

# Analyze IRESP outputs from deconvolution of fMRI task

Code ▾

Load R libraries

This is an R Markdown (<http://rmarkdown.rstudio.com>) Notebook. When you execute code within the notebook, the results appear beneath the code.

Set up default plotting parameters (may need to be modified for each plot).

Hide

```
text.size = 10
se.width = .075

#ggplot theme
ggplot.theme <- theme(
  panel.grid.minor = element_blank(),
  panel.grid.major = element_blank(),
  panel.background= element_blank(),
  panel.border = element_blank(),
  panel.spacing = unit(0.2,'in'),
  axis.line = element_line(size = .5, colour = "black"),
  axis.title = element_text(size = text.size),
  axis.text = element_text(size = text.size),
  strip.background = element_rect(colour = 'black'),
  strip.text = element_text(size = text.size),
  legend.direction='horizontal',
  legend.spacing=unit(.02,'in'),
  legend.position = 'top',#c(.5,.95),
  legend.background = element_blank(),
  legend.margin = margin(rep(.1,4)),
  legend.key = element_rect(fill=NA),
  legend.title = element_blank(),
  legend.text = element_text(size = text.size),
  legend.text.align = 1
)
```

Let's load data!

Hide

```

#let the user enter which ROI they want to look at on the command line
ROI<-readline(prompt="Enter LC, hpc, Nacc, VTA bf123, or bf4:  ")
LC

#load the first data file
iresp_ROI=as.data.frame(read.table(paste("/home/er482/CAC/MRI_data/ABE_analysis/iresp_re
sults_",ROI,"/T1_img_tone_",ROI,".txt",sep=""),sep=",",header=TRUE))

#format the data for analysis
iresp_ROI$subj = factor(iresp_ROI$subject)
iresp_ROI$img = factor(iresp_ROI$imgtype)

iresp_ROI$trial = 'No Tone' #we're making a new categorical variable for trial type
iresp_ROI$trial[iresp_ROI$trialtype==1] = 'Distractor' #mark distractor and target trial
s
iresp_ROI$trial[iresp_ROI$trialtype==2] = 'Target'

iresp_ROI$age = "Younger" #we're making a new categorical variable with for age group
iresp_ROI$age[iresp_ROI$agegroup==1] <- "Middle"
iresp_ROI$age[iresp_ROI$agegroup==2] <- "Older"

iresp_ROI$age<-factor(iresp_ROI$age,levels=c("Younger","Middle","Older")) #specify the l
evels and order of the age groups
iresp_ROI$trial<-factor(iresp_ROI$trial,levels=c("No Tone","Distractor","Target")) #spec
ify the levels and order of the tone types

iresp_ROI$imgCategory="Scene" #we're making a new categorical variable for image type
iresp_ROI$imgCategory[iresp_ROI$img=="Chair"]<- "Object"
iresp_ROI$imgCategory[iresp_ROI$img=="Car"]<- "Object"
iresp_ROI$imgCategory[iresp_ROI$img=="Female"]<- "Face"
iresp_ROI$imgCategory[iresp_ROI$img=="Male"]<- "Face"

#*****
#load the second data file
#processing steps are the same as for the first data file
iresp_ROI2=as.data.frame(read.table(paste("/home/er482/CAC/MRI_data/ABE_analysis/iresp_r
esults_",ROI,"/T2_img_tone_",ROI,".txt",sep=""),sep=",",header=TRUE))

iresp_ROI2$subj = factor(iresp_ROI2$subject)
iresp_ROI2$img = factor(iresp_ROI2$imgtype)

iresp_ROI2$trial = 'No Tone'
iresp_ROI2$trial[iresp_ROI2$trialtype==1] = 'Distractor'
iresp_ROI2$trial[iresp_ROI2$trialtype==2] = 'Target'

iresp_ROI2$age = "Younger"
iresp_ROI2$age[iresp_ROI2$agegroup==1] <- "Middle"
iresp_ROI2$age[iresp_ROI2$agegroup==2] <- "Older"

iresp_ROI2$age<-factor(iresp_ROI2$age,levels=c("Younger","Middle","Older"))
iresp_ROI2$trial<-factor(iresp_ROI2$trial,levels=c("No Tone","Distractor","Target"))

```

```

iresp_ROI2$imgCategory="Scene"
iresp_ROI2$imgCategory[iresp_ROI2$img=="Chair"]<- "Object"
iresp_ROI2$imgCategory[iresp_ROI2$img=="Car"]<- "Object"
iresp_ROI2$imgCategory[iresp_ROI2$img=="Female"]<- "Face"
iresp_ROI2$imgCategory[iresp_ROI2$img=="Male"]<- "Face"

#*****

#set the colors I want to use for the 3 types of trials I will be analyzing
trialTypeColors = c('dodgerblue','chocolate1','firebrick')

```

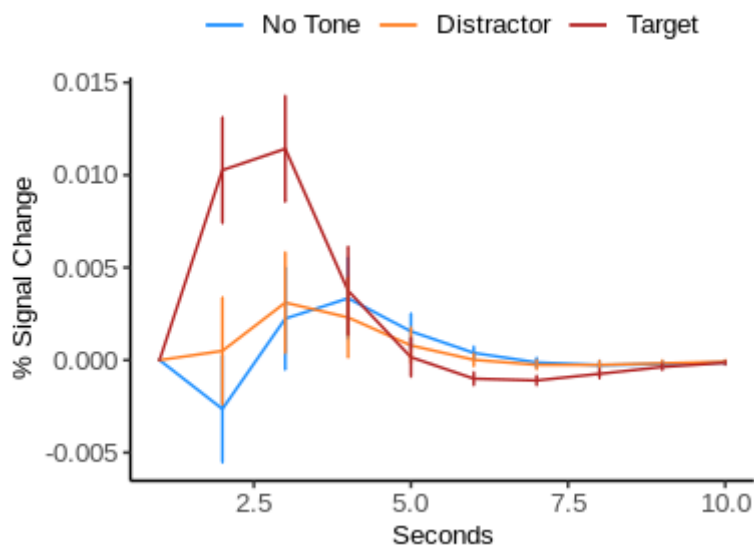
Following this chunk, all the data is loaded and ready for analysis. Next up, graphs.

Hide

```

ggplot(iresp_ROI,
  aes(x=timepoint, y=data, color=trial)) +
  scale_color_manual(values = trialTypeColors) +
  stat_summary(fun = mean, geom = "line") +
  stat_summary(fun.data = mean_se, geom = "errorbar", width=se.width, position = po
sition_dodge(width = 0)) +
  labs(x="Seconds", y="% Signal Change") +
  ggplot.theme

```



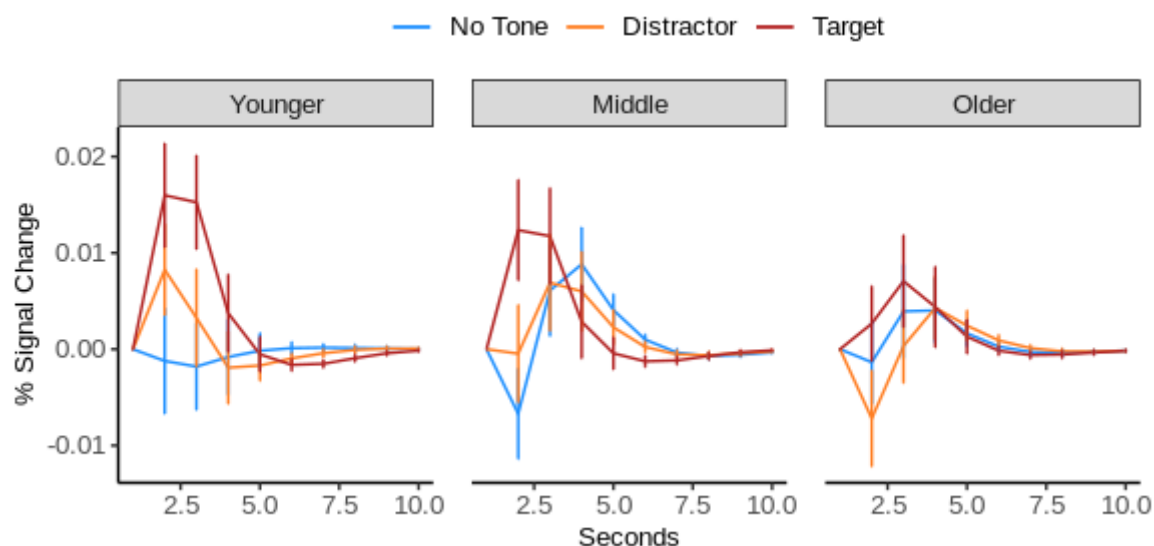
This is a basic first look at the estimated hemodynamic response in our ROI of choice across all age groups and conditions. Next, let's see how hemodynamic response differs across age categories.

Hide

```

ggplot(iresp_ROI,
  aes(x=timepoint, y=data, color=trial)) +
  scale_color_manual(values=trialTypeColors) +
  stat_summary(fun = mean, geom = "line") +
  stat_summary(fun.data = mean_se, geom = "errorbar", width=se.width, position = po
sition_dodge(width = 0)) +
  facet_wrap(~age) +
  labs(x="Seconds", y="% Signal Change") +
  ggplot.theme

```



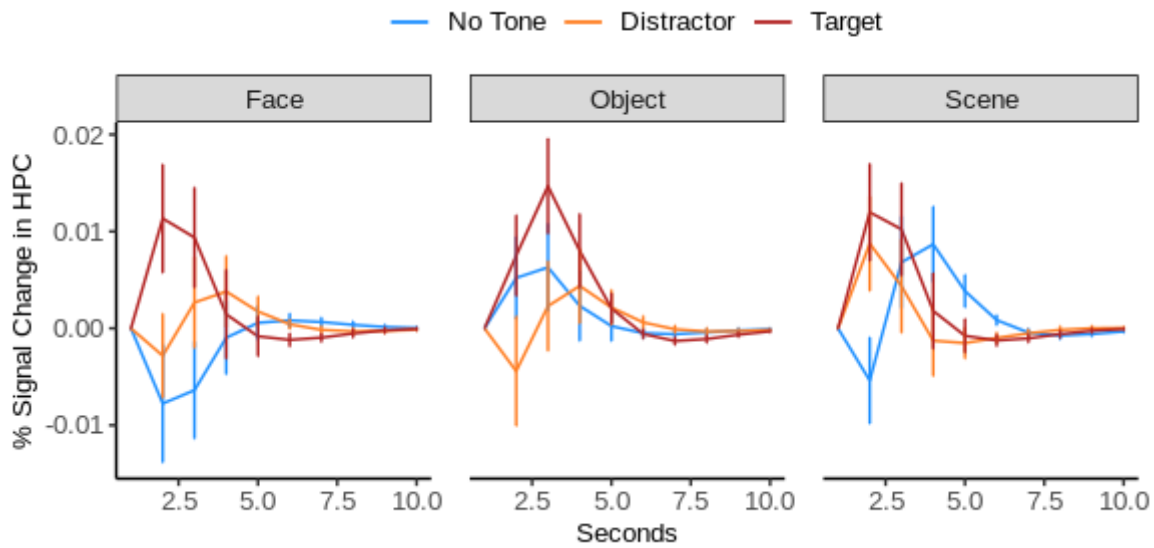
This shows estimated hemodynamic response in the ROI of interest across our 3 age groups. Next, let's see how the responses differ by image category.

Hide

```

ggplot(iresp_ROI,
  aes(x=timepoint, y=data, color=trial)) +
  scale_color_manual(values=trialTypeColors) +
  stat_summary(fun = mean, geom = "line") +
  stat_summary(fun.data = mean_se, geom = "errorbar", width=se.width, position = po
sition_dodge(width = 0)) +
  facet_wrap(~imgCategory) +
  labs(x="Seconds", y="% Signal Change in HPC") +
  ggplot.theme

```

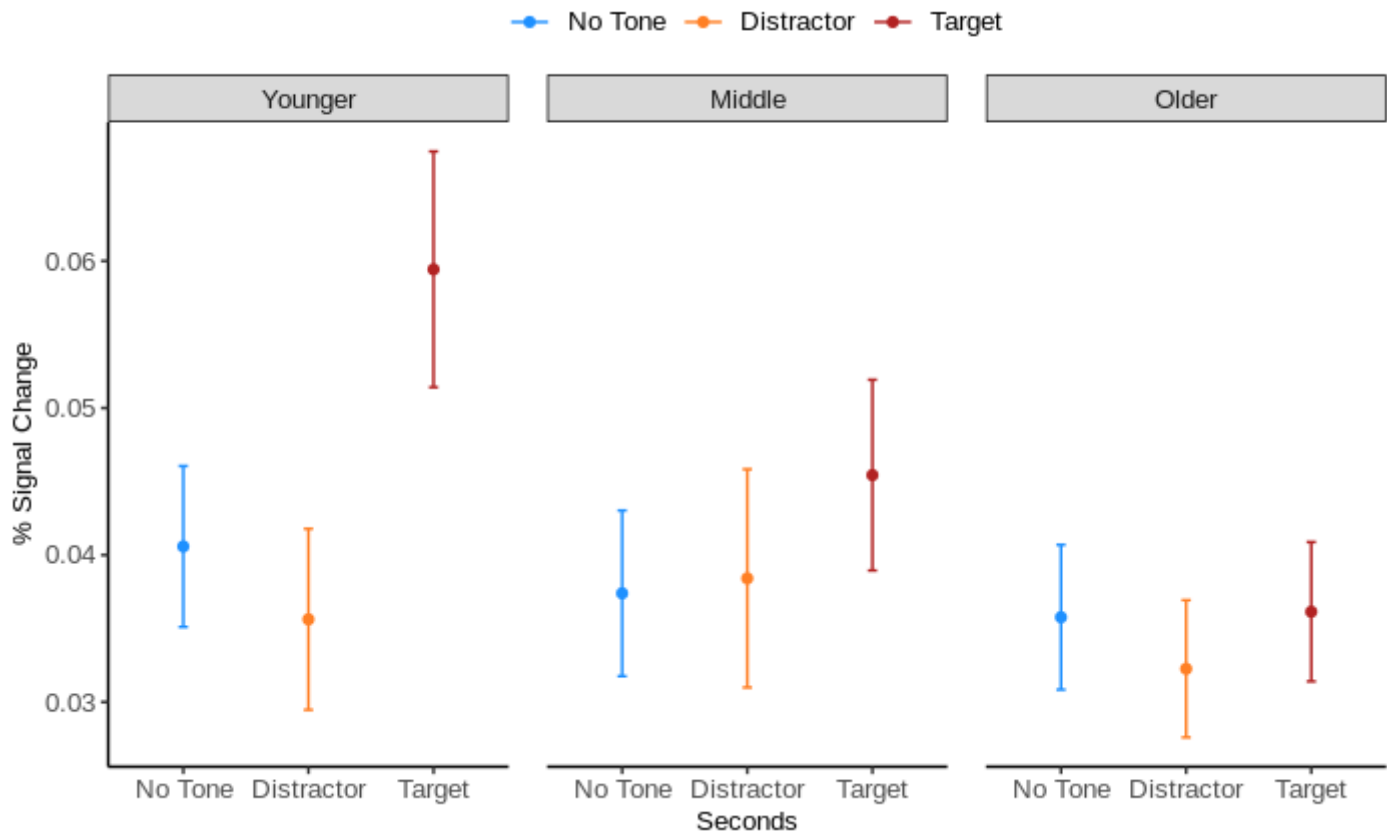


Hemodynamic responses in the ROI of interest across the 3 types of image categories.

Now let's have a look at some summary data.

Hide

```
ggplot(iresp_ROI2[iresp_ROI2$imgCategory=="Face",],
  aes(x=trial, y=maxval, color=trial)) +
  scale_color_manual(values=trialTypeColors) +
  stat_summary(fun = mean, geom = "point") +
  stat_summary(fun.data = mean_se, geom = "errorbar", width=se.width, position = po
sition_dodge(width = 0)) +
  facet_wrap(~age) +
  labs(x="Seconds", y="% Signal Change") +
  ggplot.theme
```

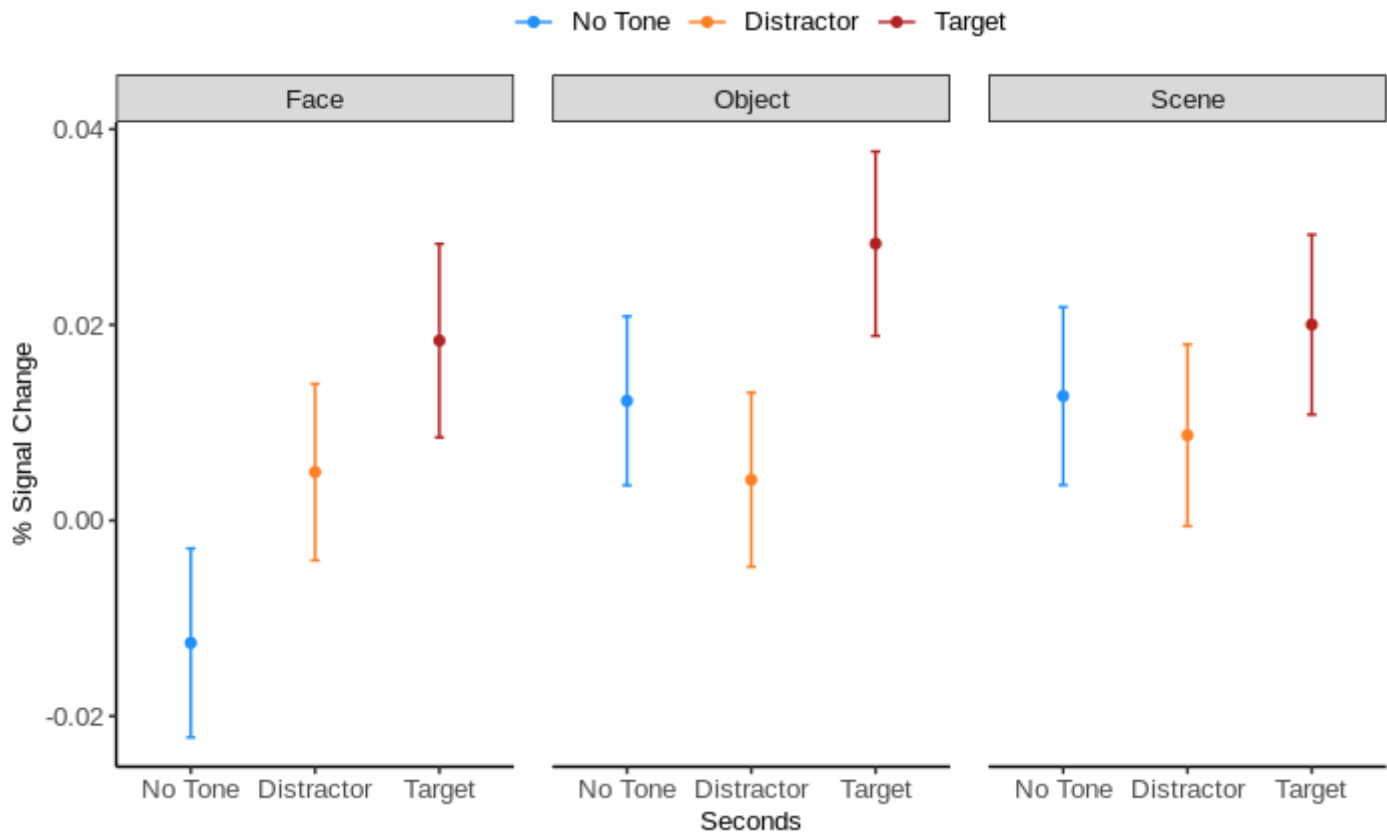


Here we are looking at peak signal change across trial types, and age groups, in only one image category (Faces).

Next, let's see how peak signal change might differ by image category.

Hide

```
ggplot(iresp_ROI2,
  aes(x=trial, y=AUC, color=trial)) +
  scale_color_manual(values=trialTypeColors) +
  stat_summary(fun = mean, geom = "point") +
  stat_summary(fun.data = mean_se, geom = "errorbar", width=se.width, position = po
sition_dodge(width = 0)) +      facet_wrap(~imgCategory) +
  labs(x="Seconds", y="% Signal Change") +
  ggplot.theme
```



Hide

NA  
NA

Peak signal change in all participants, across trial types and image categories.

Now, let's build and evaluate some linear mixed effects models.

Hide

```
#test for effects of trial type and age group on peak signal change with a random effect
of subject
full_model = lmer(AUC ~ trial*age + (1|subj), data = iresp_ROI2)
#summary(full_model) #get model details
anova(full_model) #get stats outputs
```

Type III Analysis of Variance Table with Satterthwaite's method

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)
trial	0.073775	0.036888	2	1218	3.2347	0.03971 *
age	0.005824	0.002912	2	69	0.2554	0.77537
trial:age	0.034461	0.008615	4	1218	0.7555	0.55437

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

Hide

```
tab_model(full_model, show.ci=0.95, show.intercept=TRUE, show.est=TRUE, show.stat=TRUE, show.
obs=TRUE, show.ngroups=FALSE, show.re.var=FALSE, show.fstat=TRUE) #create a nicely formatte
d output of model results
```

Hide

```
#test for the effect of trial type on peak signal change in one age group only, with a r
andom effect of subject
#can be worth investigating if there is a significant effect of trial type in one age gr
oup only
singleAgeGroup_model = lmer(AUC ~ trial + (1|subj), data = iresp_ROI2[iresp_ROI2$age=="Y
ounger",]) #adjust this line to pick which group
#summary(singleAgeGroup_model) #get model details
anova(singleAgeGroup_model) #get stats outputs
```

```
Type III Analysis of Variance Table with Satterthwaite's method
      Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
trial 0.097945 0.048973     2   474   3.718 0.02499 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Hide

```
tab_model(singleAgeGroup_model, show.ci=0.95, show.intercept=TRUE, show.est=TRUE, show.stat=
TRUE, show.obs=TRUE, show.ngroups=FALSE, show.re.var=FALSE, show.fstat=TRUE) #create a nicel
y formatted output of model results
```

Hide

```
#test for effects of trial type and age group on peak signal change for target tones onl
y
oneToneType_model = lmer(AUC ~ age + (1|subj), data = iresp_ROI2[iresp_ROI2$trial=="Targ
et",])
#summary(oneToneType_model)
anova(oneToneType_model)
```

```
Type III Analysis of Variance Table with Satterthwaite's method
      Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
age 0.0092554 0.0046277     2    69   0.4309 0.6516
```

Hide

```
tab_model(oneToneType_model, show.ci=0.95, show.intercept=TRUE, show.est=TRUE, show.stat=TRU
E, show.obs=TRUE, show.ngroups=FALSE, show.re.var=FALSE, show.fstat=TRUE)
```

Hide



```
#next, instead of using summary statistics like peak signal change, we'll use the entire
estimated hemodynamic response
full_model_bypoint = lmer(data ~ timepoint*age*trial + (1|subj), data = iresp_ROI)
summary(full_model_bypoint)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: data ~ timepoint * age * trial + (1 | subj)
Data: iresp_ROI
```

```
REML criterion at convergence: -53050.3
```

```
Scaled residuals:
```

```
      Min       1Q   Median       3Q      Max
-14.0543  -0.1891  -0.0125   0.1392  11.3278
```

```
Random effects:
```

```
Groups   Name             Variance Std.Dev.
subj     (Intercept)  0.0000105 0.003241
Residual                    0.0009545 0.030895
```

```
Number of obs: 12960, groups: subj, 72
```

```
Fixed effects:
```

	Estimate	Std. Error	df	t value	Pr(> t )
(Intercept)	-1.089e-03	1.740e-03	1.800e+03	-0.626	0.531541
timepoint	1.345e-04	2.624e-04	1.287e+04	0.512	0.608331
ageMiddle	3.044e-03	2.781e-03	1.800e+03	1.094	0.273943
ageOlder	2.818e-03	2.507e-03	1.800e+03	1.124	0.261130
trialDistractor	4.017e-03	2.303e-03	1.287e+04	1.744	0.081117 .
trialTarget	1.120e-02	2.303e-03	1.287e+04	4.864	1.16e-06 **
*					
timepoint:ageMiddle	-2.847e-04	4.195e-04	1.287e+04	-0.679	0.497411
timepoint:ageOlder	-3.143e-04	3.782e-04	1.287e+04	-0.831	0.406000
timepoint:trialDistractor	-5.472e-04	3.711e-04	1.287e+04	-1.475	0.140362
timepoint:trialTarget	-1.431e-03	3.711e-04	1.287e+04	-3.857	0.000115 **
*					
ageMiddle:trialDistractor	-2.621e-03	3.681e-03	1.287e+04	-0.712	0.476454
ageOlder:trialDistractor	-6.734e-03	3.319e-03	1.287e+04	-2.029	0.042451 *
ageMiddle:trialTarget	-5.360e-03	3.681e-03	1.287e+04	-1.456	0.145374
ageOlder:trialTarget	-9.003e-03	3.319e-03	1.287e+04	-2.713	0.006681 **
timepoint:ageMiddle:trialDistractor	3.268e-04	5.933e-04	1.287e+04	0.551	0.581730
timepoint:ageOlder:trialDistractor	9.115e-04	5.348e-04	1.287e+04	1.704	0.088364 .
timepoint:ageMiddle:trialTarget	5.813e-04	5.933e-04	1.287e+04	0.980	0.327181
timepoint:ageOlder:trialTarget	1.144e-03	5.348e-04	1.287e+04	2.138	0.032500 *

```
---
```

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

```
Correlation matrix not shown by default, as p = 18 > 12.
```

```
Use print(x, correlation=TRUE) or
vcov(x) if you need it
```

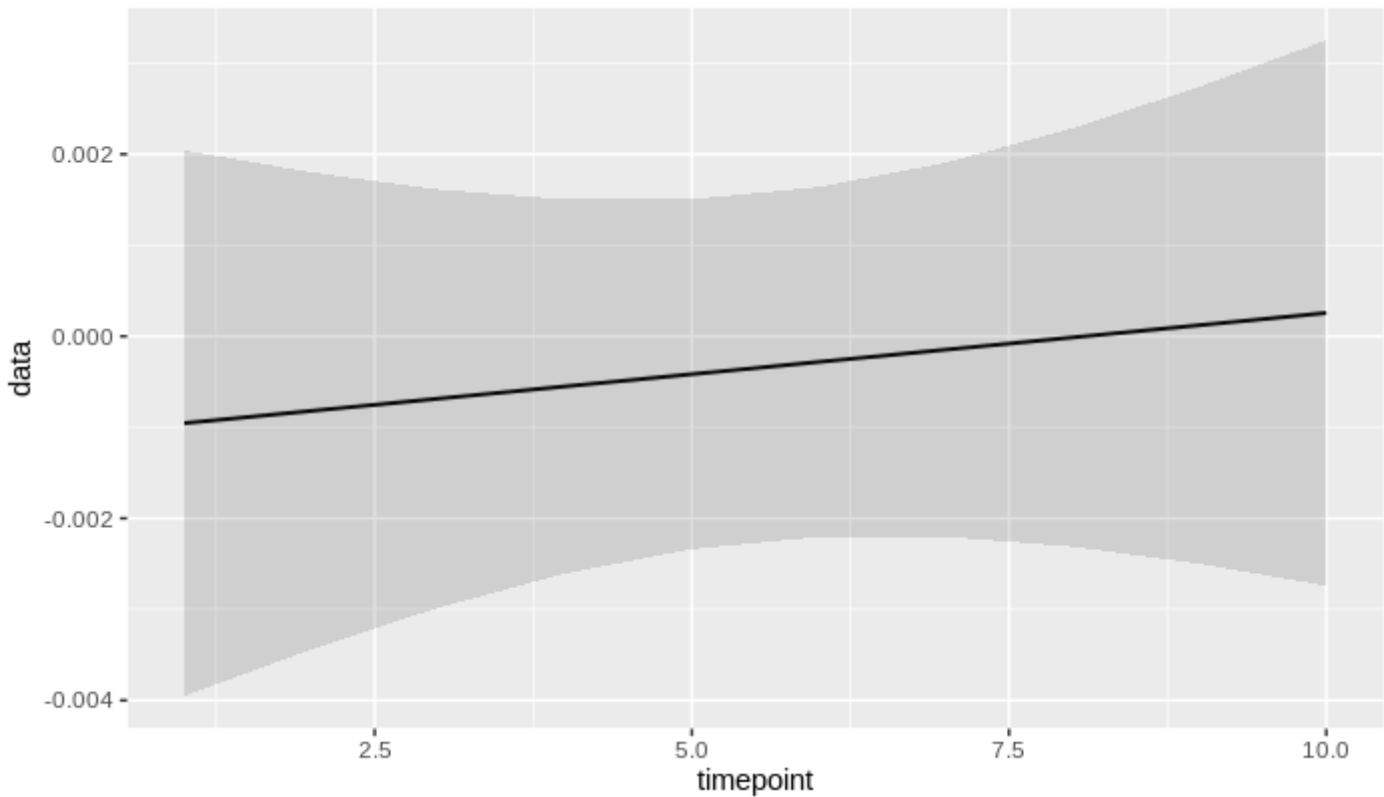
```
#anova(full_model_bypoint)
tab_model(full_model_bypoint, show.ci=0.95, show.intercept=TRUE, show.est=TRUE, show.stat=TRUE, show.obs=TRUE, show.ngroups=FALSE, show.re.var=FALSE, show.fstat=TRUE, pred.labels=c("Intercept", "Timepoint", "Age:Middle", "Age:Older", "Trial:Distractor", "Trial:Target", "Timepoint*Middle", "Timepoint*Older", "Timepoint*Distractor", "Timepoint*Target", "Middle*Distractor", "Older*Distractor", "Middle*Target", "Older*Target", "Timepoint*Middle*Distractor", "Timepoint*Older*Distractor", "Timepoint*Middle*Target", "Timepoint*Older*Target")) #this time, I've put in the effort to make it fully formatted
plot_model(full_model_bypoint, type="pred") #let's plot the model's predictions
```

```
$timepoint
```

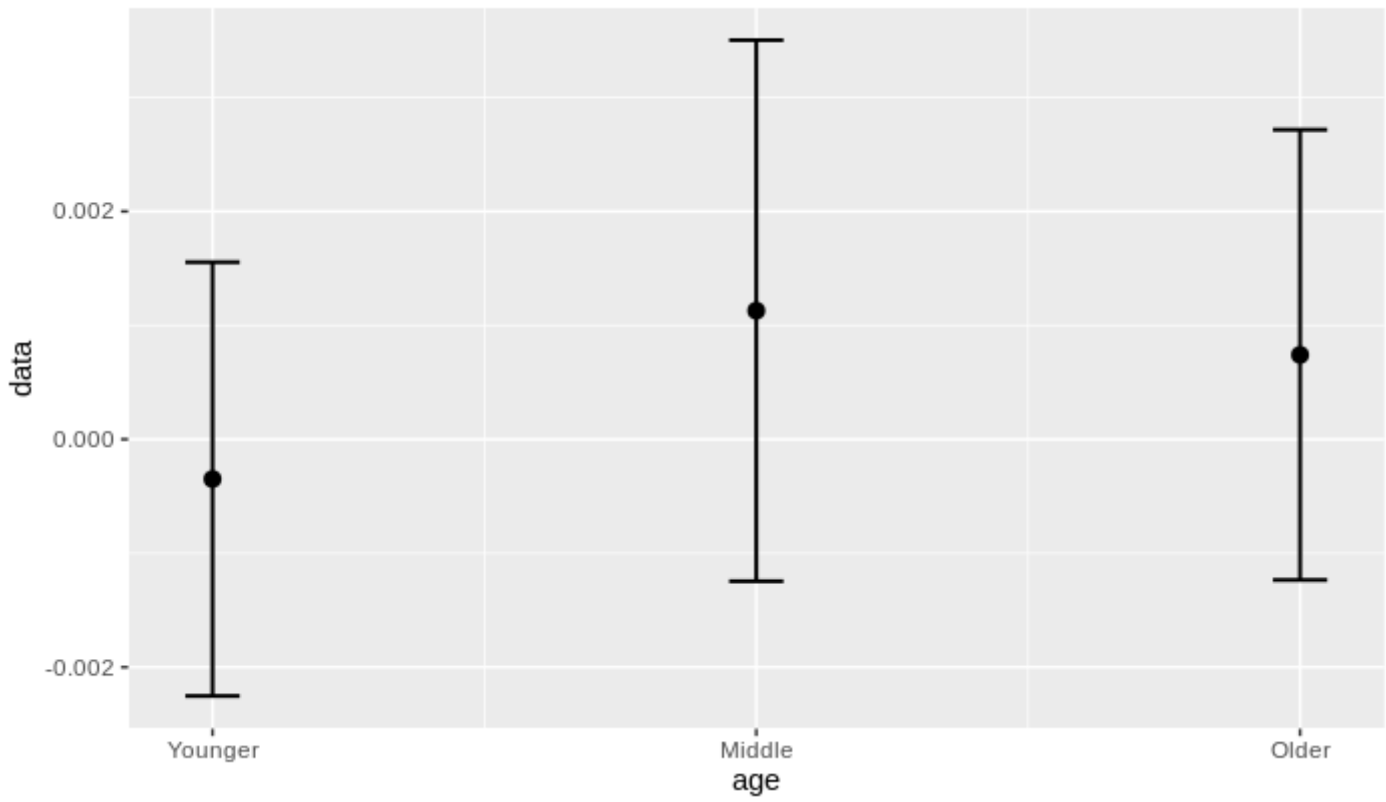
```
$age
```

```
$trial
```

Predicted values of data



Predicted values of data



Predicted values of data

