

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 <- '3s'
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 13.26718 0.002 -0.008989224 4.532317 0.076

```

```

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 33.16503 0 -0.49977993 5.279699
47 Supracalcarine Cortex 31.98627 0 -0.40716621 4.712508
24 Intracalcarine Cortex 29.30523 0 -0.40588941 4.398255
32 Cuneal Cortex 27.72143 0 -0.36579824 5.289758
29 Cingulate Gyrus, anterior division 26.18845 0 -0.49351962 4.547156
31 Precuneous Cortex 24.28142 0 -0.08424546 5.692329
36 Lingual Gyrus 24.26136 0 -0.32052527 4.311826
2 Insular Cortex 21.53005 0 0.18433728 4.460171
41 Frontal Operculum Cortex 19.22037 0 -0.80496618 4.639239
35 Parahippocampal Gyrus, posterior division 17.98500 0 -0.03943910 4.632752
25 Frontal Medial Cortex 17.93846 0 0.02131449 4.398071
      p_adj
30 0
47 0
24 0
32 0
29 0
31 0
36 0
2 0
41 0
35 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 33.165 0
Supracalcarine Cortex 31.986 0
Intracalcarine Cortex 29.305 0
Cuneal Cortex 27.721 0
Cingulate Gyrus, anterior division 26.188 0
Precuneous Cortex 24.281 0
Lingual Gyrus 24.261 0
Insular Cortex 21.53 0
Frontal Operculum Cortex 19.22 0
Parahippocampal Gyrus, posterior division 17.985 0
Frontal Medial Cortex 17.938 0

```

```

write_csv(results, paste(outdir, 'boundperm_results_', re_time, '.csv', sep = ""))
sigs <- as.numeric(rownames(topareas))
sigs <- append(sigs, 49)

```

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=550, height=400)
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=paste('Latency -', re_time, sep="")) +
  scale_x_continuous(limits=c(-17,17), breaks=seq(-15,15,5)) +
  theme_grey(base_size = 25)
dev.off()
```

pdf
2

Plot FIRs

We'll 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. For this first part we plot the grand average over all saliences as that's more clearly interpretable in conjunction with the cortical plots.

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

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

```
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 - 8 # adjust latencies so that behav boundary is 0
areas <- unique(data$area)

# Plot HC large
conds <- c('high','mid','low')
asel <- areas[49]
plotdata <- subset(data, area %in% asel & cond %in% conds)
png(paste(figdir, 'bounds_FIRs_', re_time, '_HC.png', sep=""), width=1100, height=700)
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 = -3, color='blue3', size=2) + # latency = -3
```

```

geom_vline(xintercept = -1, color='green3', size=2) + # latency = -1
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(-8,7)) +
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()

```

pdf
2

```

# Plot others small, grand average
conds <- c('all')
asel <- areas[sigs]
plotdata <- subset(data, area %in% asel & cond %in% conds)
pd <- position_dodge(0.2)
png(paste(figdir, 'bounds_FIRs_', re_time, '_avg_cortical.png', sep=""), width=1400, height=700)
ggplot(plotdata, aes(x=delay, y=beta)) +
  geom_hline(yintercept = 0, color='grey50', size=1) +
  geom_vline(xintercept = -3, color='blue3', size=2) + # latency = -3
  geom_vline(xintercept = -1, color='green3', size=2) + # latency = -1
  geom_point(position=pd, size=3) +
  geom_line(position=pd, size=1) +
  geom_errorbar(aes(ymin=beta-se, ymax=beta+se), width=0.3, size=0.7, position=pd) +
  #scale_color_grey(name='salience') +
  scale_x_continuous(breaks=seq(-8,7)) +
  scale_y_continuous(limits=c(-7,12), breaks=seq(-5,10,5)) +
  labs(title=paste('FIR responses')) +
  xlab('delay (sec)') +
  facet_wrap(~ area, ncol=4, scales='free') +
  theme_bw(base_size = 20) +
  theme(panel.grid=element_blank())
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$noobs >= 7] <- 1
bounds$salience[bounds$noobs >= 10] <- 2

# Clean some unnecessary columns
bounds <- subset(bounds, select=c(id,noobs,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")

# Check correlations
cor.test(bounds$salience, bounds$meanvol)
```

Pearson's product-moment correlation

```
data: bounds$salience and bounds$meanvol
t = -2.4113, df = 77, p-value = 0.01828
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.45917825 -0.04658916
sample estimates:
      cor
-0.26497
```

```
cor.test(bounds$salience, bounds$voldiff)
```

Pearson's product-moment correlation

```
data: bounds$salience and bounds$voldiff
t = 0.86378, df = 77, p-value = 0.3904
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1258731 0.3123093
```

```
sample estimates:
      cor
0.0979638
```

```
cor.test(bounds$noobs, bounds$meanvol)
```

Pearson's product-moment correlation

```
data: bounds$noobs and bounds$meanvol
t = -2.4307, df = 77, p-value = 0.0174
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.46085634 -0.04871285
sample estimates:
      cor
-0.266948
```

```
cor.test(bounds$noobs, bounds$voldiff)
```

Pearson's product-moment correlation

```
data: bounds$noobs and bounds$voldiff
t = 1.2793, df = 77, p-value = 0.2046
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.07937505 0.35408284
sample estimates:
      cor
0.1442673
```

```
# Run glm on HC, salience
dataset1 <- data[data$area == areas[49],]
m1 <- lmer(beta ~ salience + meanvol + voldiff + (1 | subj) + (1 | id), dataset1)
anova(m1)
```

Type III Analysis of Variance Table with Satterthwaite's method

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
salience	189669	189669	1	62	4.9862	0.02917 *
meanvol	7133	7133	1	62	0.1875	0.66650
voldiff	2048	2048	1	62	0.0538	0.81726

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
m11 <- lmer(beta ~ salience + (1 | subj) + (1 | id), dataset1)
anova(m11)
```

Type III Analysis of Variance Table with Satterthwaite's method

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
salience	231126	231126	1	64	6.0761	0.0164 *

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Run glm on HC, nObs
dataset1 <- data[data$area == areas[49],]
m2 <- lmer(beta ~ nobis + meanvol + voldiff + (1 | subj) + (1 | id), dataset1)
anova(m2)
```

Type III Analysis of Variance Table with Satterthwaite's method

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
nobis	204721	204721	1	62	5.3819	0.02365 *
meanvol	3056	3056	1	62	0.0803	0.77777
voldiff	1018	1018	1	62	0.0268	0.87056

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
m21 <- lmer(beta ~ nobis + (1 | subj) + (1 | id), dataset1)
anova(m21)
```

Type III Analysis of Variance Table with Satterthwaite's method

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
nobis	252186	252186	1	63.999	6.6297	0.01235 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
# 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)) {
  dataset1 <- data[data$area == areas[i],]
  anv1 <- anova(lmer(beta ~ salience + meanvol + voldiff + (1 | subj) + (1 | id), dataset1))
  anv2 <- anova(lmer(beta ~ nobis + meanvol + voldiff + (1 | subj) + (1 | id), dataset1))
  results$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
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
```

```

boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
boundary (singular) fit: see ?isSingular
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_obs, decreasing = T),]
topareas

```

	area	f_sal	p_sal	f_obs
30	Cingulate Gyrus, posterior division	16.16975	0.0001596161	25.211842
29	Cingulate Gyrus, anterior division	16.56074	0.0001356087	21.466149
36	Lingual Gyrus	15.28863	0.0002313237	19.312237
24	Intracalcarine Cortex	16.20530	0.0001572619	17.745262
47	Supracalcarine Cortex	15.42152	0.0002186605	16.438793
41	Frontal Operculum Cortex	12.39913	0.0008119068	13.314467
27	Subcallosal Cortex	12.26186	0.0004684480	6.055388
	p_obs	p_sal_adj	p_obs_adj	
30	4.617616e-06	0.007548569	0.0002262632	
29	1.898349e-05	0.006644829	0.0009112077	
36	4.424307e-05	0.010409565	0.0020794244	
24	8.325590e-05	0.007548569	0.0038297716	
47	1.426661e-04	0.010058383	0.0064199744	
41	5.417898e-04	0.034911991	0.0238387500	
27	1.666250e-02	0.020611711	0.4998750318	

```

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

```

	area	f_sal	p_sal	f_obs	p_obs	p_sal_adj	p_obs_adj
49	Hippocampus	4.986202	0.02916968	5.381894	0.02365382	0.9625996	0.6445788

```

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
})

```

Cingulate Gyrus, posterior division	16.17	0	0.008	25.212	0	0
Cingulate Gyrus, anterior division	16.561	0	0.007	21.466	0	0.001
Lingual Gyrus	15.289	0	0.01	19.312	0	0.002
Intracalcarine Cortex	16.205	0	0.008	17.745	0	0.004


```
sigs <- as.numeric(rownames(topareas))
sigs <- append(sigs, 49)
```

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

for (i in c(1:49)) {
  dataset1 <- data[data$area == areas[i],]
  anv1 <- anova(lmer(beta ~ salience + (1 | subj) + (1 | id), dataset1))
  anv2 <- anova(lmer(beta ~ nobis + (1 | subj) + (1 | id), dataset1))
  results_nocon$f_sal[i] <- anv1$`F value`[1]
  results_nocon$p_sal[i] <- anv1$`Pr(>F)`[1]
  results_nocon$f_obs[i] <- anv2$`F value`[1]
  results_nocon$p_obs[i] <- anv2$`Pr(>F)`[1]
}
```

9

```

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

# Select results
topareas_nocon <- results_nocon[results_nocon$p_sal_adj < 0.05 | results_nocon$p_obs_adj < 0.05,]
topareas_nocon <- topareas_nocon[order(topareas_nocon$f_obs, decreasing = T),]
topareas_nocon

```

	area	f_sal	p_sal	f_obs
29	Cingulate Gyrus, anterior division	20.251710	2.931893e-05	25.857183
30	Cingulate Gyrus, posterior division	14.236319	3.548487e-04	21.134587
36	Lingual Gyrus	16.518809	1.341377e-04	20.238947
24	Intracalcarine Cortex	16.956117	1.117642e-04	18.092706
41	Frontal Operculum Cortex	15.931704	1.717041e-04	17.332846
47	Supracalcarine Cortex	15.776815	1.833296e-04	16.378016
28	Paracingulate Gyrus	8.235508	5.560652e-03	12.671881
27	Subcallosal Cortex	12.302040	4.585032e-04	6.225487

	p_obs	p_sal_adj	p_obs_adj
29	3.441027e-06	0.001436627	0.0001686103
30	2.070082e-05	0.015613342	0.0009936394
36	2.946771e-05	0.006304473	0.0013849824
24	6.994200e-05	0.005364680	0.0032173318
41	9.559825e-05	0.007898389	0.0043019214
47	1.422910e-04	0.008249830	0.0062608051
28	7.061029e-04	0.205744115	0.0303624239
27	1.264096e-02	0.019715637	0.3829092245

```

results_nocon[results_nocon$area == areas[49],]

```

	area	f_sal	p_sal	f_obs	p_obs	p_sal_adj	p_obs_adj
49	Hippocampus	6.076052	0.0163965	6.629705	0.01235191	0.5738776	0.3829092

```

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

```

Cingulate Gyrus, anterior division	20.252	0	0.001	25.857	0	0
Cingulate Gyrus, posterior division	14.236	0	0.016	21.135	0	0.001
Lingual Gyrus	16.519	0	0.006	20.239	0	0.001
Intracalcarine Cortex	16.956	0	0.005	18.093	0	0.003
Frontal Operculum Cortex	15.932	0	0.008	17.333	0	0.004
Supracalcarine Cortex	15.777	0	0.008	16.378	0	0.006
Paracingulate Gyrus	8.236	0.006	0.206	12.672	0.001	0.03
Subcallosal Cortex	12.302	0	0.02	6.225	0.013	0.383

```

sigs_nocon <- as.numeric(rownames(topareas_nocon))
sigs_nocon <- append(sigs, 49)

```

Plot FIRs

With salience

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

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

```
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 - 8 # adjust latencies so that behav boundary is 0
areas <- unique(data$area)
conds <- c('high','mid','low')

# Plot all with salience
asel <- areas[sigs]
plotdata <- subset(data, area %in% asel & cond %in% conds)
pd <- position_dodge(0.2)
png(paste(figdir,'bounds_FIRs_',re_time, '_sal_cortical.png', sep=""), width=1400, height=500)
ggplot(plotdata, aes(x=delay, y=beta, color=cond)) +
  geom_hline(yintercept = 0, color='grey50', size=1) +
  geom_vline(xintercept = -3, color='blue3', size=2) + # latency = -3
  #geom_vline(xintercept = -1, color='green3', size=2) + # latency = -1
  geom_point(position=pd, size=3) +
  geom_line(position=pd, size=1) +
  geom_errorbar(aes(ymin=beta-se, ymax=beta+se), width=0.3, size=0.7, position=pd) +
  scale_color_grey(name='salience') +
  scale_x_continuous(breaks=seq(-8,7)) +
  scale_y_continuous(limits=c(-7,18), breaks=seq(-5,15,5)) +
  labs(title=paste('FIR responses')) +
  xlab('delay (sec)') +
  facet_wrap(~ area, ncol=4, scales='free') +
  theme_bw(base_size = 20) +
  theme(panel.grid=element_blank())
dev.off()
```

pdf
2

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	14.36218	12.08093	0.2906752	0.7725962	0.08264152	1

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

	area	mean_lis1	mean_lis2	t
11	Middle Temporal Gyrus, anterior division	-0.8446181	-15.926837	1.827171
43	Parietal Operculum Cortex	9.4054551	0.949726	1.350540
15	Inferior Temporal Gyrus, posterior division	2.4565525	-4.607025	1.217448
42	Central Opercular Cortex	6.5383614	-2.131595	1.191225
29	Cingulate Gyrus, anterior division	29.6676144	22.419352	1.141616
32	Cuneal Cortex	31.4039873	23.731998	1.127764
30	Cingulate Gyrus, posterior division	37.3968033	28.580617	1.067881
39	Temporal Occipital Fusiform Cortex	12.9878692	4.736630	1.065569

	p	d	p_adj
11	0.07389533	0.5156857	1
43	0.18321916	0.3823119	1
15	0.22969264	0.3466994	1
42	0.23947423	0.3373737	1
29	0.25935049	0.3235867	1

```
32 0.26502910 0.3185144 1
30 0.29097949 0.3026303 1
39 0.29195438 0.3002527 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
}
```

```
Middle Temporal Gyrus, anterior division -0.845 -15.927 0.516 0.074 1
Parietal Operculum Cortex 9.405 0.95 0.382 0.183 1
Inferior Temporal Gyrus, posterior division 2.457 -4.607 0.347 0.23 1
Central Opercular Cortex 6.538 -2.132 0.337 0.239 1
Cingulate Gyrus, anterior division 29.668 22.419 0.324 0.259 1
Cuneal Cortex 31.404 23.732 0.319 0.265 1
Cingulate Gyrus, posterior division 37.397 28.581 0.303 0.291 1
Temporal Occipital Fusiform Cortex 12.988 4.737 0.3 0.292 1
```

Plot swarms

I.e. the values that went into the t-test

```
library(ggbeeswarm)
```

Warning: package 'ggbeeswarm' was built under R version 4.0.2

```
asel <- areas[49]
plotdata <- subset(data, area %in% asel)
png(paste(figdir, 'fam_swarm_', re_time, '_HC.png', sep=""), width=600, height=500)
ggplot(plotdata, aes(x=listening, y=beta)) +
  geom_violin(color='grey40', fill='bisque2', size=1, draw_quantiles=0.5) +
  geom_quasirandom(dodge.width=1, size=4) +
  labs(title=paste('Effect of familiarity on HC response')) +
  xlab('Listening') +
  theme_bw(base_size = 24) +
  theme(panel.grid=element_blank())
dev.off()
```

```
pdf
2
```

Plot FIRs by grp

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_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())
```

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 10.69904 0.038  1.866707 5.190742 0.988

```

```

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

```

```

      area      beta p perm_mean perm_sd p_adj
29 Cingulate Gyrus, anterior division 35.86869 0 4.766185 4.902877 0
2      Insular Cortex 32.23890 0 2.464093 4.757457 0
32      Cuneal Cortex 27.61241 0 2.679093 5.498810 0
47      Supracalcarine Cortex 27.29055 0 4.877853 4.849920 0
24      Intracalcarine Cortex 25.83786 0 3.868627 4.599572 0
41      Frontal Operculum Cortex 23.91681 0 6.535029 4.799281 0
36      Lingual Gyrus 23.42647 0 1.656477 4.634595 0
40      Occipital Fusiform Gyrus 22.82935 0 4.721020 5.492654 0
30 Cingulate Gyrus, posterior division 22.81602 0 4.309939 5.734724 0
48      Occipital Pole 17.49597 0 1.498964 4.420203 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, anterior division 35.869 0
Insular Cortex 32.239 0
Cuneal Cortex 27.612 0
Supracalcarine Cortex 27.291 0
Intracalcarine Cortex 25.838 0
Frontal Operculum Cortex 23.917 0
Lingual Gyrus 23.426 0
Occipital Fusiform Gyrus 22.829 0
Cingulate Gyrus, posterior division 22.816 0
Occipital Pole 17.496 0

```

```

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... well maybe yes, there's room in the fig

```
# 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=500, height=400)
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()
```

pdf
2

```
# Plot others
asel <- areas[sigs]
bounds <- subset(data, perm == 0 & area %in% asel)
perms <- subset(data, perm != 0 & area %in% asel)
png(paste(figdir, 'bounds_vs_audio_', re_time, '_cortical.png', sep=""), width=1300, height=400)
ggplot(data=perms, aes(group=area)) +
  geom_histogram(aes(x=beta), binwidth = 2) +
  geom_vline(mapping=aes(xintercept=beta), data=bounds, color='red', size=1.5) +
  facet_wrap( ~ area, nrow=2) +
  labs(x='beta values', y='count (total 1000)') +
  theme_grey(base_size = 19)
dev.off()
```

pdf
2

```
rm(data, bounds, perms, results, topareas, areas, a, asel, bound_beta, perm_betas, i, datapath, plotlist, p, sigs)
```

Warning in rm(data, bounds, perms, results, topareas, areas, a, asel,
bound_beta, : object 'plotlist' not found