Analyses for event segmentation thesis

Jenni Saaristo

18.8.2021

These are the final analyses reported in the thesis.

Main analyses

The analyses for the main research questions: responses for bounds versus perms, modulation by salience, and plotting these with the FIR models.

Calculate significance of bounds vs perms and plot results

Define which data to use for all results:

```
re_time <- '1s'
if (re_time == '3s') {is_3s <- TRUE} else {is_3s <- FALSE}
```

Statistical analyses

```
datapath <- paste(indir,'AVG_boundperms_',re_time,'_all.csv', sep="")
data <- read_csv(datapath, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
areas <- unique(data$area)

results <- data.frame(area=areas, beta=0, p=1, perm_mean=0, perm_sd=0)
for (a in areas) {
  perm_betas <- data$beta[data$perm != 0 & data$area == a]
  bound_beta <- data$beta[data$perm == 0 & data$area == a]
  p <- length(perm_betas[perm_betas > bound_beta])/length(perm_betas) # in fact we just want 1-tailed
  results$beta[results$area == a] <- bound_beta
  results$p[results$area == a] <- p
  results$perm_mean[results$area == a] <- mean(perm_betas)
  results$perm_sd[results$area == a] <- sd(perm_betas)
}
results$p_adj <- p.adjust(results$p, method='holm')
results[results$area == areas[49],]
```

```
          area     beta     p  perm_mean  perm_sd p_adj
49 Hippocampus 5.500318 0.115 0.03081762 4.384556     1
```

```
topareas <- results[results$p_adj < 0.05,]
topareas <- topareas[order(topareas$beta, decreasing = TRUE),]
topareas
```

```
                                     area     beta p   perm_mean  perm_sd
30       Cingulate Gyrus, posterior division 32.05827 0 -0.59157923 5.201382
31                          Precuneous Cortex 25.72171 0 -0.34749195 5.673152
47                     Supracalcarine Cortex 25.57428 0 -0.61049747 4.718722
24                     Intracalcarine Cortex 21.72983 0 -0.59643287 4.344622
32                             Cuneal Cortex 21.21921 0 -0.72466522 5.255012
35 Parahippocampal Gyrus, posterior division 19.07078 0 -0.09085367 4.567164
36                             Lingual Gyrus 16.97368 0 -0.53987416 4.273520
25                     Frontal Medial Cortex 16.08432 0  0.25332684 4.395494
   p_adj
30     0
31     0
47     0
24     0
32     0
35     0
36     0
25     0
```

```
for (i in 1:nrow(topareas)) {
  writeLines(paste(topareas$area[i], round(topareas$beta[i],3), round(topareas$p_adj[i],6), sep = '\t'))
}
```

```
Cingulate Gyrus, posterior division 32.058  0
Precuneous Cortex   25.722  0
Supracalcarine Cortex   25.574  0
Intracalcarine Cortex   21.73   0
Cuneal Cortex   21.219  0
Parahippocampal Gyrus, posterior division   19.071  0
Lingual Gyrus   16.974  0
Frontal Medial Cortex   16.084  0
```

```
write_csv(results, paste(outdir,'boundperm_results_',re_time,'.csv', sep=""))
sigs <- as.numeric(rownames(topareas))
```

Plot bounds vs perms

We'll only plot this for HC, the rest will make do with FIRs and stats

```
# Hippocampus
a = c(49)
asel <- areas[a]
bounds <- subset(data, perm == 0 & area %in% asel)
perms <- subset(data, perm != 0 & area %in% asel)
```

```
png(paste(figdir,'bounds_vs_perms_',re_time,'_HC.png',sep=""), width=1000, height=600)
ggplot(data=perms, aes(group=area)) +
  geom_histogram(aes(x=beta), binwidth = 2) +
  geom_vline(mapping=aes(xintercept=beta), data=bounds, color='red', size=2) +
  facet_wrap( ~ area,ncol=2) +
  labs(x='beta values', y='count (total 1000)', title='Boundaries vs. permutations') +
  theme_grey(base_size = 25)
dev.off()
```

```
pdf
  2
```

Modulation by salience

```
suppressMessages(library(lme4))
suppressMessages(library(lmerTest))
```

```
Warning: package 'lmerTest' was built under R version 4.0.2
```

```
datapath <- paste(indir,'betas_series_long_',re_time,'.csv', sep="")
data <- read_csv(datapath, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
boundpath <- paste('/Users/jenska/code/python/eventcode/1_create_boundaries/out/boundaries_f',re_time,'
bounds <- read_csv(boundpath, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
areas <- unique(data$area)

# Salience for bounds (bins: 5-6, 7-9, 10-17)
bounds$salience <- 0
bounds$salience[bounds$nobs >= 7] <- 1
bounds$salience[bounds$nobs >= 10] <- 2

# Clean some unnecessary columns
bounds <- subset(bounds, select=c(id,nobs,salience,meanvol,voldiff))
data <- subset(data, select=-c(X1))
data <- dplyr::rename(data, id=bound)

# Also remove the post-hoc bounds from data
data <- data[data$id != 999,]

# Join the data frames
data <- inner_join(data,bounds, by="id")

# Run glm on HC, salience
```

```
datasel <- data[data$area == areas[49],]
m1 <- lmer(beta ~ salience + meanvol + voldiff + (1 | subj) + (1 | id), datasel)
anova(m1)
```

```
Type III Analysis of Variance Table with Satterthwaite's method
         Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
salience  78069   78069     1 63.999  2.0636 0.1557
meanvol   12356   12356     1 63.999  0.3266 0.5697
voldiff   18804   18804     1 63.999  0.4970 0.4834
```

```
# Run glm on HC, nObs
datasel <- data[data$area == areas[49],]
m2 <- lmer(beta ~ nobs + meanvol + voldiff + (1 | subj) + (1 | id), datasel)
anova(m2)
```

```
Type III Analysis of Variance Table with Satterthwaite's method
        Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
nobs     91538   91538     1    64  2.4196 0.1248
meanvol   2541    2541     1    64  0.0672 0.7964
voldiff  18754   18754     1    64  0.4957 0.4839
```

```
# Run for all ROIs
results <- data.frame(area=areas, f_sal=0, p_sal=1, f_obs=0, p_obs=1)

for (i in c(1:49)) {
  datasel <- data[data$area == areas[i],]
  anv1 <- anova(lmer(beta ~ salience +  meanvol + voldiff + (1 | subj) + (1 | id), datasel))
  anv2 <- anova(lmer(beta ~ nobs +  meanvol + voldiff + (1 | subj) + (1 | id), datasel))
  results$f_sal[i] <- anv1$`F value`[1]
  results$p_sal[i] <- anv1$`Pr(>F)`[1]
  results$f_obs[i] <- anv2$`F value`[1]
  results$p_obs[i] <- anv2$`Pr(>F)`[1]
}
```

```
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
```

```
results$p_sal_adj <- p.adjust(results$p_sal, method='holm')
results$p_obs_adj <- p.adjust(results$p_obs, method='holm')

# Select results
topareas <- results[results$p_sal_adj < 0.05 | results$p_obs_adj < 0.05,]
topareas <- topareas[order(topareas$f_sal, decreasing = T),]
topareas
```

```
           area    f_sal       p_sal    f_obs        p_obs p_sal_adj
36 Lingual Gyrus 9.328812 0.003285878 12.99564 0.0006114604  0.161008
    p_obs_adj
36 0.02996156
```

```
results[results$area == areas[49],]
```

```
          area    f_sal     p_sal    f_obs     p_obs p_sal_adj p_obs_adj
49 Hippocampus 2.063564 0.1557276 2.419582 0.1247585         1         1
```

```
for (i in 1:nrow(topareas)) {
  writeLines(paste(topareas$area[i], round(topareas$f_sal[i],3), round(topareas$p_sal[i],3), round(topa
}
```

```
Lingual Gyrus   9.329   0.003   0.161   12.996  0.001   0.03
```

Plotting FIRs

We'll actually only use the ones from 3s, with an added line for the 1s bound. In addition to HC we'll plot all areas that are significant in either condition, because the reader should have all that info available – it sucks when you don't have it.

```
datapath <- paste(indir,'AVG_fir_',re_time,'_all.csv', sep="")
data <- read_csv(datapath, col_types = cols())
data$cond <- as.factor(data$cond)
data$cond <- ordered(data$cond, levels=c('high','mid','low','all'))

data <- filter(data, grepl('delay', regressor))
data$delay <- as.numeric(gsub(data$regressor, pattern="[^0-9]", replacement=""))
data$delay <- data$delay - 5

areas <- unique(data$area)
conds <- c('high','mid','low')
#conds <- c('all')

# Plot HC
asel <- areas[49]
plotdata <- subset(data, area %in% asel & cond %in% conds)
png(paste(figdir,'bounds_FIRs_',re_time,'_HC.png', sep=""), width=1000, height=600)
pd <- position_dodge(0.2)
ggplot(plotdata, aes(x=delay, y=beta, color=cond)) +
  geom_hline(yintercept = 0, color='grey50', size=1) +
  geom_vline(xintercept = 0, color='blue3', size=2) +  # delay = 0
  geom_vline(xintercept = 2, color='green3', size=2) +  # delay = 2
  geom_point(position=pd, size=8) +
  geom_line(position=pd, size=3) +
  geom_errorbar(aes(ymin=beta-se, ymax=beta+se), width=0.5, size=1, position=pd) +
  scale_color_grey(name='salience') +
  scale_x_continuous(breaks=seq(-5,10)) +
  labs(title=paste('Modulation by salience')) +
  xlab('delay (sec)') +
  facet_wrap(~ area, ncol=1) +
  theme_bw(base_size = 24) +
  theme(panel.grid=element_blank())
dev.off()
```

```
# Plot others
plotlist = list()
anums = sigs
for (i in 1:length(anums)) {
  asel <- areas[anums[i]]
  plotdata <- subset(data, area %in% asel & cond %in% conds)
  pd <- position_dodge(0.2)
  p <- ggplot(plotdata, aes(x=delay, y=beta, color=cond)) +
    geom_hline(yintercept = 0, color='grey50', size=1) +
    geom_vline(xintercept = 0, color='blue3', size=2) +  # delay = 0
    geom_vline(xintercept = 2, color='green3', size=2) +  # delay = 2
    geom_point(position=pd, size=6) +
    geom_line(position=pd, size=2) +
    geom_errorbar(aes(ymin=beta-se, ymax=beta+se), width=0.5, size=1, position=pd) +
    scale_color_grey(name='salience') +
    scale_x_continuous(breaks=seq(-5,10)) +
    scale_y_continuous(limits=c(-7,18), breaks=seq(-5,15,5)) +
    labs(title=paste('Modulation by salience')) +
    xlab('delay (sec)') +
    facet_wrap(~ area, ncol=1) +
    theme_bw(base_size = 24) +
    theme(panel.grid=element_blank())
  plotlist[[i]] <- p
}

for (i in 1:length(anums)) {
  png(paste(figdir,'bounds_FIRs_',re_time, '_', anums[[i]], '.png', sep=""), width=800, height=500)
  print(plotlist[[i]])
  dev.off()
}
```

Secondary analyses: familiarity effects

Nothing is even close to significant at 3s, even less at 1s – does it make sense to even plot any of these? Well maybe HC has theoretical interest, so we should mention it.

```
library(effsize)
```

```
Warning: package 'effsize' was built under R version 4.0.2
```

```
datapath <- paste(indir,'betas_boundperms_long_',re_time,'.csv', sep="")
data <- read_csv(datapath, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
areas <- unique(data$area)

# Get subj info and merge
subjdata <- read_csv('code/python/eventcode/subj_info.csv', col_types = cols())
data <- merge(data, subjdata)
```

```
results <- data.frame(area = areas, mean_lis1=0, mean_lis2=0, t=0, p=1, d=0)

for (a in areas) {
  lis1_betas <- data$beta[data$listening == 'first' & data$area == a]
  lis2_betas <- data$beta[data$listening == 'second' & data$area == a]
  stat <- t.test(lis1_betas,lis2_betas)
  eff <- cohen.d(lis1_betas, lis2_betas)
  results$mean_lis1[results$area == a] <- stat$estimate[1]
  results$mean_lis2[results$area == a] <- stat$estimate[2]
  results$t[results$area == a] <- stat$statistic
  results$p[results$area == a] <- stat$p.value
  results$d[results$area == a] <- eff$estimate
}
results$p_adj <- p.adjust(results$p, method='holm')

rm(lis1_betas,lis2_betas,stat,eff)

results[results$area == areas[49],]
```

```
          area mean_lis1 mean_lis2          t         p           d p_adj
49 Hippocampus  4.518362  6.564103 -0.3168905 0.7528393 -0.09058043     1
```

```
topareas <- results[results$d > 0.3,]
topareas <- topareas[order(topareas$p, decreasing = FALSE),]
topareas
```

```
                area mean_lis1 mean_lis2        t          p         d p_adj
31 Precuneous Cortex  33.55874  17.23161 2.213658 0.03164716 0.6256473     1
32     Cuneal Cortex  25.85798  16.19388 1.166743 0.24925551 0.3315474     1
```

```
for (i in 1:nrow(topareas)) {
  writeLines(paste(topareas$area[i], round(topareas$mean_lis1[i],3), round(topareas$mean_lis2[i],3),  r
}
```

```
Precuneous Cortex   33.559  17.232  0.626   0.032   1
Cuneal Cortex   25.858  16.194  0.332   0.249   1
```

Plot FIRs by grp?

Yes this should be done, though only for HC

```
datapath1 <- paste(indir,'AVG_fir_',re_time,'_1st.csv', sep="")
data1 <- read_csv(datapath1, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
datapath2 <- paste(indir,'AVG_fir_',re_time,'_2nd.csv', sep="")
data2 <- read_csv(datapath2, col_types = cols())
```

```
Warning: Missing column names filled in: 'X1' [1]
```

```
data1 <- subset(data1, cond=='all')
data2 <- subset(data2, cond=='all')
data1$lis <- '1st'
data2$lis <- '2nd'
data <- rbind(data1, data2)
rm(datapath1,datapath2,data1,data2)

data <- filter(data, grepl('delay', regressor))
data$delay <- as.numeric(gsub(data$regressor, pattern="[^0-9]", replacement=""))
data$delay <- data$delay - 5

areas <- unique(data$area)

# Plot HC
asel <- areas[49]
plotdata <- subset(data, area %in% asel)
png(paste(figdir,'fam_FIRs_',re_time,'_HC.png', sep=""), width=1000, height=600)
pd <- position_dodge(0.2)
ggplot(plotdata, aes(x=delay, y=beta, color=lis)) +
  geom_hline(yintercept = 0, color='grey50', size=1) +
  geom_vline(xintercept = 0, color='blue3', size=2) +  # delay = 0
  geom_vline(xintercept = 2, color='green3', size=2) +  # delay = 2
  geom_point(position=pd, size=8) +
  geom_line(position=pd, size=3) +
  geom_errorbar(aes(ymin=beta-se, ymax=beta+se), width=0.5, size=1, position=pd) +
  scale_color_grey(name='Listening') +
  scale_x_continuous(breaks=seq(-5,10)) +
  labs(title=paste('Modulation by familiarity')) +
  xlab('delay (sec)') +
  #facet_wrap(~ area, ncol=1) +
  theme_bw(base_size = 24) +
  theme(panel.grid=element_blank())
dev.off()
```

```
pdf
  2
```

Test bounds against non-bounds (audio gaps)

```
datapath <- paste(indir,'AVG_audioperms_',re_time,'_all.csv', sep="")
data <- read_csv(datapath, col_types = cols())
areas <- unique(data$area)

results <- data.frame(area=areas, beta=0, p=1, perm_mean=0, perm_sd=0)
for (a in areas) {
  perm_betas <- data$beta[data$perm != 0 & data$area == a]
  bound_beta <- data$beta[data$perm == 0 & data$area == a]
  p <- length(perm_betas[perm_betas > bound_beta])/length(perm_betas) # in fact we just want 1-tailed
  results$beta[results$area == a] <- bound_beta
```

```
  results$p[results$area == a] <- p
  results$perm_mean[results$area == a] <- mean(perm_betas)
  results$perm_sd[results$area == a] <- sd(perm_betas)
}
results$p_adj <- p.adjust(results$p, method='holm')
results[results$area == areas[49],]

topareas <- results[results$p_adj < 0.05,]
topareas <- topareas[order(topareas$beta, decreasing = TRUE),]
topareas

for (i in 1:nrow(topareas)) {
  writeLines(paste(topareas$area[i], round(topareas$beta[i],3), round(topareas$p_adj[i],6), sep = '\t'))
}

write_csv(results, paste(outdir,'audioperm_results_',re_time,'.csv', sep=""))
sigs <- as.numeric(rownames(topareas))
```

Plot

Not sure if it makes sense to plot these... for HC maybe? It would be nice to be able to plot FIRs contrasted with a non-bound, but I guess it doesn't really make sense...

```
# Plot HC
a = c(49)
asel <- areas[a]
bounds <- subset(data, perm == 0 & area %in% asel)
perms <- subset(data, perm != 0 & area %in% asel)
png(paste(figdir,'bounds_vs_audio_',re_time,'_HC.png',sep=""), width=1000, height=600)
ggplot(data=perms, aes(group=area)) +
  geom_histogram(aes(x=beta), binwidth = 2) +
  geom_vline(mapping=aes(xintercept=beta), data=bounds, color='red', size=2) +
  facet_wrap( ~ area,ncol=2) +
  labs(x='beta values', y='count (total 1000)', title='Boundaries vs. audiogaps') +
  theme_grey(base_size = 25)
dev.off()

# Plot others
plotlist = list()
for (i in 1:length(sigs)) {
  asel <- areas[sigs[i]]
  bounds <- subset(data, perm == 0 & area %in% asel)
  perms <- subset(data, perm != 0 & area %in% asel)
  p <- ggplot(data=perms, aes(group=area)) +
  geom_histogram(aes(x=beta), binwidth = 2) +
  geom_vline(mapping=aes(xintercept=beta), data=bounds, color='red', size=2) +
  facet_wrap( ~ area,ncol=2) +
  labs(x='beta values', y='count (total 1000)', title='Boundaries vs. audiogaps') +
  theme_grey(base_size = 25)
  plotlist[[i]] <- p
}
for (i in 1:length(sigs)) {
  png(paste(figdir,'bounds_vs_audio_',re_time, '_', sigs[[i]], '.png', sep=""), width=800, height=500)
```

```
  print(plotlist[[i]])
  dev.off()
}
```