Vignette Analysis with Linear Mixed Effects

Steps: (1) get in the data, (2) transform it into something that can plug into statistical models, (3) test two types of model: vignette types separate (emotion, syntax, etc.), then all together in a "competitive" model (emotion vs. syntax vs...). In general, emotion comes out on top... with the rest non-significant.

```
library(lme4) # the regression (multi-level) library
```

```
## Warning: package 'lme4' was built under R version 3.1.3
```

```
## Loading required package: Matrix
```

```
library(pander) # library for displaying tables
```

```
## Warning: package 'pander' was built under R version 3.1.3
```

```
library(ggplot2) # plotting library
```

```
## Warning: package 'ggplot2' was built under R version 3.1.3
```

```
panderOptions('knitr.auto.asis', FALSE)

setwd('~/Dropbox/projects/studies/JointActionTake2/entitativity/tailer.entitativity.analysis') # move t
#fname = 'experiment.1.feel.in.questions.csv'
fname = 'experiment.2.no.feelings.csv'
a = read.csv(fname,header=T,stringsAsFactors=F) # load in the sheet into a variable
                                                # note first line contains question content
#pander(a[1,]) # check out the first row of data
#pander(a[1:5,1:4]) # look at first 10 columns
#pander(colnames(a))

# let's extract the data into a pleasant sheet with factors needed
types = c("Syn","Emo","Lex","Spac")
for (i in 2:dim(a)[1]) { # loop through all entries (skipped first line: content)
  for (typ in types) { # loop through stim types
    for (j in 1:3) { # loop through stimulus #
      colnA = which(colnames(a)==paste(typ,j,'..A',sep=''))
      colnN = which(colnames(a)==paste(typ,j,'..N',sep=''))
      targetCol = 0
      if (a[i,colnA]==1) { # this was an align
        align = "aligned"
        targetCol = colnA
      } else if (a[i,colnN]==1) {
        align = "non-aligned"
        targetCol = colnN
      }
      if (targetCol>0) {
        dataRow = data.frame(a[i,1],typ,j,align,substr(a[1,targetCol],1,50),a[i,c((targetCol+1):(target
        respnames = c()
```

```
        for (nm in a[1,c((targetCol+1):(targetCol+5))]) {
          words = unlist(strsplit(nm,' '))
          respnames = c(respnames,paste(words[(length(words)-4):length(words)],collapse='_'))
        }
        colnames(dataRow) = c('id','typ','n','align','item',respnames)
        if (i==2) {
          allData = dataRow
        } else {
          allData = rbind(allData,dataRow)
        }
      }
    }
  }
}

# here's what it now ooks like
pander(allData[1:4,])
```

Table 1: Table continues below

	id	typ	n	align
2	R_OP5LjMOTxSWdPhL	Spac	3	aligned
3	R_1LXNCwR2Lyizht6	Syn	1	non-aligned
31	R_1LXNCwR2Lyizht6	Syn	2	aligned
32	R_1LXNCwR2Lyizht6	Syn	3	non-aligned

Table 2: Table continues below

	item
2	A man walks into the Memphis Police Department and
3	Carol has accepted an invitation to attend a murde
31	John, a small business owner, has been chosen for
32	Alice, a college sophomore, has an appointment to

Table 3: Table continues below

	individuals_form_a_unified_group
2	1
3	4
31	4
32	1

Table 4: Table continues below

	if_I_observed_this_conversation
2	1
3	4
31	2
32	1

Table 5: Table continues below

	these_individuals_can_work_together
2	1
3	4
31	4
32	1

Table 6: Table continues below

	agree)-These_people_are_like_minded
2	1
3	4
31	4
32	1

	(strongly_agree)-These_people_share_goals
2	1
3	5
31	4
32	1

```
dim(table(as.character(allData$id)))
```

[1] 74

```
allData = allData[allData$id %in% names(table(allData$id))[table(allData$id)==12],]
dim(table(as.character(allData$id)))
```

[1] 72

```
if (fname=='experiment.2.no.feelings.csv') {
  colnames(allData) = gsub('\\(','',colnames(allData)) # for follow up study, cut parentheses/punct for
  colnames(allData) = gsub('\\)','',colnames(allData))
  colnames(allData) = gsub('\\-','_',colnames(allData))
}

allData$indiv = as.numeric(allData$individuals) # make sure R sees these as numbers
allData$ifI = as.numeric(allData$if_I)
```

```
allData$theseindiv = as.numeric(allData$these_individuals_can_work_together)

# handle different variables names in the follow up study
if (fname=='experiment.2.no.feelings.csv') {
  allData$thesepeople = as.numeric(allData$agree_These_people_are_like_minded)
  allData$feelthese = as.numeric(allData$strongly_agree_These_people_share_goals)
} else {
  allData$thesepeople = as.numeric(allData$these_people_are_like_minded)
  allData$feelthese = as.numeric(allData$feel_these_people_share_goals)
}
if (fname=='experiment.2.no.feelings.csv') {
  allData$togetherScore = (allData$indiv+allData$ifI+allData$theseindiv+allData$feelthese+allData$these
} else {
  allData$togetherScore = (allData$indiv+allData$ifI+allData$theseindiv+allData$feelthese+allData$these
}

# get rid of missing data!
# some qualtrics participants seem not to have responded
dim(allData)
```

[1] 864 16

```
allData = allData[!is.na(allData$togetherScore),]
dim(allData)
```

[1] 850 16

```
# before plotting by item, let's ensure we have them aligned
itemAlign = read.table('item_pairing.txt',sep='\t',stringsAsFactors=F)
allData$itemID = ''
for (i in 1:dim(allData)[1]) {
  #print(i)
  itemIx = grep(substr(as.character(allData[i,]$item),1,40),itemAlign$V2)
  if (length(itemIx)==0) {
    print(allData[i,]$item)
  }
  else {
    allData[i,]$itemID = itemAlign[itemIx,1][1]
  }
}

pander(paste('Time to test the combined effects of alignment X stimulus type'))
```

Time to test the combined effects of alignment X stimulus type

```
lmo.outcome = lmer(togetherScore~align*typ+(1+typ+align|id)+(1+align|itemID),data=allData)
coefs.outcome = data.frame(summary(lmo.outcome)$coefficient)
coefs.outcome$p = 2*(1-pnorm(abs(coefs.outcome$t.value)))
pander(coefs.outcome)
```

Table 8: Table continues below

	Estimate	Std..Error	t.value
(Intercept)	4.584	0.6482	7.072
alignnon-aligned	-0.03195	0.1875	-0.1704
typSyn	0.02513	0.9063	0.02773
typEmo	0.251	0.9049	0.2773
typLex	-0.3156	0.9052	-0.3486
alignnon-aligned:typSyn	-0.05252	0.2615	-0.2008
alignnon-aligned:typEmo	-1.932	0.2631	-7.341
alignnon-aligned:typLex	0.2947	0.2623	1.123

	p
(Intercept)	1.529e-12
alignnon-aligned	0.8647
typSyn	0.9779
typEmo	0.7815
typLex	0.7274
alignnon-aligned:typSyn	0.8408
alignnon-aligned:typEmo	2.112e-13
alignnon-aligned:typLex	0.2613

```
for (typ in types) {
  pander(paste('Time to test the effect of alignment for stimulus type:',typ))
  lmo.outcome = lmer(togetherScore~align+(1+align|id),data=allData[allData$typ==typ,])
  coefs.outcome = data.frame(summary(lmo.outcome)$coefficient)
  coefs.outcome$p = 2*(1-pnorm(abs(coefs.outcome$t.value)))
  pander(coefs.outcome)
}
```

Time to test the effect of alignment for stimulus type: Syn

	Estimate	Std..Error	t.value	p
(Intercept)	4.681	0.1143	40.94	0

alignnon-aligned -0.2186 0.1761 -1.241 0.2145

Time to test the effect of alignment for stimulus type: Emo

	Estimate	Std..Error	t.value	p
(Intercept)	4.824	0.1436	33.61	0

alignnon-aligned -1.926 0.1721 -11.19 0

Time to test the effect of alignment for stimulus type: Lex

	Estimate	Std..Error	t.value	p
(Intercept)	4.299	0.1764	24.37	0

alignnon-aligned 0.219 0.2507 0.8737 0.3823

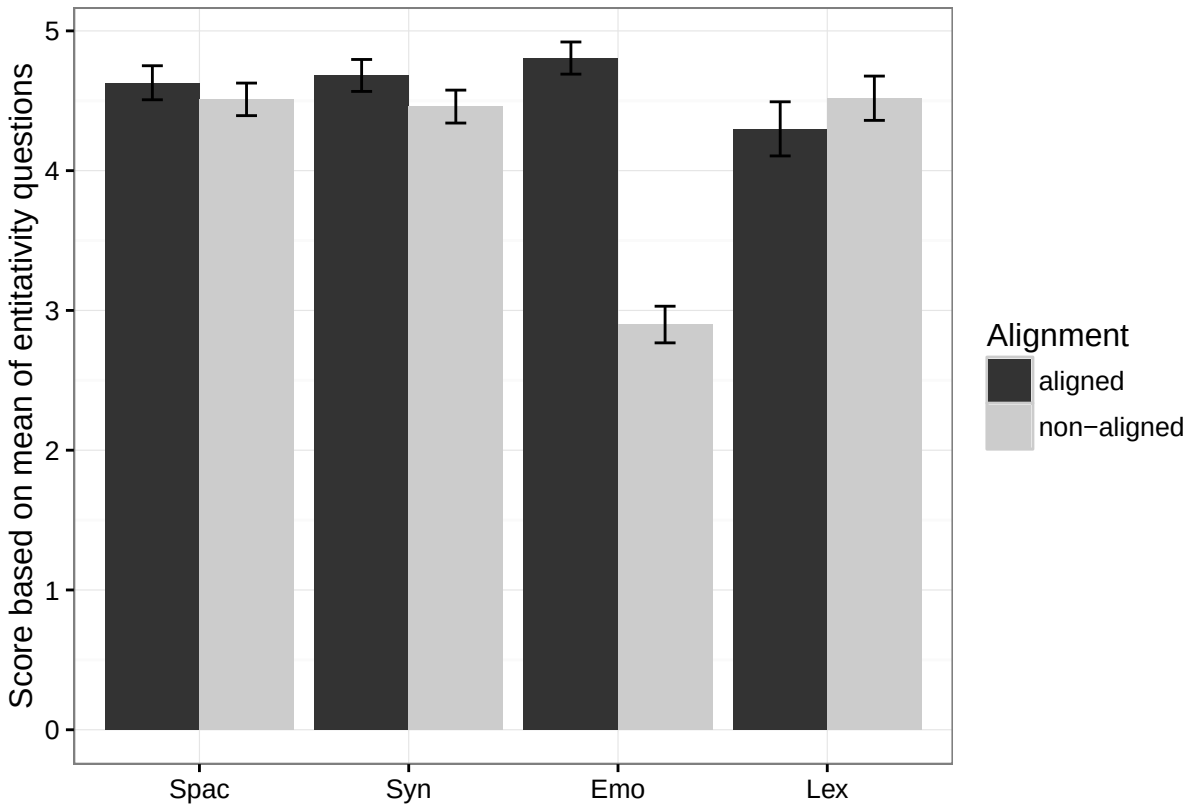
Time to test the effect of alignment for stimulus type: Spac

	Estimate	Std..Error	t.value	p
(Intercept)	4.599	0.1364	33.72	0

alignnon-aligned -0.05879 0.1371 -0.4288 0.6681

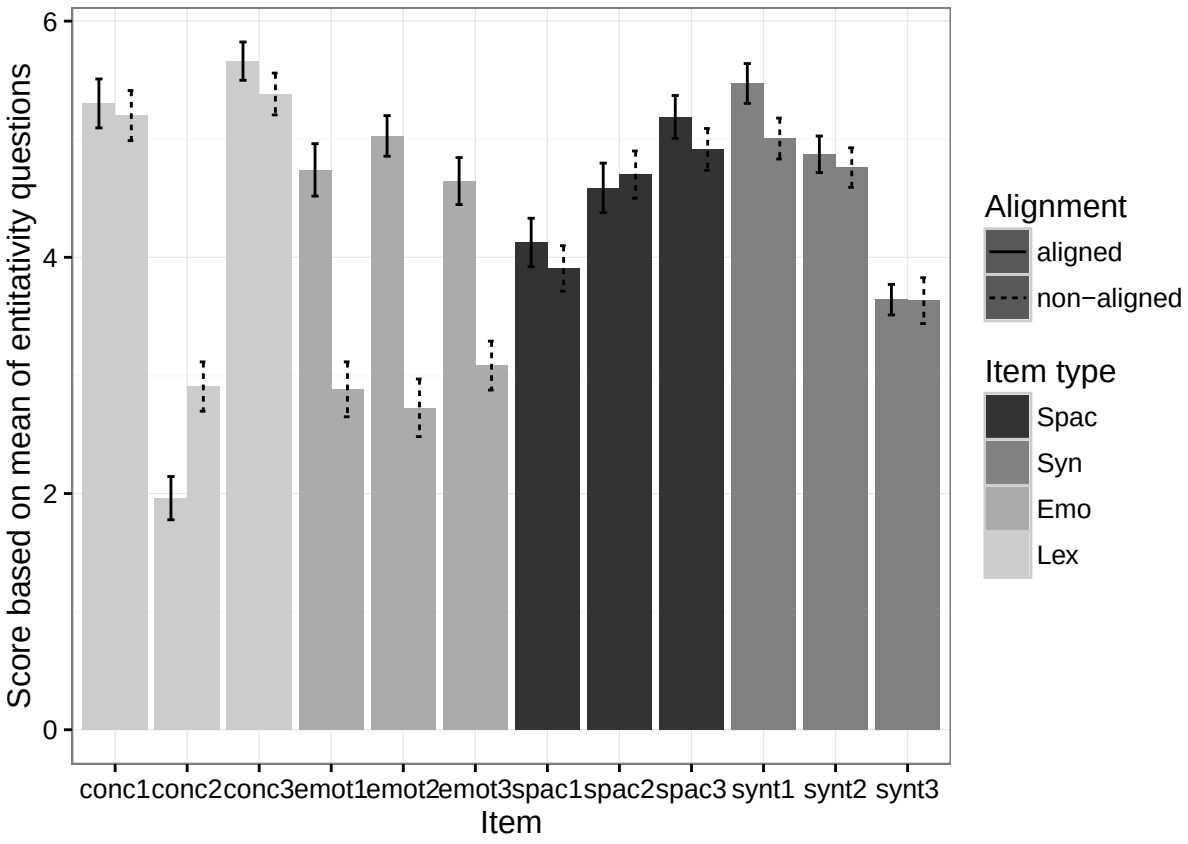
```
b = aggregate(allData$togetherScore,by=list(allData$align,allData$typ),FUN=mean) # get means
b_sd = aggregate(allData$togetherScore,by=list(allData$align,allData$typ),FUN=sd)
b_n = aggregate(allData$togetherScore*0+1,by=list(allData$align,allData$typ),FUN=sum)

ggplot(b, aes(fill=as.character(Group.1), y=x, x=Group.2))+scale_fill_grey()+ # togetherScore is a stan
  geom_bar(position="dodge", stat="identity")+
  geom_errorbar(aes(ymin=b$x-b_sd$x/sqrt(b_n$x),
                    width=.2,
                    ymax=b$x+b_sd$x/sqrt(b_n$x)),position=position_dodge(width=.9))+
  ylab('Score based on mean of entitativity questions')+
  xlab('')+labs(fill="Alignment") + theme_bw()
```




```
b = aggregate(allData$togetherScore,by=list(allData$align,allData$typ,allData$itemID),FUN=mean) # get m
b_sd = aggregate(allData$togetherScore,by=list(allData$align,allData$typ,allData$itemID),FUN=sd)
b_n = aggregate(allData$togetherScore*0+1,by=list(allData$align,allData$typ,allData$itemID),FUN=sum)

p <- ggplot(b, aes(fill=Group.2, y=x, x=Group.3,linetype=as.character(Group.1)))+scale_fill_grey() # to
p + geom_bar(position="dodge", stat="identity")+
  ylab('Score based on mean of entitativity questions')+
  xlab('Item')+
  geom_errorbar(aes(ymin=b$x-b_sd$x/sqrt(b_n$x),
                    width=.2,
                    ymax=b$x+b_sd$x/sqrt(b_n$x)),position=position_dodge(width=.9))+
  labs(fill = "Item type", linetype="Alignment") + theme(axis.text.x = element_blank()) + theme_bw()
```


Score based on mean of entitativity questions
6
4
2
0
conc1
conc2
conc3
emot1
emot2
emot3
spac1
spac2
spac3
synt1
synt2
synt3
Item
Alignment
aligned
non−aligned
Item type
Spac
Syn
Emo
Lex