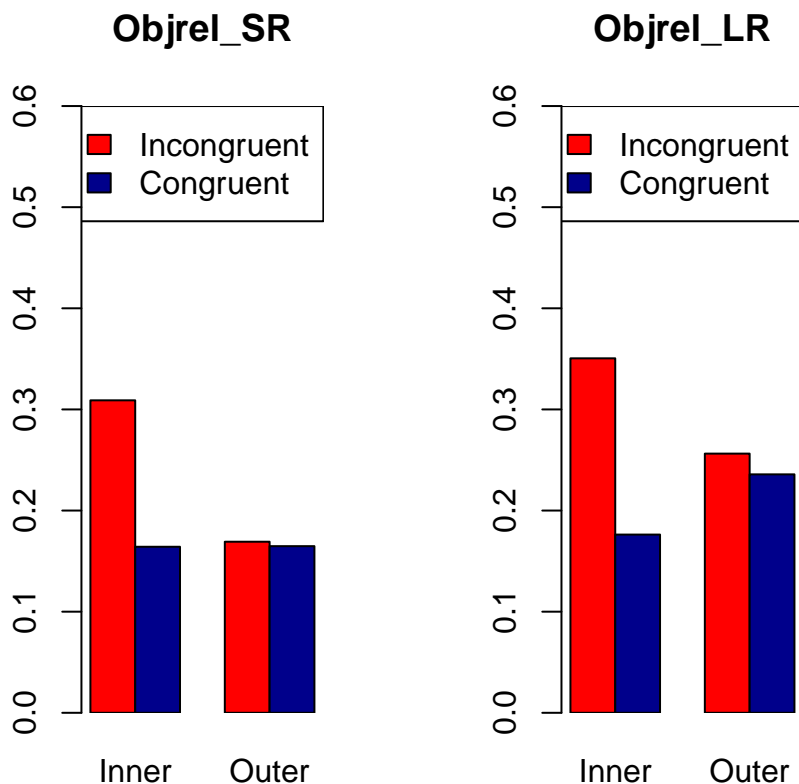
##      subject      violation_position congruent_subjects congruent_attractor
## 24      : 180      inner:6431          0:4815          0 :3219
## 28      : 180      outer:3212          1:4828          1 :3221
## 30      : 180
## 34      : 180
## 46      : 180
## 52      : 180
## (Other):8563
## congruent_subjects_attractor valid_answer is_outlier      number_v2
##      :3203      Min.    :0.0000 False:9465 plural  :5627
## False_False:1605      1st Qu.:1.0000 True : 178 singular:4016
## False_True :1610      Median :1.0000
## True_False :1614      Mean    :0.8236
## True_True  :1611      3rd Qu.:1.0000
##      Max.    :1.0000
##
## nested long
## 0:3216 0:3203
## 1:6427 1:6440
##

```

```
##
##
##
##
```

PLOT - error rates

```
par(mfrow=c(1,2))
for (l in 0:1) {
  struct = ifelse(l=="0", "Objrel_SR", "Objrel_LR")
  curr_data = subset(Data, nested == 1 & long == 1)
  curr_bysuj = with(curr_data, tapply(valid_answer, list(congruent_subjects=congruent_subjects, violation_position=violation_position), FUN=function(x) {
    curr_bysuj <- 1 - curr_bysuj
  }, MARGIN=2))
  barplot(curr_bysuj, col=c("red", "darkblue"), main = struct, ylim=c(0,0.6), names.arg = c("Inner", "Outer"),
    legend("topright", c("Incongruent", "Congruent"), fill=c("red", "darkblue"))
}
```



STATS - objrel-SR: violation_position * subjects-congruency

```
print('ANOVA')

## [1] "ANOVA"

curr_Data = subset(Data, long == 0 & nested == 1)
bysuj = with(curr_Data, aggregate(valid_answer, list(subject=subject, violation_position=violation_position), FUN=function(x) {
  bysuj$error <- 1 - bysuj$x
}
```

```
anov = aov(error ~ violation_position * congruent_subjects + Error(subject/(violation_position*congruent
summary(anov)
```

```
##
## Error: subject
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals 54   2.128  0.03942
##
## Error: subject:violation_position
##           Df Sum Sq Mean Sq F value  Pr(>F)
## violation_position 1   0.224   0.2240   7.493 0.00837 **
## Residuals         54   1.614   0.0299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:congruent_subjects
##           Df Sum Sq Mean Sq F value  Pr(>F)
## congruent_subjects 1 0.2924 0.29238   15.33 0.000255 ***
## Residuals         54 1.0299 0.01907
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:violation_position:congruent_subjects
##           Df Sum Sq Mean Sq F value  Pr(>F)
## violation_position:congruent_subjects 1 0.2744 0.2744   11.25 0.00146 **
## Residuals         54 1.3178 0.0244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('GLMM')
```

```
## [1] "GLMM"
```

```
glmm_with_random <- glmer(valid_answer ~ violation_position * congruent_subjects + (1 | subject), data=
summary(glmm_with_random)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: valid_answer ~ violation_position * congruent_subjects + (1 |
## subject)
## Data: curr_Data
##
##           AIC          BIC    logLik deviance df.resid
##    2083.4     2111.8   -1036.7   2073.4      2131
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0224  0.2995  0.4205  0.5057  0.9264
##
## Random effects:
## Groups Name             Variance Std.Dev.
## subject (Intercept) 0.2548    0.5048
## Number of obs: 2136, groups: subject, 55
##
```

```
## Fixed effects:
##
##              Estimate Std. Error z value
## (Intercept)      0.8486    0.1182   7.178
## violation_positionouter      0.8152    0.1513   5.388
## congruent_subjects1      0.8544    0.1520   5.619
## violation_positionouter:congruent_subjects1 -0.8212    0.2251  -3.648
##
##              Pr(>|z|)
## (Intercept)      7.10e-13 ***
## violation_positionouter      7.12e-08 ***
## congruent_subjects1      1.92e-08 ***
## violation_positionouter:congruent_subjects1 0.000264 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) vltn_p cngr_1
## vltn_pstntr -0.506
## cngrnt_sb1 -0.503  0.401
## vltn_pst:_1  0.340 -0.672 -0.675

#glmm_wo_random <- glm(valid_answer ~ violation_position * congruent_subjects, data=curr_Data, family="")
#summary(glmm_wo_random)
#anova(glmm_with_random, glmm_wo_random)
```

STATS - objrel-LR: violation__position * subjects-congruency

```
print('ANOVA')

## [1] "ANOVA"

curr_Data = subset(Data, long == 1 & nested == 1)
bysuj = with(curr_Data, aggregate(valid_answer, list(subject=subject, violation_position=violation_position), FUN=mean))
bysuj$error <- 1 - bysuj$x
anov = aov(error ~ violation_position * congruent_subjects + Error(subject/(violation_position*congruent_subjects)))
summary(anov)

##
## Error: subject
##              Df Sum Sq Mean Sq F value Pr(>F)
## Residuals  54   1.96  0.0363
##
## Error: subject:violation_position
##              Df Sum Sq Mean Sq F value Pr(>F)
## violation_position  1 0.0101 0.01009   0.445  0.508
## Residuals        54 1.2258 0.02270
##
## Error: subject:congruent_subjects
##              Df Sum Sq Mean Sq F value  Pr(>F)
## congruent_subjects  1 0.5149  0.5149   53.46 1.28e-09 ***
## Residuals        54 0.5201  0.0096
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Error: subject:violation_position:congruent_subjects
##               Df Sum Sq Mean Sq F value    Pr(>F)
## violation_position:congruent_subjects  1 0.3247   0.3247    26.32 4.03e-06 ***
## Residuals                        54 0.6662   0.0123
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('GLMM')
```

```
## [1] "GLMM"
```

```
glmm_with_random <- glmer(valid_answer ~ violation_position * congruent_subjects + (1 | subject), data=
summary(glmm_with_random)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: valid_answer ~ violation_position * congruent_subjects + (1 |
## subject)
## Data: curr_Data
##
##      AIC      BIC    logLik deviance df.resid
##  4709.4   4741.2  -2349.7   4699.4     4286
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.2081 -0.9333  0.4805  0.5919  1.1501
##
## Random effects:
## Groups Name          Variance Std.Dev.
## subject (Intercept) 0.1956    0.4422
## Number of obs: 4291, groups: subject, 55
##
## Fixed effects:
##
##              Estimate Std. Error z value
## (Intercept)      0.64694    0.08878   7.287
## violation_positionouter      0.46799    0.09672   4.839
## congruent_subjects1      0.95951    0.10426   9.203
## violation_positionouter:congruent_subjects1 -0.84739    0.14580  -5.812
##
##              Pr(>|z|)
## (Intercept)      3.17e-13 ***
## violation_positionouter      1.31e-06 ***
## congruent_subjects1      < 2e-16 ***
## violation_positionouter:congruent_subjects1 6.17e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) vltn_p cngr_1
## vltn_pstntr -0.497
## cngrnt_sbj1 -0.460  0.427
## vltn_pst:_1  0.329 -0.664 -0.715
```

STATS: objrel (SR vs. LR) - long * subjects-congruency (inner verb only)

```
print('ANOVA')
```

```
## [1] "ANOVA"
```

```
curr_Data = subset(Data, violation_position == 'inner' & nested == 1)
bysuj = with(curr_Data, aggregate(valid_answer, list(subject=subject, long=long, congruent_subjects=congruent_subjects), FUN=function(x) {
  bysuj$error <- 1 - bysuj$x
  anov = aov(error ~ long * congruent_subjects + Error(subject/(long*congruent_subjects)), data=bysuj)
  summary(anov)
}))
```

```
##
## Error: subject
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals 54  1.641  0.0304
##
## Error: subject:long
##           Df Sum Sq Mean Sq F value Pr(>F)
## long       1 0.0317 0.03174   1.386  0.244
## Residuals 54 1.2365 0.02290
##
## Error: subject:congruent_subjects
##           Df Sum Sq Mean Sq F value  Pr(>F)
## congruent_subjects 1  1.383  1.3830  66.57 5.47e-11 ***
## Residuals         54  1.122  0.0208
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:long:congruent_subjects
##           Df Sum Sq Mean Sq F value Pr(>F)
## long:congruent_subjects 1 0.0124 0.01241  0.836  0.365
## Residuals         54 0.8019 0.01485
```

```
print('GLMM')
```

```
## [1] "GLMM"
```

```
glmm_with_random <- glmer(valid_answer ~ long * congruent_subjects + (1 | subject), data=curr_Data, family=binomial)
summary(glmm_with_random)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: valid_answer ~ long * congruent_subjects + (1 | subject)
## Data: curr_Data
##
##           AIC          BIC    logLik deviance df.resid
##    3506.5     3536.9  -1748.2   3496.5     3210
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0934 -1.0589  0.4585  0.6245  0.9555
##
```

```
## Random effects:
## Groups Name Variance Std.Dev.
## subject (Intercept) 0.1384 0.372
## Number of obs: 3215, groups: subject, 55
##
## Fixed effects:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.8344 0.1076 7.752 9.04e-15 ***
## long1 -0.1944 0.1150 -1.690 0.0911 .
## congruent_subjects1 0.8399 0.1510 5.562 2.67e-08 ***
## long1:congruent_subjects1 0.1091 0.1830 0.596 0.5511
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) long1 cngr_1
## long1 -0.728
## cngrnt_sbj1 -0.551 0.517
## lng1:cngr_1 0.457 -0.629 -0.824
```

STATS: objrel (SR vs. LR) - long * violation-position * subjects-congruency (inner verb only)

```
print('ANOVA')
```

```
## [1] "ANOVA"
```

```
curr_Data = subset(Data, nested == 1)
bysuj = with(curr_Data, aggregate(valid_answer, list(subject=subject, long=long, violation_position=violation_position), FUN=function(x) {
  bysuj$error <- 1 - bysuj$x
  anov = aov(error ~ long * congruent_subjects*violation_position + Error(subject/(long*congruent_subjects))
  summary(anov)
```

```
##
## Error: subject
## Df Sum Sq Mean Sq F value Pr(>F)
## Residuals 54 3.092 0.05726
##
## Error: subject:long
## Df Sum Sq Mean Sq F value Pr(>F)
## long 1 0.2658 0.26584 14.4 0.000375 ***
## Residuals 54 0.9966 0.01846
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:congruent_subjects
## Df Sum Sq Mean Sq F value Pr(>F)
## congruent_subjects 1 0.7917 0.7917 51.59 2.08e-09 ***
## Residuals 54 0.8287 0.0153
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```



```

## Error: subject:violation_position
##               Df Sum Sq Mean Sq F value Pr(>F)
## violation_position  1 0.1646 0.16460    4.858 0.0318 *
## Residuals          54 1.8295 0.03388
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:long:congruent_subjects
##               Df Sum Sq Mean Sq F value Pr(>F)
## long:congruent_subjects  1 0.0156 0.01564    1.171 0.284
## Residuals          54 0.7214 0.01336
##
## Error: subject:long:violation_position
##               Df Sum Sq Mean Sq F value Pr(>F)
## long:violation_position  1 0.0695 0.06951    3.714 0.0592 .
## Residuals          54 1.0107 0.01872
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:congruent_subjects:violation_position
##               Df Sum Sq Mean Sq F value Pr(>F)
## congruent_subjects:violation_position  1 0.5981 0.5981   24.75 7e-06 ***
## Residuals          54 1.3048 0.0242
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:long:congruent_subjects:violation_position
##               Df Sum Sq Mean Sq F value Pr(>F)
## long:congruent_subjects:violation_position  1 0.0011 0.001056    0.084 0.773
## Residuals          54 0.6792 0.012578

print('GLMM')

## [1] "GLMM"

glmm_with_random <- glmer(valid_answer ~ long * congruent_subjects * violation_position + (1 | subject)
summary(glmm_with_random)

## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: valid_answer ~ long * congruent_subjects * violation_position +
## (1 | subject)
## Data: curr_Data
##
##      AIC      BIC    logLik deviance df.resid
## 6776.0   6836.9  -3379.0   6758.0     6418
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.4126  0.3291  0.4569  0.5787  1.1319
##
## Random effects:
## Groups Name Variance Std.Dev.
## subject (Intercept) 0.1865 0.4319

```

```

## Number of obs: 6427, groups:  subject, 55
##
## Fixed effects:
##
##                                     Estimate Std. Error z value
## (Intercept)                        0.83900    0.11190   7.498
## long1                             -0.19571    0.11568  -1.692
## congruent_subjects1                 0.84793    0.15169   5.590
## violation_positionouter             0.80946    0.15097   5.362
## long1:congruent_subjects1           0.10978    0.18389   0.597
## long1:violation_positionouter      -0.34210    0.17920  -1.909
## congruent_subjects1:violation_positionouter -0.81433    0.22496  -3.620
## long1:congruent_subjects1:violation_positionouter -0.03188    0.26795  -0.119
##                                     Pr(>|z|)
## (Intercept)                        6.50e-14 ***
## long1                             0.090669 .
## congruent_subjects1                2.27e-08 ***
## violation_positionouter             8.24e-08 ***
## long1:congruent_subjects1           0.550500
## long1:violation_positionouter       0.056254 .
## congruent_subjects1:violation_positionouter 0.000295 ***
## long1:congruent_subjects1:violation_positionouter 0.905308
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr) long1  cngr_1 vltn_p ln1:_1 lng1:_ cn_1:_
## long1      -0.703
## cngrnt_sb1 -0.535  0.518
## vltn_pstntr -0.537  0.521  0.398
## lng1:cngr_1  0.443 -0.629 -0.824 -0.328
## lng1:vltn_p  0.454 -0.646 -0.335 -0.842  0.406
## cngrnt_s1:_  0.361 -0.350 -0.674 -0.671  0.556  0.565
## lng1:cn_1:_ -0.304  0.432  0.566  0.563 -0.686 -0.669 -0.839

```