

Assignment 1

Group 9 - Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad

13 feb 2019

```
# Loading and installing Libraries
library(pacman)
pacman::p_load(tidyverse, dplyr, plyr, lme4, nlme)

# setting working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Portfolios/1")

# Loading sleepstudy file into R
sleepstudy <- read.csv("sleepstudy.csv")
```

- Task 1 -

```
#1a
# slicing out participant #308
participant1 <- subset(sleepstudy, Subject == "308")

# applying Linear regression to data from subject 308
participant1lm <- lm(Reaction ~ Days, participant1)

# printing statistical summary,
summary(participant1lm)

##
## Call:
## lm(formula = Reaction ~ Days, data = participant1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -106.397   -4.098    9.688   22.269   61.674 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 244.19     28.08   8.695 2.39e-05 ***
## Days        21.77      5.26   4.137  0.00326 **  
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.78 on 8 degrees of freedom
```

```

## Multiple R-squared:  0.6815, Adjusted R-squared:  0.6417
## F-statistic: 17.12 on 1 and 8 DF,  p-value: 0.003265

# F-statistics are  $F(1,8) = 17.12$ 

#1b
# The relevant F-distribution has 1,8 degrees of freedom.

#1c
# At  $F(1,8) = 5.32$  the test becomes significant. Above this cut-off point the test
will become significantly different from  $H_0$ , assuming the critical value is  $p < 0.$ 
 $05$ .

#1d
# plotting F-distribution with 1,8 degrees of freedom:

# defining cut-off point for the F-distribution in question
cutoff = 5.32

# making a string of 100 numbers between 0 and 20 to use when generating the F distribution
nn<-seq(0,20,len=100)

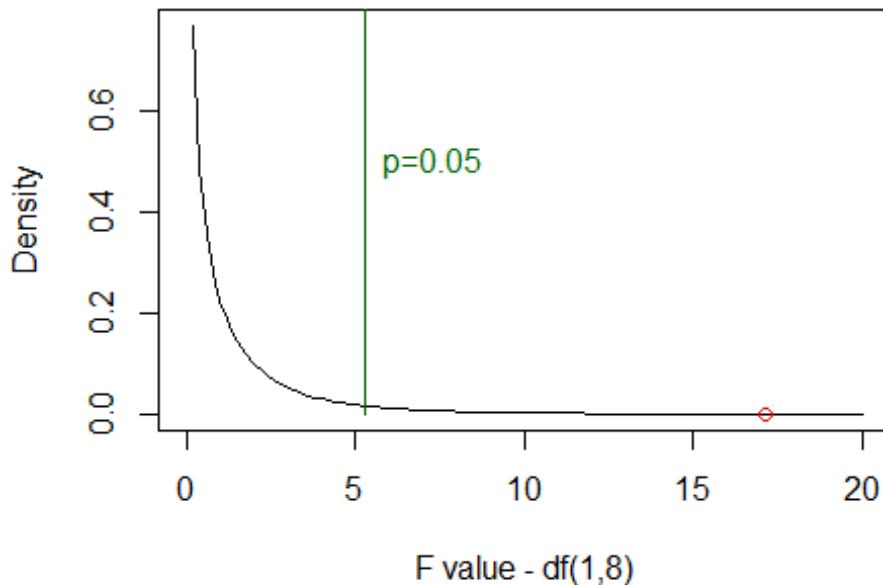
# getting the F distribution using df()
fdist<-df(nn,1,8)
plot(nn,fdist, type='l',xlab='F value - df(1,8)',ylab='Density')

# plotting a vertical line at the cutoff (5.32)
lines(c(cutoff,cutoff),c(0,1),col='darkgreen')

# adding explanation for the line at the cutoff
text(cutoff+2,0.5,'p=0.05',col='darkgreen')

# drawing our tests specific F-value outcome (17.12) as a point on the curve
points(17.12,df(17.12,1,8),col='red')

```



- Task 2 -

```
#2a
# applying regression for reaction time as a function of days of sleep deprivation
# in order to find the slopes and intercepts from each subject in the study
lmsleep <- dplyr::group_by(sleepstudy, Subject)
lmsleep <- lmsleep %>% lm(Reaction ~ Days, data = .)

# creating dataframe with intercepts and slopes
results <- lapply(lmsleep, coef)
print(results)

## # Subject (Intercept) Days
## 1 308 244.1927 21.764702
## 2 309 205.0549 2.261785
## 3 310 203.4842 6.114899
## 4 330 289.6851 3.008073
## 5 331 285.7390 5.266019
## 6 332 264.2516 9.566768
## 7 333 275.0191 9.142045
## 8 334 240.1629 12.253141
## 9 335 263.0347 -2.881034
## 10 337 290.1041 19.025974
## 11 349 215.1118 13.493933
## 12 350 225.8346 19.504017
```

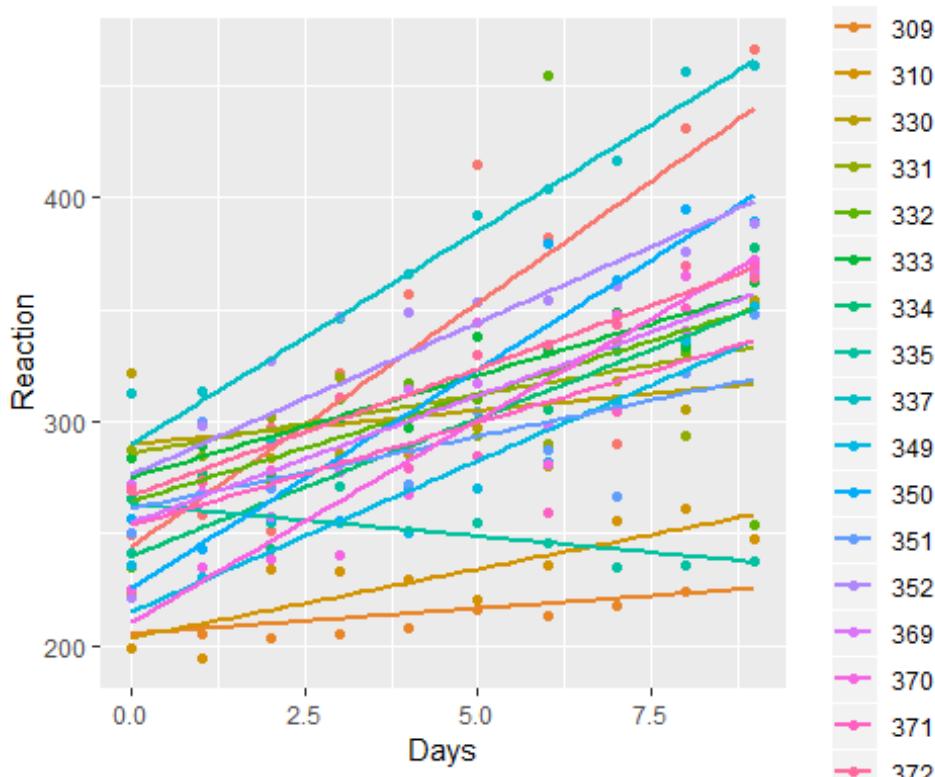
```

## 13    351    261.1470  6.433498
## 14    352    276.3721 13.566549
## 15    369    254.9681 11.348109
## 16    370    210.4491 18.056151
## 17    371    253.6360  9.188445
## 18    372    267.0448 11.298073

#2b
# making subjects onto catagorical values instead of continuous
sleepstudy$Subject <- as.factor(sleepstudy$Subject)

# creating a scatterplot with the regression lines for each participant ##### DOESN T WORK, ONLY 1 REGRESSION LINE)
scatter <- ggplot(sleepstudy, aes(x = Days, y = Reaction, color = Subject)) + geom_point() + geom_smooth(method = lm, se = F)
scatter

```



```

#2c
# creating Lists of Lists for each Subject with values from Linear regression
list <- nlme::lmList(Days ~ Reaction | Subject, sleepstudy)

# creating a List of statistical summaries
list.df <- summary(list)

# selecting statistical coefficients whilst creating a dataframe

```

```

list.df <- data.frame(list.df$coefficients)

# manually entering degrees of freedom to dataframe
list.df <- data.frame(list.df, df_1 = c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1), df_2
= c(8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8))
print(list.df)

##      Estimate..Intercept. Std..Error..Intercept. t.value..Intercept.
## 308       -6.213159        2.796757       -2.2215588
## 309      -33.671925       12.696449       -2.6520742
## 310      -22.606526        6.760532       -3.3438976
## 330      -11.430882        8.453904       -1.3521424
## 331      -15.626461        7.261501       -2.1519602
## 332      -2.016846        3.103855       -0.6497875
## 333      -24.805176        6.725928       -3.6879931
## 334      -14.432939        4.535222       -3.1824107
## 335      39.009225       11.532153        3.3826488
## 337      -13.932518        4.059824       -3.4318038
## 349      -14.006149        4.137011       -3.3855718
## 350      -9.466169        3.210628       -2.9483853
## 351      -15.872383        6.406191       -2.4776630
## 352      -14.018095        4.556108       -3.0767699
## 369      -18.186110        5.239636       -3.4708729
## 370      -9.272468        3.196375       -2.9009329
## 371      -14.143341        5.183245       -2.7286653
## 372      -21.154777        5.685575       -3.7207807
##      Pr....t....Intercept. Estimate.Reaction Std..Error.Reaction
## 308      0.0278737286     0.03131277      0.007981294
## 309      0.0088968609     0.17735165      0.058922454
## 310      0.0010531740     0.11734362      0.029149024
## 330      0.1784495269     0.05253878      0.027808958
## 331      0.0330677000     0.06504239      0.023385462
## 332      0.5168646726     0.02120665      0.009907003
## 333      0.0003195760     0.09269146      0.021187860
## 334      0.0017896013     0.06411381      0.015220923
## 335      0.0009247299    -0.13799824      0.046052318
## 337      0.0007828617     0.04905906      0.010685013
## 349      0.0009156614     0.06709150      0.014837249
## 350      0.0037283897     0.04453460      0.010054851
## 351      0.0143806575     0.07022592      0.021984372
## 352      0.0025052732     0.05488119      0.013383385
## 369      0.0006849466     0.07412922      0.017006772
## 370      0.0043049839     0.04721421      0.010760010
## 371      0.0071517496     0.06320118      0.017451410
## 372      0.0002840047     0.08070430      0.017784229
##      t.value.Reaction Pr....t....Reaction df_1 df_2
## 308      3.923270     1.347999e-04      1      8
## 309      3.009916     3.086315e-03      1      8

```

```

## 310      4.025645    9.150920e-05    1     8
## 330      1.889275    6.086540e-02    1     8
## 331      2.781317    6.138826e-03    1     8
## 332      2.140571    3.399287e-02    1     8
## 333      4.374744    2.319077e-05    1     8
## 334      4.212216    4.437382e-05    1     8
## 335      -2.996554   3.216409e-03    1     8
## 337      4.591389    9.516150e-06    1     8
## 349      4.521829    1.270704e-05    1     8
## 350      4.429165    1.859207e-05    1     8
## 351      3.194357    1.721903e-03    1     8
## 352      4.100696    6.858231e-05    1     8
## 369      4.358806    2.473278e-05    1     8
## 370      4.387934    2.198497e-05    1     8
## 371      3.621552    4.049719e-04    1     8
## 372      4.537970    1.188552e-05   1     8

#2d
# finding the number of participants displaying statistically significant effects
# of sleep deprivation
significant <- filter(list.df, Pr....t....Reaction < 0.05)
nrow(significant)

## [1] 17

# there are 17 subjects who displayed significant effects of the sleep deprivation

```

- Task 3 -

```

#3a
# testing whether the slopes deviate from 0
t.test(results$Days, mu = 0)

##
## One Sample t-test
##
## data: results$Days
## t = 6.7715, df = 17, p-value = 3.264e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  7.205956 13.728615
## sample estimates:
## mean of x
## 10.46729

#3b
# We chose to conduct a t-test to test the hypothesis that slopes are larger than
# zero against the null-hypothesis that the slopes are zero as it allows to test for
# evidence of a significant difference between the sample mean and a hypothesized va
lue.

```

```

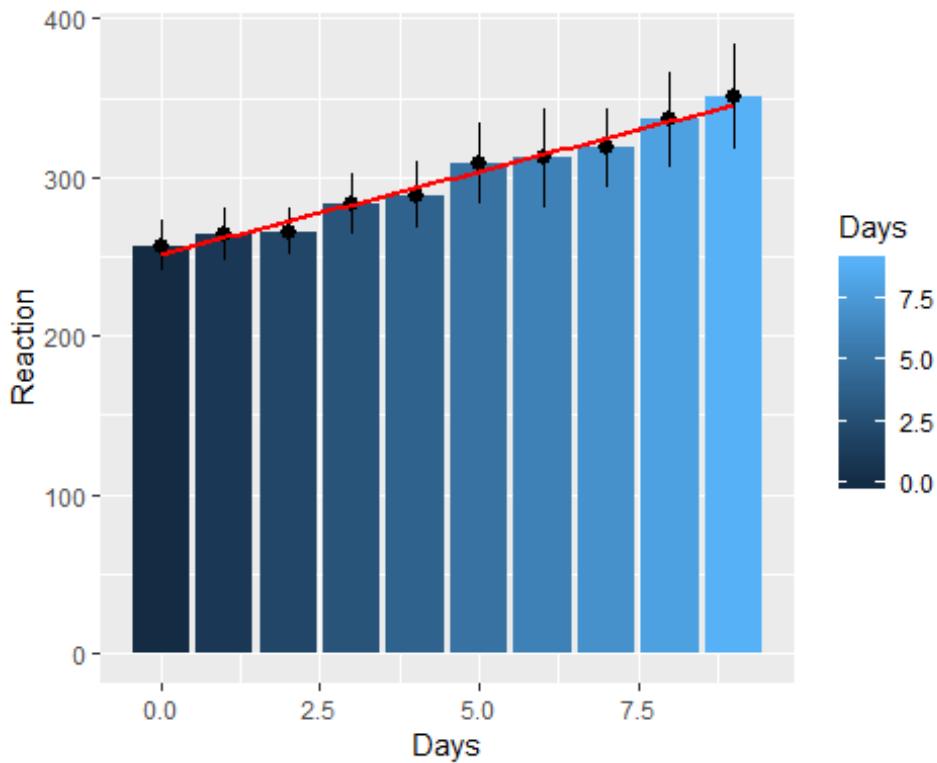
#3c
# Creating a data frame with the t-test inferential statistics
coeficients_results <- data.frame(t_value = c(10.37), p_value = c(2.325e-14), df =
c(53), mean = c(201.0501))
print(coeficients_results)

##   t_value   p_value df      mean
## 1 10.37 2.325e-14 53 201.0501

# the slopes are significantly different from zero,  $t(53) = 10.37$ ,  $p\text{-value} < 0.0001$ 

#3d
# creating a graph with each day as a column
barchart <- ggplot(sleepstudy, aes(Days, Reaction, fill = Days)) + stat_summary(fun.y = mean, geom = "bar") + stat_summary(fun.data = mean_cl_normal, geom = "pointrange") + geom_smooth(method = "lm", se = F, color = "red")
barchart

```



Assignment 2

Group 9 - Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad

13 feb 2019

```
# Load Libraries
library(pacman)
p_load(lme4, rgl, plyr, grid, gridExtra, wesanderson, widyr, igraph, ggraph, SnowballC, RCur
l, XML, RColorBrewer, tm, wordcloud, utf8, readtext, tidytext, scales, tidyverse, dplyr, tidy
r, topicmodels, LDavis, scales, jpeg)

# set working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Por
tfolios/2")

# create vectors for participant 372
Reaction372<-c(269.41, 273.47, 297.60, 310.63, 287.17, 329.61, 334.48, 343.22, 369
.14, 364.12)
Days372<-c(0,1,2,3,4,5,6,7,8,9)

#1a
# create constant vector
constantVector<- c(1,1,1,1,1,1,1,1,1,1)

#1b
# find inner product
innerProduct <- sum(constantVector*Days372)

#1c
# If two vectors are separated by a 90 degrees angle, they are uncorrelated and we
say that they are orthogonal. Orthogonal vectors have a dot-product of zero. If th
e angle is smaller, the vectors are correlated. If the angle is bigger, the vecto
rs are anti-correlated

#1d
# create matrix
X <- matrix(c(Days372,constantVector),nrow=10,ncol=2)
# use least sum of squares manually to find slope and intercept
beta.hat <- solve( t(X) %*% X ) %*% t(X) %*% Reaction372
beta.hat

##          [,1]
## [1,]  11.298
## [2,] 267.044
```

```

# slope equals 267.04, while intercept equals 11.30

#1e
# using lm to check manually calculated coefficients
lm(Reaction372~0+X)

##
## Call:
## lm(formula = Reaction372 ~ 0 + X)
##
## Coefficients:
##       X1      X2
##  11.3  267.0

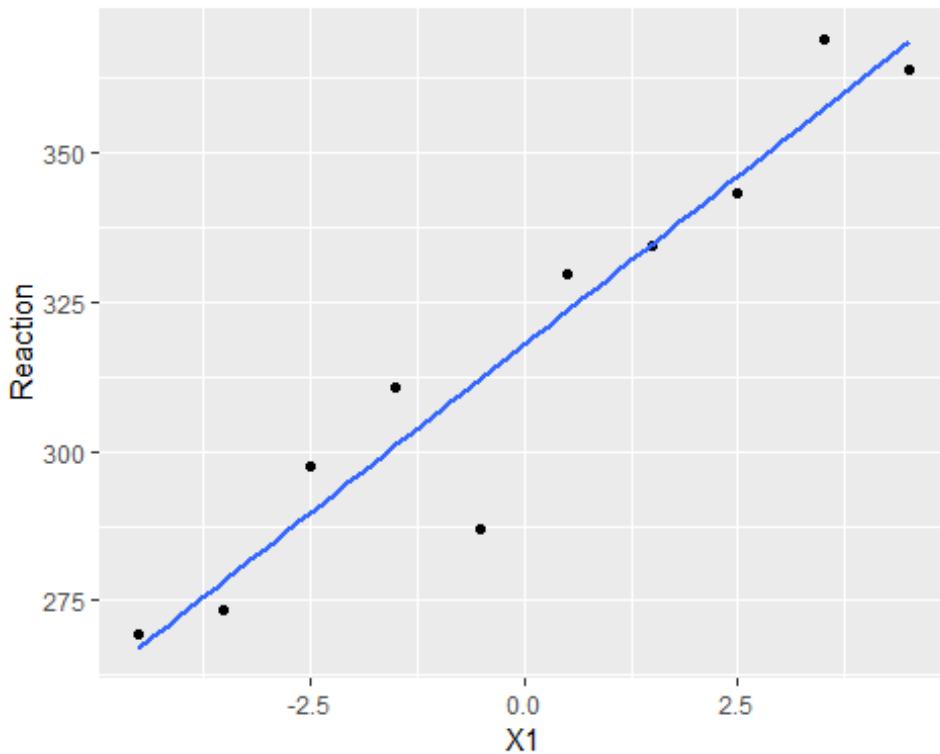
#1f
# subtract mean values from individual values to create mean-centered days vector
newVector <- Days372-mean(Days372)
# create new matrix with new set of values
X_new <- matrix(c(newVector,constantVector),nrow=10,ncol=2)
# use least sum of squares manually to find slope and intercept
beta.hatNew <- solve( t(X_new) %*% X_new ) %*% t(X_new) %*% Reaction372
beta.hatNew

##          [,1]
## [1,]  11.298
## [2,] 317.885

# the intercept changes, while the slope remains the same

#1g
# create a scatterplot displaying mean-centered days covariate against response time
X_df <- data.frame(X_new)
X_df$Reaction <- Reaction372
ggplot(X_df,aes(x=X1,y=Reaction))+geom_point()+geom_smooth(method = lm, se=F)

```



```

# Load data into matrix
matrix<-readJPEG('portfolio_assignment2_matrices_data.jpg', native = FALSE)

#2a
# calculating different matrix x characteristics
min(matrix)
## [1] 0.0627451

max(matrix)
## [1] 1

mean(matrix)
## [1] 0.5118474

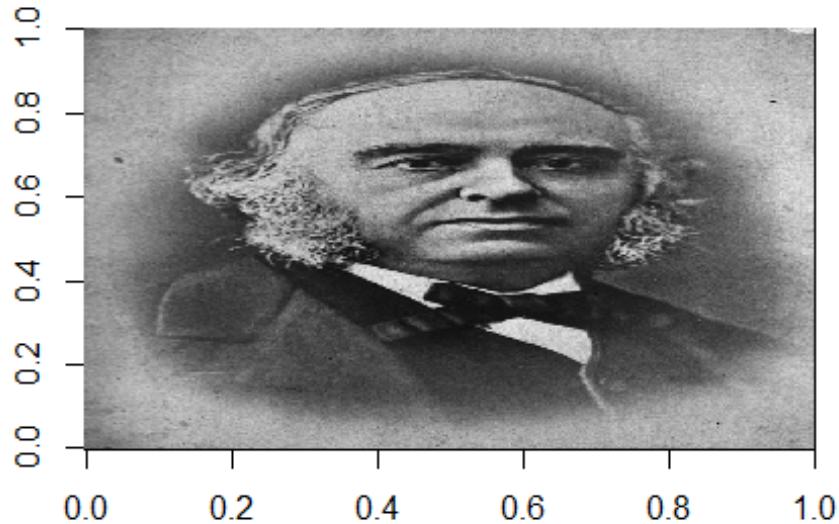
nrow(matrix)
## [1] 900

ncol(matrix)
## [1] 606

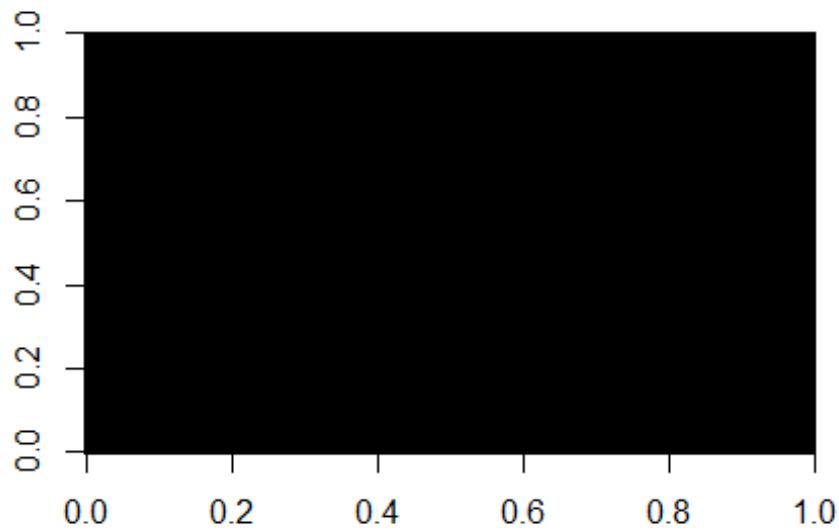
# the matrix has a minimum pixel value at 0.063 and a max of 1. The mean pixel value is 0.512, while the number of rows is 900 and the number of columns is 606.

```

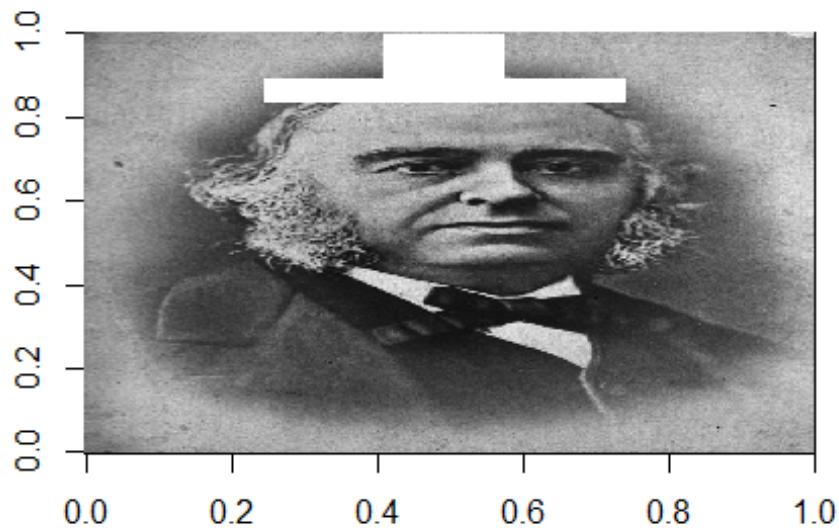
```
#2b  
# rotate image appropriately  
rotate <- function(x) t(apply(x, 2, rev))  
r_matrix <- rotate(matrix)  
# create gray scale image from matrix  
image(r_matrix, col=gray(1:100/100))
```



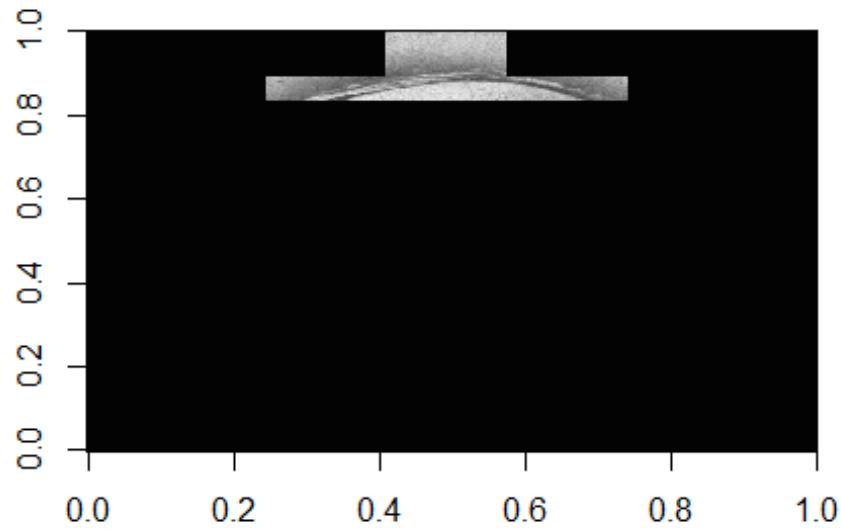
```
#2c  
# create black matrix in same dimensions as previous  
matrixBlack <- matrix(0L, nrow = 900, ncol = 606)  
matrixBlackR <- rotate(matrixBlack)  
# print black matrix as image  
image(matrixBlackR, col=gray(0))
```



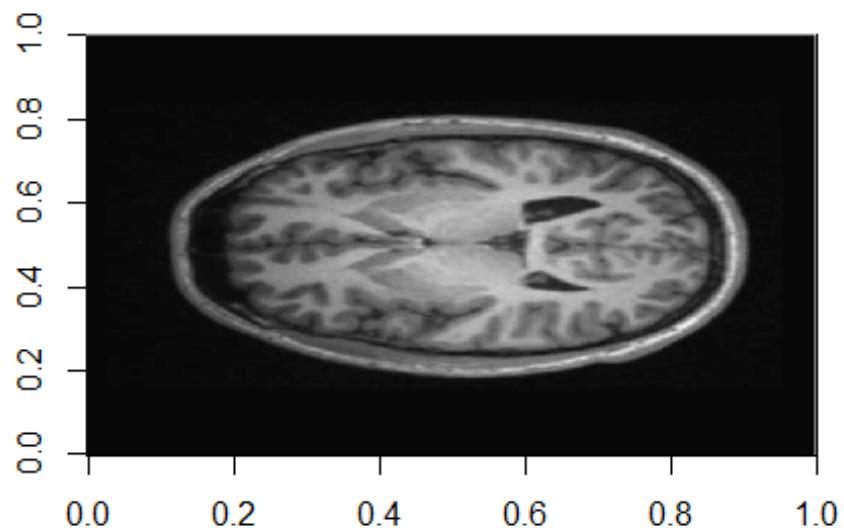
```
#2d  
# create a white hat in Broca matrix  
r_matrix1 <- r_matrix  
r_matrix1[450: 150, 750: 800] <- 1  
r_matrix1[350:250, 800:895] <- 1  
image(r_matrix1, col=gray(1:100/100))
```



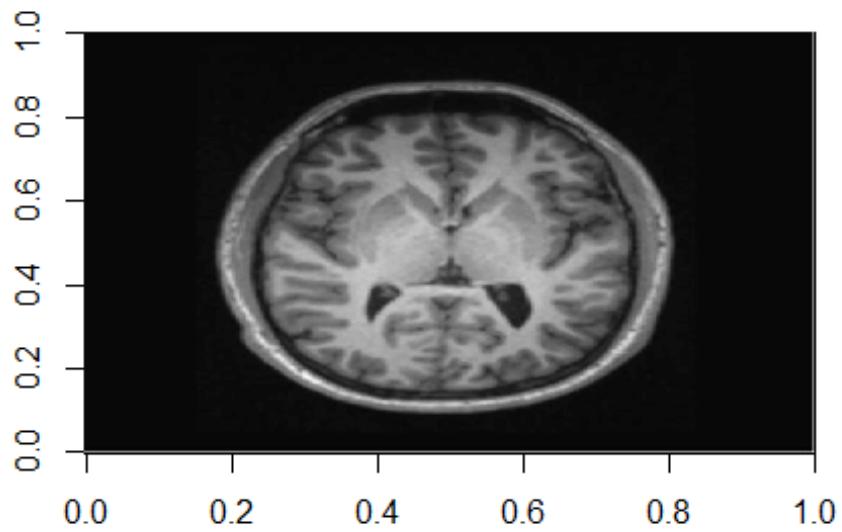
```
#2e
# create image that was hidden behind hat in 2d
matrixBlackR[450: 150, 750: 800] <- 1
matrixBlackR[350:250, 800:895] <- 1
r_matrix2 <- r_matrix
image(matrixBlackR*r_matrix2, col = gray(1:100/100))
```



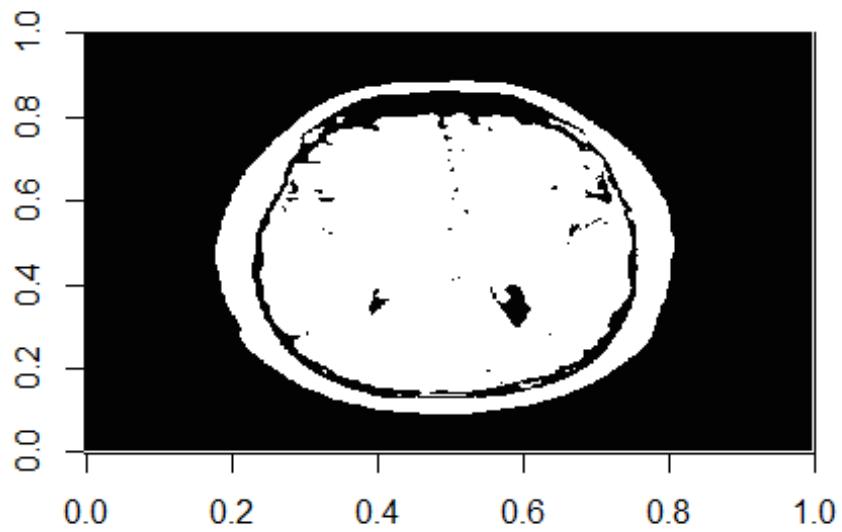
```
#3a  
# Loading image of brain  
brain <- readJPEG('portfolio_assignment2_matrices_data2.jpg', native = FALSE)  
image(brain, col=gray(1:100/100))
```



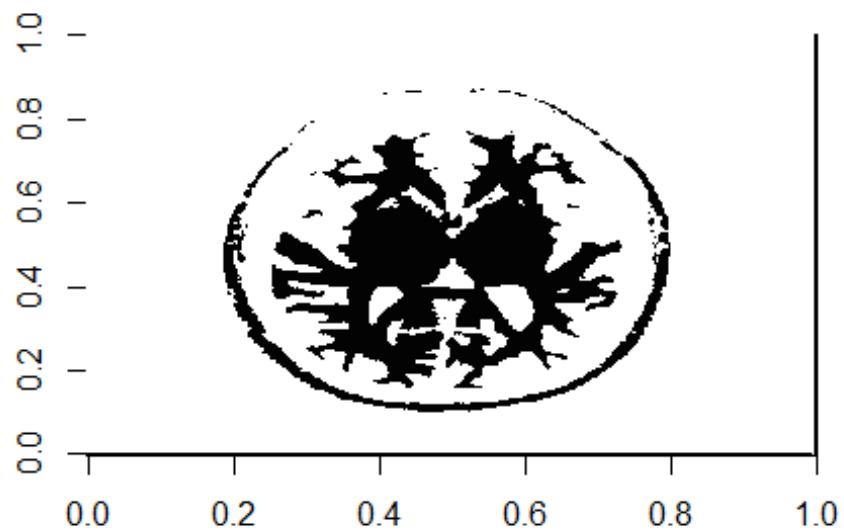
```
# rotate appropriately
Brain <- rotate(brain)
# print image
image(Brian, col=gray(1:100/100))
```



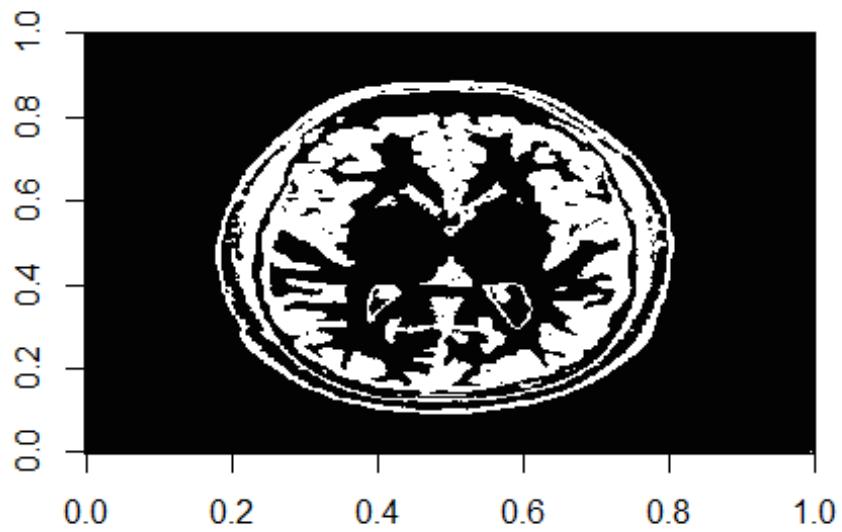
```
#3b  
# make mask 1 where pixel values above average are converted to white/ones are larger than the mean value of the whole image  
mask1 <- Brian>mean(Brian)  
# print mask 1  
image(mask1, col=gray(1:100/100))
```



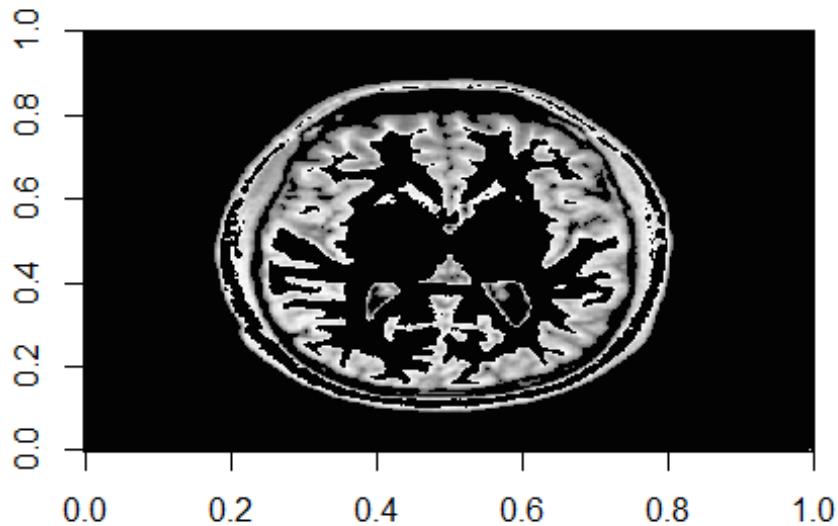
```
#3c  
# make mask 2 where pixel values under 2.5 times the mean are converted to white/ones  
mask2 <- Brian<mean(Brian)*2.5  
# print mask 2  
image(mask2, col=gray(1:100/100))
```



```
#3d  
# create a mask 3 with ones at the places where mask 1 and 2 overlap  
mask3<- mask1*mask2  
# print mask 3  
image(mask3, col=gray(1:100/100))
```



```
#3e  
# apply mask 3 to the brain in order to detect gray matter  
Brian <- mask3*Brain  
# print image of brain overlaid by mask 3  
image(Brian, col=gray(1:100/100))
```



```

# we have to some extent managed to select the gray matter from the white matter and dark sulci

#3f count number of pixels in combined mask
npixel<-sum(mask3)
# the mask contains 50004 pixels

#4a
# solve two equations with two unknown using matrixes
# entering the linear equations into matrix form
brocaBodega <- matrix(c(3,1,1,2),ncol=2)
drinksSold <- c(116,92)

# solve the system of Linear equations. If  $Mx = z$  where  $M$  is a matrix and  $x$  and  $z$  are vectors the solution is  $x = M^{-1}z$ 
boughtByGender<- solve(brocaBodega)%*%drinksSold
boughtByGender

##      [,1]
## [1,]    28
## [2,]    32

# the result of the equation is that there are 28 males and 32 females

```

Assingment 3

Group 9 - Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea HÃ,lmkjÃ'r Kaad

2/27/2019

```
library(pacman)
p_load(psych,lme4,rgl,plyr,grid,gridExtra,wesanderson,widyr,igraph,ggraph,SnowballC,RCurl,XML,RColorBrewer,tm,wordcloud,utf8,readtext,tidytext,scales,tidyverse,dplyr,tidyr,topicmodels,LDAvis,scales,reshape2)

# setting working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Portfolios/3")

# importing the eksperimental data
fmri<-as.matrix(read.csv("aud_fmri_data37.csv", header = FALSE))

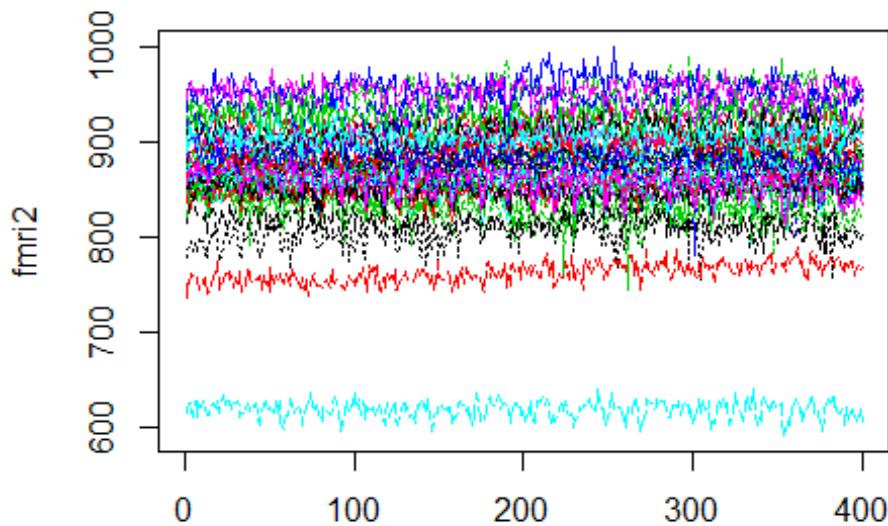
# making data to a time-series
fmri2<-ts(fmri)

# importing design or model
fmrides<-as.matrix(read.csv("aud_fmri_design.csv", header = FALSE))

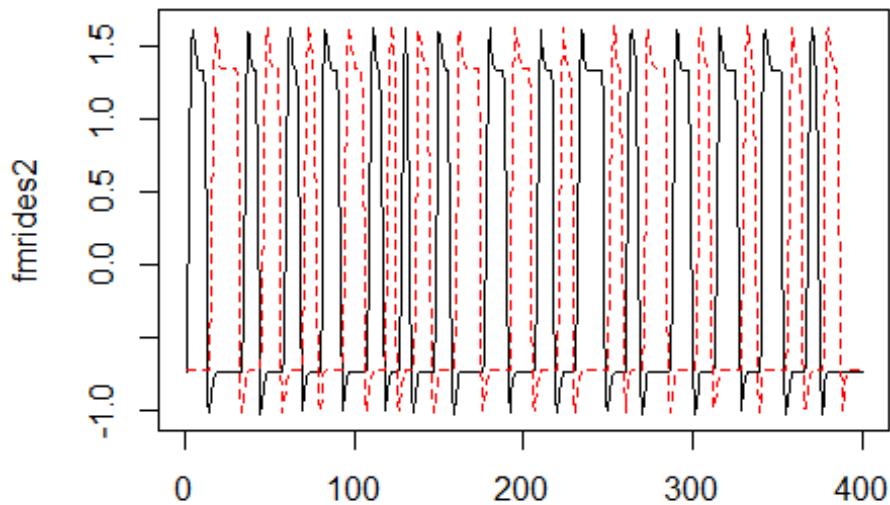
# making the model a time-series
fmrides2<-ts(fmrides)
```

Analyses

```
#1
#1.a.
# creating a LinePlot of the data from all participants as a function of time
matplotlib(x=fMRI2,type="l")
```



```
#1.b.  
# creating a lineplot showcasing the model covariates  
matplotlib(x=fmrides2,type="l")
```



#2

#Looking at the Lineplot, we can see that each participant listened to 15 stories (blocks or peaks) in each condition

#3a

```
# using the lm function in order to test whether the two model covariates are correlated
checkCorr<-lm(fmrides2[,1] ~ fmrides2[,2])
summary(checkCorr)

##
## Call:
## lm(formula = fmrides2[, 1] ~ fmrides2[, 2])
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.41315 -0.75018  0.00973  0.93735  1.22740 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -2.387e-07  4.205e-02    0.00      1    
## fmrides2[, 2] -5.428e-01  4.210e-02   -12.89  <2e-16 ***
```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8409 on 398 degrees of freedom
## Multiple R-squared: 0.2946, Adjusted R-squared: 0.2929
## F-statistic: 166.3 on 1 and 398 DF, p-value: < 2.2e-16

# the two covariates are correlated with a correlation coefficient r = 0.54, shared variance R^2 = 0.29, and p-value < .0001

#3b
# testing whether the two models have been mean-centered
mean(fmrides2[,2])
## [1] 2.045e-06
mean(fmrides2[,1])
## [1] -1.34875e-06

# since both models have a mean value extremely close to zero, we can assume that they have been mean-centered

#4
# The percentage of shared variance between the covariates is ~30%, which is calculated in 3a as the R squared

#5
# We have chosen to perform a regression analysis using lm on participant 1's data in order to find correlations between data and the two predicted models

#5a
# fitting the model including the intercept to the participant data
inclIntercept<-lm(fmri2[,1] ~ fmrides2)
summary(inclIntercept)

##
## Call:
## lm(formula = fmri2[, 1] ~ fmrides2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.8673  -5.2246   0.4506   5.0971  20.0510

```

```

## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 867.3275   0.3557 2438.64 <2e-16 ***
## fmrides2V1    9.5823   0.4240   22.60 <2e-16 ***
## fmrides2V2    8.9265   0.4240   21.05 <2e-16 ***  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.113 on 397 degrees of freedom
## Multiple R-squared:  0.6097, Adjusted R-squared:  0.6077 
## F-statistic: 310.1 on 2 and 397 DF,  p-value: < 2.2e-16

# the model explained a significant proportion of the variance found, R^2 = .61, F(2, 397) = 310, p < .0001.

#5b
# fitting the model excluding the intercept to the date
exIntercept<- lm(fmri2[,1] ~ 0+fmrides2)
summary(exIntercept)

## 
## Call:
## lm(formula = fmri2[, 1] ~ 0 + fmrides2)
## 
## Residuals:
##     Min      1Q Median      3Q     Max  
## 848.5  862.1  867.8  872.4  887.4  
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## fmrides2V1    9.582    51.832   0.185   0.853    
## fmrides2V2    8.928    51.831   0.172   0.863    
## 
## Residual standard error: 869.5 on 398 degrees of freedom
## Multiple R-squared:  0.0001043, Adjusted R-squared: -0.00492 
## F-statistic: 0.02075 on 2 and 398 DF,  p-value: 0.9795

# the model did not explain a significant proportion of the variance, R^2 = -.0049 F(2, 398) = 0.0208, p-value > 0.9

#5c
# fitting the first covariate as a model
firstCov<-lm(fmri2[,1] ~ fmrides2[,1])
summary(firstCov)

```

```

## 
## Call:
## lm(formula = fmri2[, 1] ~ fmrides2[, 1])
## 
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -27.8865 -6.8788  0.8642  7.1143 25.2019 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 867.3275    0.5168 1678.388 <2e-16 ***
## fmrides2[, 1] 4.7369    0.5174   9.155 <2e-16 ***  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 10.34 on 398 degrees of freedom
## Multiple R-squared:  0.174, Adjusted R-squared:  0.1719 
## F-statistic: 83.81 on 1 and 398 DF, p-value: < 2.2e-16

# the model using the first covariate explained a significant proportion of the variance found, R^2 = .17, F(1, 398) = 83.8, p < .0001.

```

#5d

```

# fitting the second covariate as a model
secondCov<-lm(fmri2[,1] ~ fmrides2[,2])
summary(secondCov)

## 
## Call:
## lm(formula = fmri2[, 1] ~ fmrides2[, 2])
## 
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -28.6493 -6.3764  0.8989  7.5855 25.3560 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 867.3275    0.5371 1614.768 < 2e-16 ***
## fmrides2[, 2] 3.7251    0.5378   6.927 1.74e-11 ***  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 10.74 on 398 degrees of freedom
## Multiple R-squared:  0.1076, Adjusted R-squared:  0.1053 
## F-statistic: 47.98 on 1 and 398 DF, p-value: 1.739e-11

```

```
# the model using the second covariate explained a significant proportion of the variance found, R^2 = .11, F(1, 398) = 48.0, p < .0001.
```

```
#5e
```

```
# fitting the 2nd covariate to the residuals from analysis of 5.c (the 1st covariate only analysis)
```

```
# finding residuals
```

```
residualsfirstCov<- residuals(firstCov)
```

```
# fitting the first model to the residuals
```

```
type3ss<-lm(residualsfirstCov~ fmrides2[,2])
```

```
summary(type3ss)
```

```
##
```

```
## Call:
```

```
## lm(formula = residualsfirstCov ~ fmrides2[, 2])
```

```
##
```

```
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
```

```
## -23.0729 -5.3106  0.9192  5.8245 20.3670
```

```
##
```

```
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -1.288e-05 4.095e-01  0.00     1
```

```
## fmrides2[, 2] 6.296e+00 4.100e-01 15.36 <2e-16 ***
```

```
## ---
```

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

```
##
```

```
## Residual standard error: 8.19 on 398 degrees of freedom
```

```
## Multiple R-squared:  0.3721, Adjusted R-squared:  0.3705
```

```
## F-statistic: 235.8 on 1 and 398 DF,  p-value: < 2.2e-16
```

```
# the model explained a significant proportion of the variance found, R^2 = .37, F(1, 398) = 236, p < .0001.
```

```
#5f
```

```
# fitting the 1st covariate to the residuals from 5.d. (the 2nd covariate only analysis)
```

```
# finding residuals
```

```
residualsSecondCov<- residuals(secondCov)
```

```
# fitting the second model to the residuals
```

```
secondType3ss<-lm(residualsSecondCov~ fmrides2[,1])
```

```
summary(secondType3ss)
```

```

## 
## Call:
## lm(formula = residualsSecondCov ~ fmrides2[, 1])
## 
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.3301  -5.1474   0.5896   5.9121  20.6890
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 9.116e-06 4.171e-01   0.00     1    
## fmrides2[, 1] 6.759e+00 4.177e-01  16.18 <2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 8.343 on 398 degrees of freedom
## Multiple R-squared:  0.3969, Adjusted R-squared:  0.3953 
## F-statistic: 261.9 on 1 and 398 DF,  p-value: < 2.2e-16

# the model explained a significant proportion of the variance found, R^2 = .40, F(1, 398) = 262, p < .0001.

```

#5g

```

# checking the impact of order by running two regression analysis with reversed order of predictors
newWorldOrder<-lm(fmri2[,1] ~ fmrides2[,1]+fmrides2[,2])
summary(newWorldOrder)

## 
## Call:
## lm(formula = fmri2[, 1] ~ fmrides2[, 1] + fmrides2[, 2])
## 
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.8673  -5.2246   0.4506   5.0971  20.0510
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 867.3275    0.3557 2438.64 <2e-16 ***
## fmrides2[, 1]  9.5823    0.4240   22.60 <2e-16 ***
## fmrides2[, 2]  8.9265    0.4240   21.05 <2e-16 *** 
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.113 on 397 degrees of freedom
## Multiple R-squared:  0.6097, Adjusted R-squared:  0.6077 
## F-statistic: 310.1 on 2 and 397 DF,  p-value: < 2.2e-16

```

```

newWorldOrder2<-lm(fmri2[,1] ~ fmrides2[,2]+fmrides2[,1])
summary(newWorldOrder2)

##
## Call:
## lm(formula = fmri2[, 1] ~ fmrides2[, 2] + fmrides2[, 1])
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -18.8673 -5.2246  0.4506  5.0971 20.0510 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 867.3275    0.3557 2438.64 <2e-16 ***
## fmrides2[, 2] 8.9265    0.4240   21.05 <2e-16 ***
## fmrides2[, 1] 9.5823    0.4240   22.60 <2e-16 ***  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.113 on 397 degrees of freedom
## Multiple R-squared:  0.6097, Adjusted R-squared:  0.6077 
## F-statistic: 310.1 on 2 and 397 DF, p-value: < 2.2e-16

# we see that order does not matter, since the estimates are the same across the two tests. This tells us that Lm uses the Type 3 method when deciding which covariate to fit to the model first.

```

Group level analyses

```

#6
# fitting the model to each of 37 participants and extracting the coefficients for each participant
fullModel <- lm(fmri2 ~ fmrides2)
fullModel$coefficients[2, ]

##          V1          V2          V3          V4          V5          V6          V7 
## 9.582330  5.773264  7.340939  2.682228  5.872990  3.209499  2.606846 
##          V8          V9          V10         V11         V12         V13         V14 
## 3.828594  2.422324  6.400593  6.989003  3.464375  4.626842  3.304558 
##          V15         V16         V17         V18         V19         V20         V21 
## 3.566603  5.566010  3.756718  5.436519  2.814987  2.829585  7.351865 
##          V22         V23         V24         V25         V26         V27         V28 
## 5.963731  5.827429  3.880726  6.534162  4.849484  4.997005  7.818241 
##          V29         V30         V31         V32         V33         V34         V35 
## 5.474936  6.648170  5.056695  4.891110  4.526764  4.904306  4.095500 
##          V36         V37 
## 10.521195 4.760363 

fullModel$coefficients[3, ]

```

```

##      V1      V2      V3      V4      V5      V6      V7
## 8.926516 6.104578 8.233871 3.133700 6.041478 3.230518 2.499852
##      V8      V9      V10     V11     V12     V13     V14
## 3.530697 0.992906 7.207420 6.538979 3.385156 3.595325 3.914555
##      V15     V16     V17     V18     V19     V20     V21
## 3.495208 5.802205 3.873774 5.276158 2.626577 2.658036 6.394320
##      V22     V23     V24     V25     V26     V27     V28
## 5.905760 5.165241 4.050404 6.399795 4.703407 3.951056 7.767283
##      V29     V30     V31     V32     V33     V34     V35
## 5.903824 6.628171 5.756162 4.607988 6.001619 3.930084 4.176857
##      V36     V37
## 10.631068 4.646695

#6a
# performing a t-test to identify if the set of coefficients from each covariate is different from zero across the entire group
# model 1
t.test(fullModel$coefficients[2, ], mu = 0)

##
## One Sample t-test
##
## data: fullModel$coefficients[2, ]
## t = 16.607, df = 36, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 4.512224 5.767586
## sample estimates:
## mean of x
## 5.139905

# the coefficients for model 1 are significantly different from zero t(36) = 16.6, p < .0001

# model 2
t.test(fullModel$coefficients[3, ], mu = 0)

##
## One Sample t-test
##
## data: fullModel$coefficients[3, ]
## t = 15.603, df = 36, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 4.413274 5.731982
## sample estimates:
## mean of x
## 5.072628

```

```

# the coefficients for model 2 are significantly different from zero t(36) = 15.6,
p < .0001

#6b
# testing the hypothesis that the contrast is different from zero across participants
# extracting the coefficients
coeffs <- fullModel$coefficients

# creating dataframe of the transposed coefficients
coeffs <- as.data.frame(t(coeffs))

# creating a contrast by subtraction of one model from the other
coeffs$contrast <- coeffs$fmrideres2V1 - coeffs$fmrideres2V2

# reshaping the wide-format dataframe to Long-format data using melt
coeffsAndContrast <- melt(coeffs)

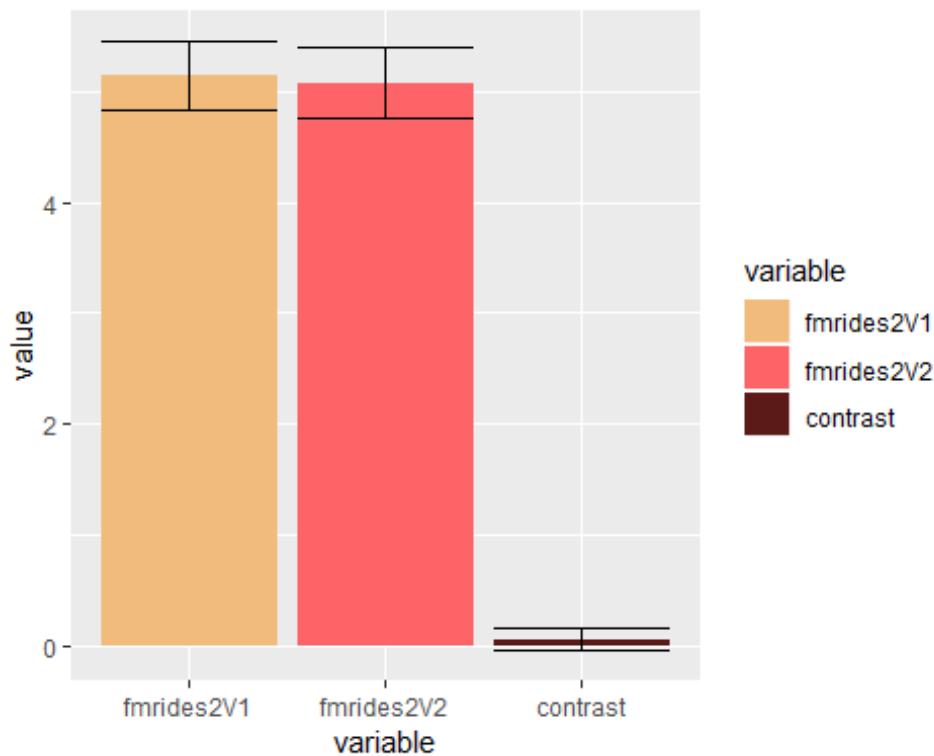
## No id variables; using all as measure variables

#### REMEMBER TO TEST

#6c
# making a bar diagram including the mean effect of the two coefficients and the contrast
p1 <- coeffsAndContrast %>%
  filter(!variable == "(Intercept)") %>%

  ggplot(aes(x = variable, y = value, fill = variable)) +
  geom_bar(stat = "summary", fun.y = mean) +
  geom_errorbar(stat = "summary", fun.data = mean_se) + scale_fill_manual(values =
  wes_palette("GrandBudapest1"))
p1

```



Adding a covariate

```
#7
# adding a covariate to each participant that models the effect of time
# create a simple model of time
list <- c(1:400)

# fitting the data to the model using time as a covariate
fullModel2 <- lm(fmri2 ~ fmrides2 + list)

# extracting coefficients
fullModel2$coefficients[2, ]

##          V1          V2          V3          V4          V5          V6          V7
## 9.550016  5.870281  7.061170  3.272271  5.883448  3.338111  2.612759
##          V8          V9          V10         V11         V12         V13         V14
## 4.702847  2.884405  6.802563  6.988008  4.502570  5.098470  3.340289
##          V15         V16         V17         V18         V19         V20         V21
## 3.746568  5.774115  3.529123  5.488261  3.304290  3.344263  7.381207
##          V22         V23         V24         V25         V26         V27         V28
## 5.800563  5.845666  3.809432  6.848006  5.358663  4.765501  7.636024
##          V29         V30         V31         V32         V33         V34         V35
## 5.468770  6.740166  5.121437  5.619891  4.423846  4.617687  4.312931
##          V36         V37
## 10.229020  4.474234

fullModel2$coefficients[3, ]
```

```

##      V1      V2      V3      V4      V5      V6      V7
## 8.884649 6.230279 7.871389 3.898191 6.055029 3.397156 2.507513
##      V8      V9      V10     V11     V12     V13     V14
## 4.663425 1.591603 7.728234 6.537690 4.730295 4.206392 3.960850
##      V15     V16     V17     V18     V19     V20     V21
## 3.728380 6.071837 3.578890 5.343197 3.260543 3.324880 6.432338
##      V22     V23     V24     V25     V26     V27     V28
## 5.694350 5.188871 3.958032 6.806427 5.363126 3.651108 7.531194
##      V29     V30     V31     V32     V33     V34     V35
## 5.895835 6.747366 5.840045 5.552235 5.868273 3.558725 4.458572
##      V36     V37
## 10.252511 4.275971

#7a
# testing whether the coefficients are significantly different from zero
t.test(fullModel2$coefficients[2, ], mu = 0)

##
## One Sample t-test
##
## data: fullModel2$coefficients[2, ]
## t = 18.3, df = 36, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 4.699321 5.870780
## sample estimates:
## mean of x
## 5.285051

# the coefficients for model 1 are significantly different from zero, t(36) = 18.3
# , p < .0001. The model is improved by factoring out the effect of time

t.test(fullModel2$coefficients[3, ], mu = 0)

##
## One Sample t-test
##
## data: fullModel2$coefficients[3, ]
## t = 17.609, df = 36, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 4.654803 5.866570
## sample estimates:
## mean of x
## 5.260686

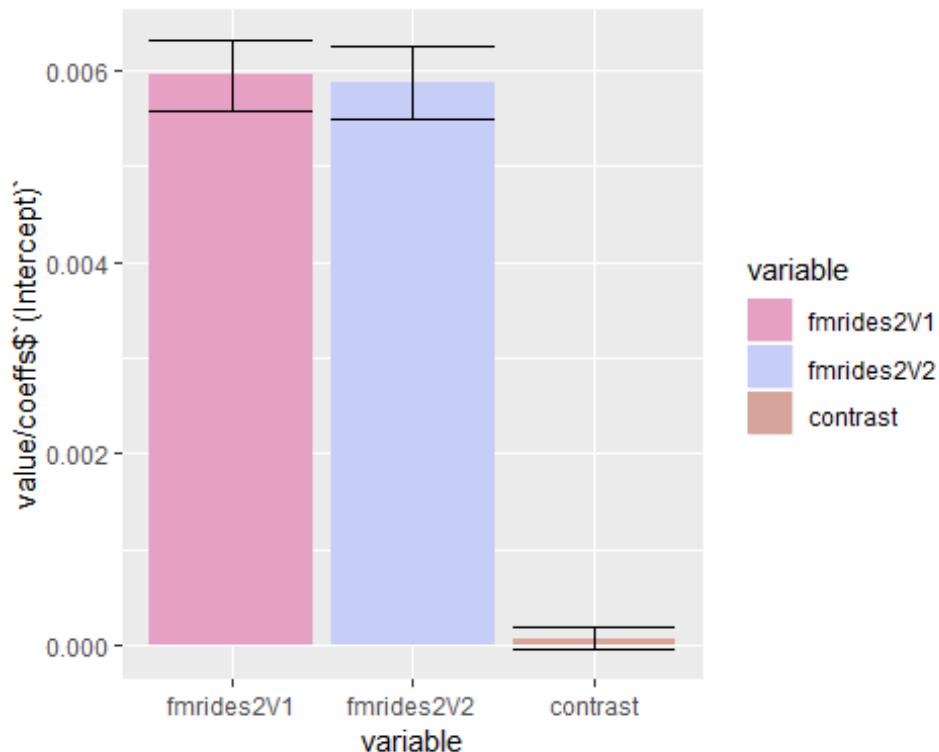
# the coefficients for model 2 are significantly different from zero, t(36) = 17.6
# , p < .0001. The model is thus improved by factoring out the effect of time.

```

```

#8
# creating a bar diagram with the bar displaying effects as percent signal change
p2 <- coeffsAndContrast %>%
filter(!variable == "(Intercept)") %>%
ggplot(aes(
x = variable,
y = value / coeffs$`"(Intercept)``,
fill = variable
)) +
geom_bar(stat = "summary", fun.y = mean) +
geom_errorbar(stat = "summary", fun.data = mean_se) +
scale_fill_manual(values = wes_palette("GrandBudapest2"))
p2

```



Portfolio 4

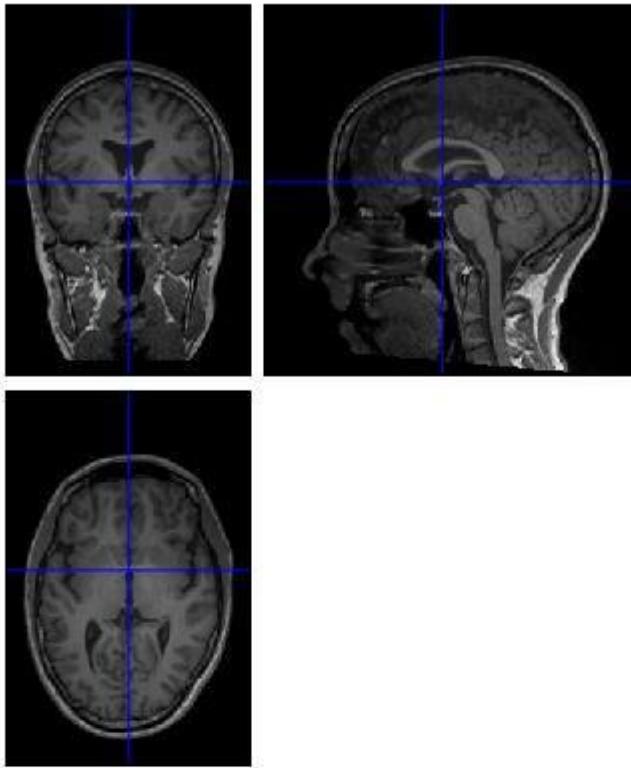
Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad

4 mar 2019

```
# Load Libraries into R
library(pacman)
p_load(tidyverse, dplyr)

# set working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Portfolios/4")

# 1a
# insert structural image
# translation: x(right) = 0.57864, y(forward) = 0.27727, z(up) = - 0.36372, rotation: pitch = -0.1, roll = 0.01, yaw = 0.04
```



Crosshair Position		Origin
mm:	0.0 0.0 0.0	
vx:	121.8 134.0 88.7	
Intensity:	335.965	
right {mm}	0.57864	
forward {mm}	0.27727	
up {mm}	-0.36372	
pitch {rad}	-0.1	
roll {rad}	0.01	
yaw {rad}	0.04	
resize {x}	1	
resize {y}	1	
resize {z}	1	
Set Origin	Reorient...	

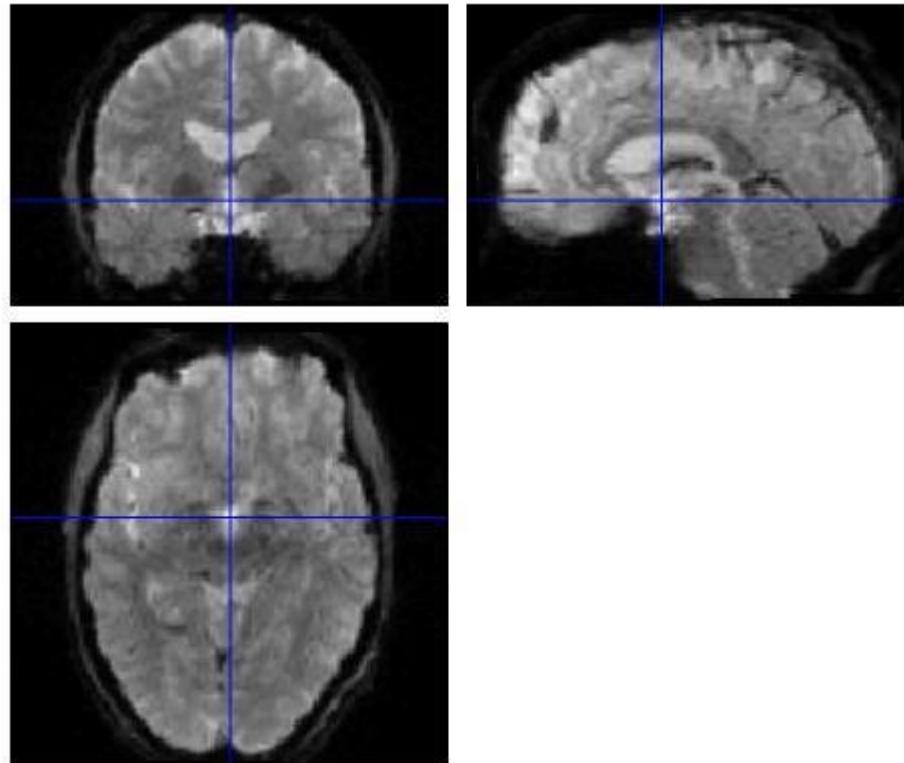
File: ..003-00001-000176-01.nii
Dimensions: 256 x 256 x 176
Datatype: uint16
Intensity: Y = 1 X
3T 3D GRIR

Vox size: -0.977 x 0.977 x 1
Origin: 122 134 88.7
Dir Cos: 0.013 0.001 1.000
0.996 -0.090 -0.013
0.090 0.996 -0.002

Full Volume	Hide Crosshair
World Space	Trilinear in...
Auto Wind...	Add Overlay...

```
# 1b
# insert functional image
```

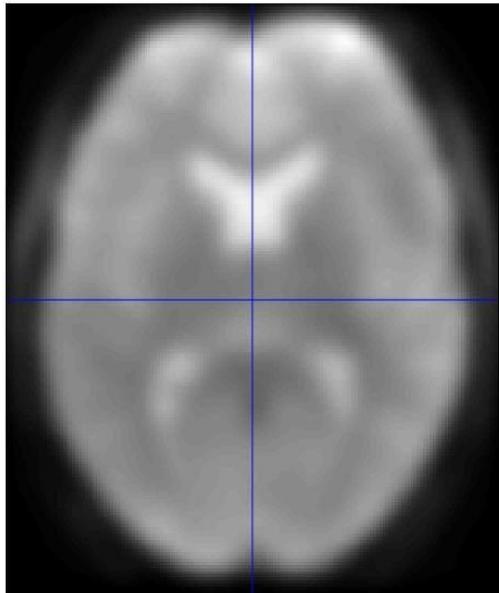
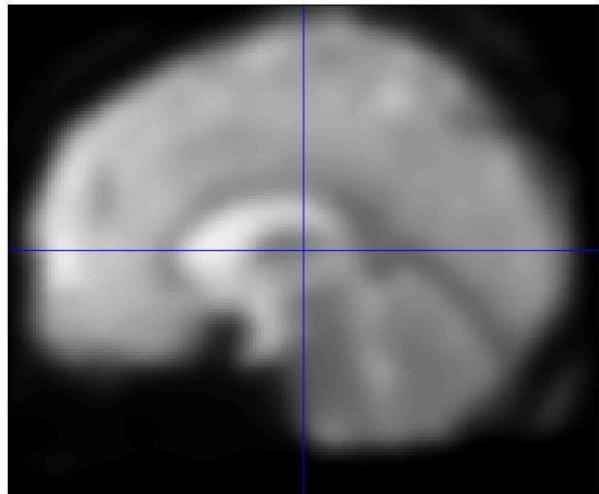
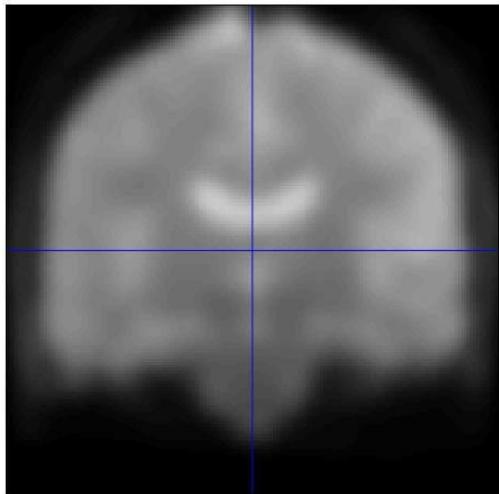
```
# translation: x(right) = -2.237, y(forward) = 21.6177, z(up) = -18.4874, rotation  
: pitch = -0.01, roll = -0.05, yaw = -0.01
```



Crosshair Position	Origin
mm:	-0.7 -0.9 0.0
vx:	47.7 54.8 23.0
Intensity:	1691.73
right {mm}	-2.237
forward {mm}	21.6177
up {mm}	-18.4874
pitch {rad}	-0.01
roll {rad}	-0.05
yaw {rad}	-0.01
resize {x}	1
resize {y}	1
resize {z}	1
<input type="button" value="Set Origin"/>	<input type="button" value="Reorient..."/>

File: ..005-00001-000001-01.nii
Dimensions: 96 x 96 x 65
Datatype: uint16
Intensity: Y = 1 X
3T 2D EP TR=3500ms/TE=27ms/FA=90deg
Vox size: -2 x 2 x 2
Origin: 48 55.2 23.1
Dir Cos: 1.000 0.014 -0.002
-0.014 1.000 0.028
0.003 -0.028 1.000
<input type="button" value="Full Volume"/> <input type="button" value="Hide Crosshair"/>
<input type="button" value="World Space"/> <input type="button" value="Trilinear in..."/>
<input type="button" value="Auto Wind..."/> <input type="button" value="Add Overlay..."/>

```
# 2a  
# realignment performed in Matlab using SPM  
  
# 2b  
# coregistration performed in Matlab using SPM  
  
# 2c  
# segmentation performed in Matlab using SPM  
  
# 2d  
# normalization performed in Matlab using SPM  
  
# 2e  
# smoothing performed in Matlab using SPM  
  
# 2f  
# report output from the graphics window as well as an example of a smoothed functional image
```



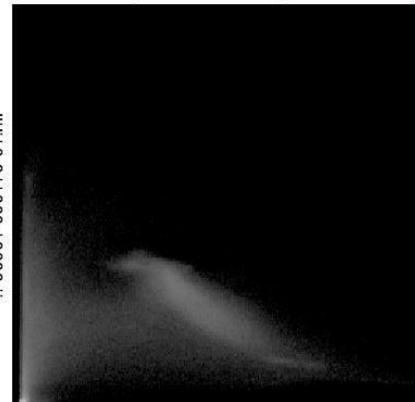
Normalised Mutual Information Coregistration

$$X1 = 0.000*X + 0.000*Y - 0.500*Z + 92.876$$

$$Y1 = -0.488*X - 0.004*Y - 0.000*Z + 117.188$$

$$Z1 = -0.004*X + 0.488*Y + 0.000*Z - 40.260$$

Original Joint Histogram



..-00001-00001-01.nii

Final Joint Histogram



..-00001-00001-01.nii

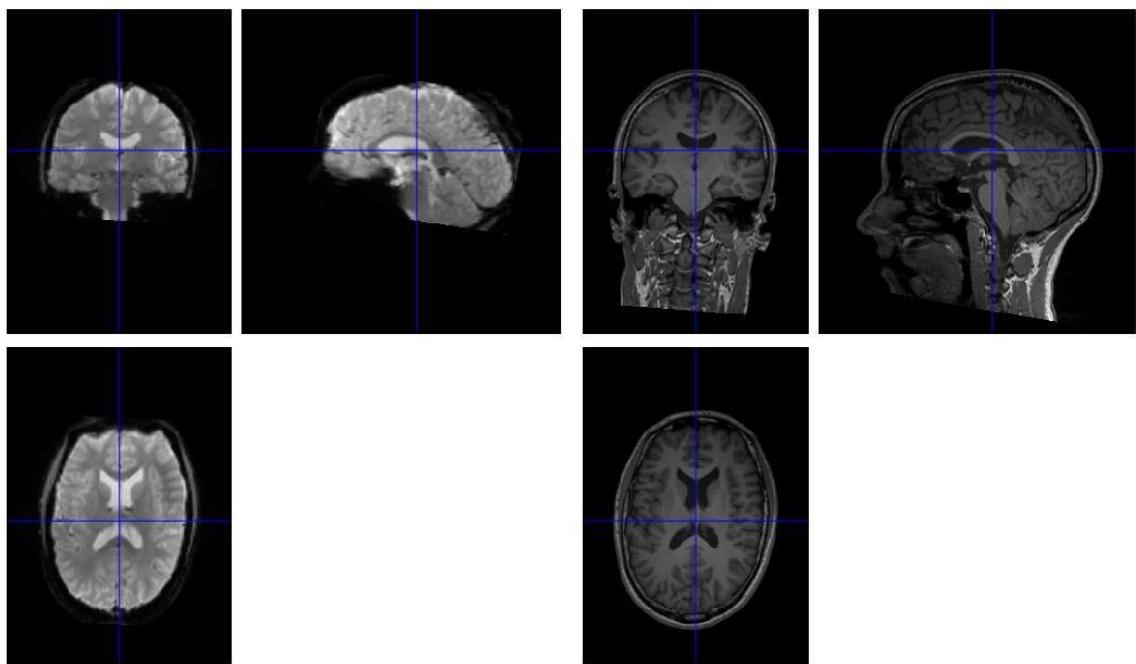
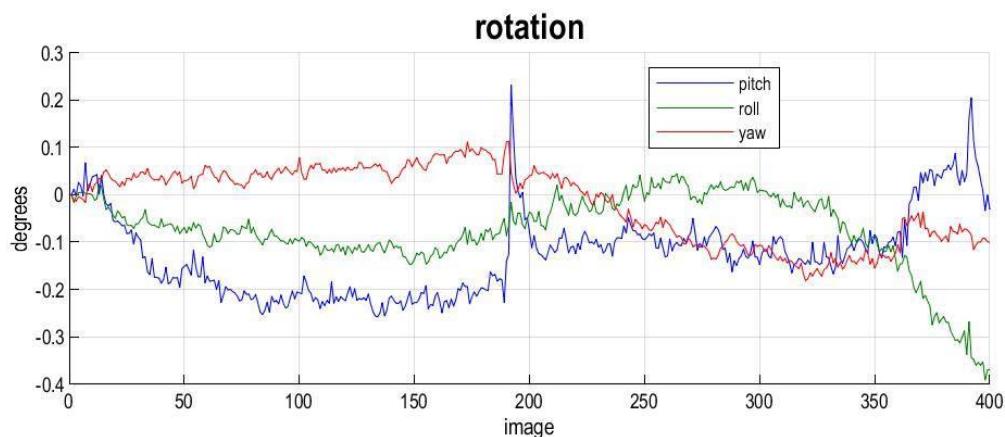
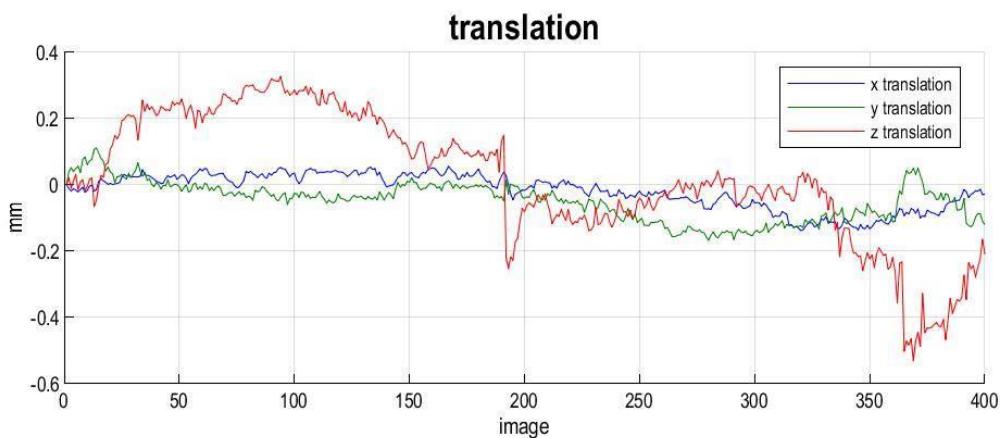


Image realignment

```
1 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00001-000001-01.nii,1
2 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00002-000002-01.nii,1
3 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00003-000003-01.nii,1
4 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00004-000004-01.nii,1
5 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00005-000005-01.nii,1
6 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00006-000006-01.nii,1
7 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00007-000007-01.nii,1
8 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00008-000008-01.nii,1
9 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00009-000009-01.nii,1
10 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00010-000010-01.nii,1
11 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00011-000011-01.nii,1
12 C:\Users\Theal\Desktop\fmri_data_raw\fmri_data\SubjectNo0001-0005-00012-000012-01.nii,1
..... etc
```

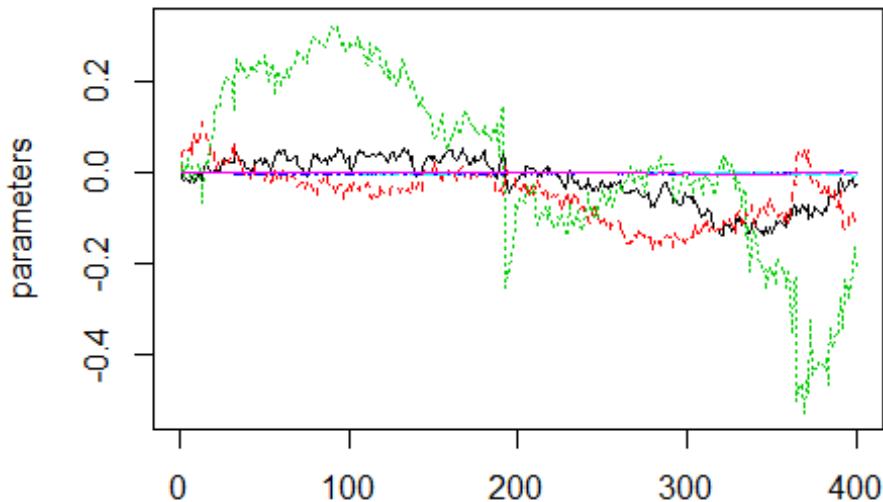


```

# 3
# read parameters into a R dataframe
parameters <- read.delim("rp_fSubjectNo0001-0005-0001-000001-01.txt", header = F,
sep = "")

# 3a
# make a lineplot of the realignment parameters
matplot(x = parameters, type = "l")

```



```

# 3b
# calculate the amount of participant movement in each direction
# find range for each parameter
df <- apply(parameters, MARGIN = 2, FUN = range)
# make a dataframe with max and min values
df <- as.data.frame(df)
# x-axis
sum(df$V1)

## [1] -0.08232204

# y-axis
sum(df$V2)

## [1] -0.05654394

# z-axis
sum(df$V3)

## [1] -0.2039516

```

```

# pitch
sum(df$V4)

## [1] -0.0004695396

# roll
sum(df$V5)

## [1] -0.006053881

# yaw
sum(df$V6)

## [1] -0.001209178

# 3c
# test correlation between realignment parameters and the fMRI model
# Load fMRI model
fmriModel <- read.csv("portfolio_assignment3_aud_fmri_design.csv", header = F)

# make a correlation matrix
cor(parameters, fmriModel)

##          V1          V2
## V1 -0.08179962 -0.01108979
## V2  0.01019338  0.01983231
## V3  0.02451261  0.04212546
## V4 -0.09122584  0.01022746
## V5  0.17216510  0.02404600
## V6 -0.02957477  0.02682083

# find p-values for correlation between model 1 and the realignment parameters
# x translation
cor.test(fmriModel$V1, parameters$V1)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and parameters$V1
## t = -1.6374, df = 398, p-value = 0.1023
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1784203  0.0163835
## sample estimates:
##          cor
## -0.08179962

# y translation
cor.test(fmriModel$V1, parameters$V2)

```

```

## 
## Pearson's product-moment correlation
## 
## data: fmriModel$V1 and parameters$V2
## t = 0.20337, df = 398, p-value = 0.839
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.08794624 0.10813702
## sample estimates:
##       cor
## 0.01019338

# z translation
cor.test(fmriModel$V1, parameters$V3)

## 
## Pearson's product-moment correlation
## 
## data: fmriModel$V1 and parameters$V3
## t = 0.48917, df = 398, p-value = 0.625
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.07371628 0.12227046
## sample estimates:
##       cor
## 0.02451261

# pitch
cor.test(fmriModel$V1, parameters$V4)

## 
## Pearson's product-moment correlation
## 
## data: fmriModel$V1 and parameters$V4
## t = -1.8276, df = 398, p-value = 0.06836
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.187599510 0.006887487
## sample estimates:
##       cor
## -0.09122584

# roll
cor.test(fmriModel$V1, parameters$V5)

## 
## Pearson's product-moment correlation
## 
## data: fmriModel$V1 and parameters$V5

```

```

## t = 3.4867, df = 398, p-value = 0.0005434
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.07538598 0.26573100
## sample estimates:
##      cor
## 0.1721651

# yaw
cor.test(fmriModel$V1, parameters$V6)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and parameters$V6
## t = -0.59027, df = 398, p-value = 0.5553
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1272575 0.0686761
## sample estimates:
##      cor
## -0.02957477

# there is a correlation between model 1 and realignment parameter 5 (roll), p-value < .0005. This is highly problematic for the further advancement of the study.

# find p-values for correlation between model 2 and the realignment parameters
# x translation
cor.test(fmriModel$V2, parameters$V1)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V1
## t = -0.22125, df = 398, p-value = 0.825
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.10902296 0.08705659
## sample estimates:
##      cor
## -0.01108979

# y translation
cor.test(fmriModel$V2, parameters$V2)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V2

```

```

## t = 0.39573, df = 398, p-value = 0.6925
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.07837181  0.11765524
## sample estimates:
##      cor
## 0.01983231

# z translation
cor.test(fmriModel$V2, parameters$V3)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V3
## t = 0.84115, df = 398, p-value = 0.4008
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.05615822  0.13960057
## sample estimates:
##      cor
## 0.04212546

# pitch
cor.test(fmriModel$V2, parameters$V4)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V4
## t = 0.20405, df = 398, p-value = 0.8384
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.08791242  0.10817071
## sample estimates:
##      cor
## 0.01022746

# roll
cor.test(fmriModel$V2, parameters$V5)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V5
## t = 0.47985, df = 398, p-value = 0.6316
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.07418062  0.12181052

```

```

## sample estimates:
##      cor
## 0.024046

# yaw
cor.test(fmriModel$V2, parameters$V6)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and parameters$V6
## t = 0.53527, df = 398, p-value = 0.5928
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.0714187 0.1245450
## sample estimates:
##      cor
## 0.02682083

# model 2 is not correlated with any of the realignment transformations

# 3d
# make a lineplot of the realignment parameters with time removed
# make a vector modelling time
timeVec <- c(1:400)

# create a model to calculate time's influence on the first align parameter, x
model1 <- lm(parameters[,1] ~ timeVec)
summary(model1)

##
## Call:
## lm(formula = parameters[, 1] ~ timeVec)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.076484 -0.018485  0.003025  0.024665  0.076150
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.337e-02 3.269e-03 16.32 <2e-16 ***
## timeVec     -3.588e-04 1.413e-05 -25.39 <2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03263 on 398 degrees of freedom

```

```

## Multiple R-squared:  0.6184, Adjusted R-squared:  0.6174
## F-statistic: 644.9 on 1 and 398 DF,  p-value: < 2.2e-16

# create a model to calculate time's influence on the second align parameter, y
model2 <- lm(parameters[,2] ~ timeVec)
summary(model2)

##
## Call:
## lm(formula = parameters[, 2] ~ timeVec)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -0.091987 -0.025229 -0.004282  0.025041  0.156428 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.878e-02 4.329e-03  4.338 1.82e-05 ***
## timeVec     -3.366e-04 1.871e-05 -17.993 < 2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.04321 on 398 degrees of freedom
## Multiple R-squared:  0.4486, Adjusted R-squared:  0.4472 
## F-statistic: 323.7 on 1 and 398 DF,  p-value: < 2.2e-16

# create a model to calculate time's influence on the third align parameter, z
model3 <- lm(parameters[,3] ~ timeVec)
summary(model3)

##
## Call:
## lm(formula = parameters[, 3] ~ timeVec)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -0.33897 -0.05861  0.02330  0.09263  0.18215 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.909e-01 1.163e-02  25.01  <2e-16 ***
## timeVec     -1.358e-03 5.027e-05 -27.02  <2e-16 *** 
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.1161 on 398 degrees of freedom
## Multiple R-squared:  0.6472, Adjusted R-squared:  0.6463 
## F-statistic: 730.1 on 1 and 398 DF,  p-value: < 2.2e-16

```

```

# create a model to calculate time's influence on the fourth align parameter, pitch
model4 <- lm(parameters[,4] ~ timeVec)
summary(model4)

##
## Call:
## lm(formula = parameters[, 4] ~ timeVec)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.0019592 -0.0010228 -0.0002650  0.0004361  0.0062621
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.313e-03  1.386e-04 -23.915   <2e-16 ***
## timeVec      5.666e-06  5.988e-07   9.461   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.001383 on 398 degrees of freedom
## Multiple R-squared:  0.1836, Adjusted R-squared:  0.1816
## F-statistic: 89.52 on 1 and 398 DF,  p-value: < 2.2e-16

# create a model to calculate time's influence on the fifth align parameter, roll
model5 <- lm(parameters[,5] ~ timeVec)
summary(model5)

##
## Call:
## lm(formula = parameters[, 5] ~ timeVec)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.0048858 -0.0007453 -0.0001455  0.0011351  0.0022997
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.218e-04  1.394e-04  -5.179 3.55e-07 ***
## timeVec     -3.058e-06  6.023e-07  -5.077 5.91e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.001391 on 398 degrees of freedom
## Multiple R-squared:  0.06082, Adjusted R-squared:  0.05846
## F-statistic: 25.77 on 1 and 398 DF,  p-value: 5.909e-07

```

```

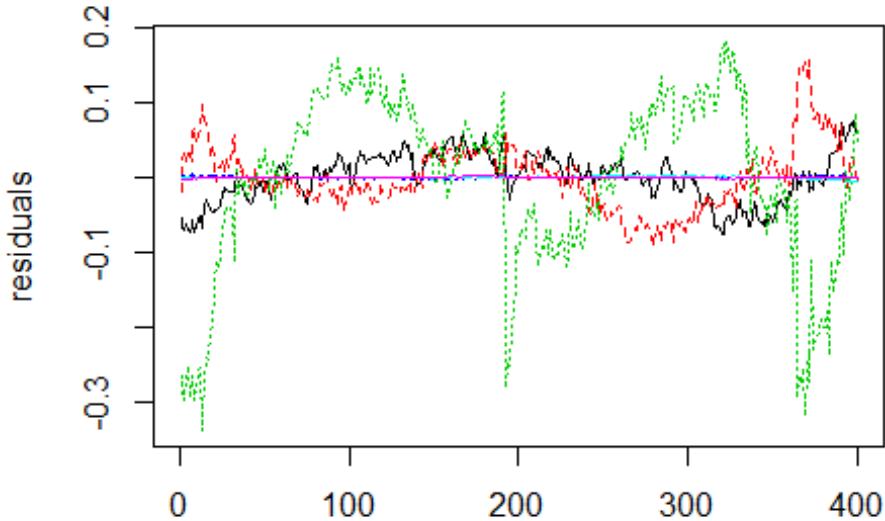
# create a model to calculate time's influence on the sixth align parameter, yaw
model6 <- lm(parameters[,6] ~ timeVec)
summary(model6)

##
## Call:
## lm(formula = parameters[, 6] ~ timeVec)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -1.797e-03 -6.762e-04 -6.832e-05  6.511e-04  2.173e-03
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.573e-03  8.675e-05   18.13   <2e-16 ***
## timeVec     -9.338e-06  3.749e-07  -24.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0008659 on 398 degrees of freedom
## Multiple R-squared:  0.6091, Adjusted R-squared:  0.6082
## F-statistic: 620.3 on 1 and 398 DF,  p-value: < 2.2e-16

# create a dataframe containing all residuals from the model summaries
residuals <- data.frame(x = residuals(model1), y = residuals(model2), z = residuals(model3),
                         pitch = residuals(model4), roll = residuals(model5), yaw = residuals(model6))

# generate a lineplot of realignment parameters with time removed, that is, a plot
# of the residuals left after fitting the model with time
matplot(x = residuals, type = "l")

```

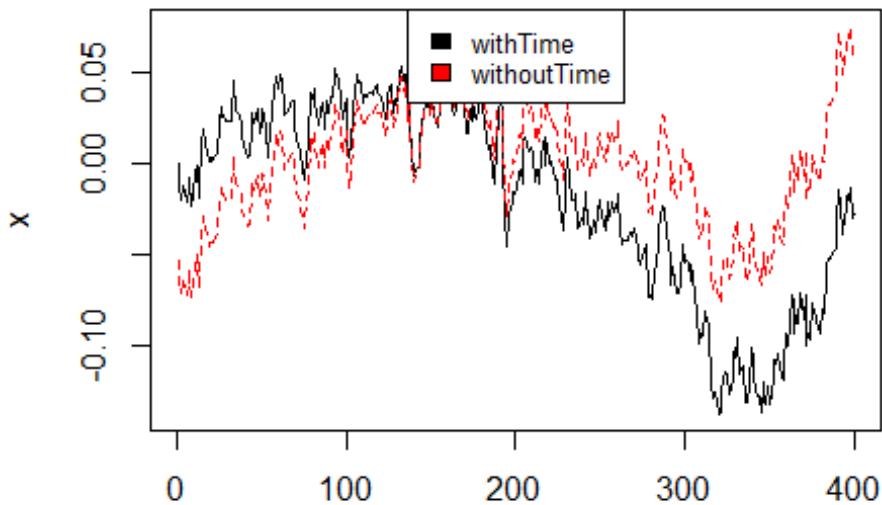


```

# 3e
# make a dataframe for the realignment parameter of the x-axis and the residuals after fitting time
x <- data.frame(withTime = parameters[,1],withoutTime = residuals[,1])

# make a Lineplot showcasing how the parameter looks with and without time. We use the parameter for the x-axis
matplotlib(x,type="l")
nn <- ncol(x)
legend("top", colnames(x),col=seq_len(nn),cex=0.8,fill=seq_len(nn))

```



```

# 3f
# calculate whether the realignment parameters (corrected for effects of time) correlate with the fMRI model 1
# x translation
cor.test(fmriModel$V1, residuals$x) #significant

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$x
## t = -3.2627, df = 398, p-value = 0.001199
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.25540787 -0.06436495
## sample estimates:
##       cor
## -0.1613981

# y translation
cor.test(fmriModel$V1, residuals$y)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$y

```

```

## t = -0.13582, df = 398, p-value = 0.892
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.10478967  0.09130477
## sample estimates:
##          cor
## -0.006807903

# z translation
cor.test(fmriModel$V1, residuals$z)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$z
## t = 0.20811, df = 398, p-value = 0.8352
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.08771031  0.10837199
## sample estimates:
##          cor
## 0.01043111

# pitch
cor.test(fmriModel$V1,residuals$pitch)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$pitch
## t = -1.8062, df = 398, p-value = 0.07164
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.186569378  0.007954924
## sample estimates:
##          cor
## -0.09016713

# roll
cor.test(fmriModel$V1,residuals$roll) #significant

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$roll
## t = 3.4803, df = 398, p-value = 0.0005562
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.07507145 0.26543698

```

```

## sample estimates:
##      cor
## 0.1718581

# yaw
cor.test(fmriModel$V1,residuals$yaw)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V1 and residuals$yaw
## t = -1.5151, df = 398, p-value = 0.1305
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.17249969 0.02248988
## sample estimates:
##      cor
## -0.07572884

# there is a significant negative correlation between movement on the x-axis and model 1, r(398) = -0.16, p-value < .05
# there is likewise a significant positive correlation between rotation on the "roll-axis" and model 1, r(398) = 0.17, p-value < .001

# calculate whether the realignment parameters (corrected for effects of time) correlate with the fMRI model 2
# x translation
cor.test(fmriModel$V2, residuals$x)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$x
## t = -1.833, df = 398, p-value = 0.06755
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.187859905 0.006617593
## sample estimates:
##      cor
## -0.09149349

# y translation
cor.test(fmriModel$V2, residuals$y)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$y

```

```

## t = -0.50684, df = 398, p-value = 0.6125
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.12314264  0.07283553
## sample estimates:
##      cor
## -0.02539758

# z translation
cor.test(fmriModel$V2, residuals$z)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$z
## t = -0.14616, df = 398, p-value = 0.8839
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.10530223  0.09079078
## sample estimates:
##      cor
## -0.007326154

# pitch
cor.test(fmriModel$V2, residuals$pitch)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$pitch
## t = 0.77301, df = 398, p-value = 0.44
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.05955963  0.13625265
## sample estimates:
##      cor
## 0.0387182

# roll
cor.test(fmriModel$V2, residuals$roll)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$roll
## t = 0.20172, df = 398, p-value = 0.8402
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.08802811  0.10805548

```

```

## sample estimates:
##          cor
## 0.01011088

# yaw
cor.test(fmriModel$V2,residuals$yaw)

##
## Pearson's product-moment correlation
##
## data: fmriModel$V2 and residuals$yaw
## t = -0.58321, df = 398, p-value = 0.5601
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.12690942  0.06902818
## sample estimates:
##          cor
## -0.02922132

# there is no significant correlation between the transformations/rotations and the second model

```

Portfolio 5

Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad

6 mar 2019

```
knitr::opts_chunk$set(echo = TRUE)

# 0. Preliminary tasks

# Load Libraries
library(pacman)
p_load(psych,lme4,rgl,plyr,grid,gridExtra,wesanderson,widyr,igraph,ggraph,SnowballC,RCurl,XML,RColorBrewer,tm,wordcloud,utf8,readtext,tidytext,scales,tidyverse,dplyr,tidyr,topicmodels,LDAvis,scales,reshape2,pastecs)

# set working directory
setwd("C:/Users/thram_000/OneDrive/cog data/exp-meth")

# 1. Checking input using R

#1a
# the purpose of this task is to determine whether there are significant differences between the durations of the two story types.

# define two vectors containing the duration data from each story type
durStory1 <- c(35, 27, 27, 36, 26, 16, 29, 42, 33, 54, 22, 38, 43, 43, 21)
durStory2 <- c(55, 33, 23, 37, 16, 30, 48, 40, 26, 24, 46, 27, 25, 27, 30)

# compare the two vectors using a parametric, Welch two sample t-test
t.test(durStory1, durStory2)

# the results of the t-test indicate no significant differences in the duration of story 1 and 2, t(28) = 0.09, p > .05.

##
## Welch Two Sample t-test
##
## data: durStory1 and durStory2
## t = 0.087292, df = 27.922, p-value = 0.9311
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -7.489661 8.156327
## sample estimates:
## mean of x mean of y
## 32.80000 32.46667
```

#1b

```
# the purpose of this task is to determine whether there are significant differences between the ratings of the two story types.

# define two vectors containing the rating data from each story type
ratingStory1 <- c(7, 4, 4, 5, 3, 1, 6, 2, 1, 4, 2, 3, 2, 4, 1)
ratingStory2 <- c(8, 2, 3, 2, 5, 7, 7, 3, 3, 2, 3, 3, 2, 4, 3)

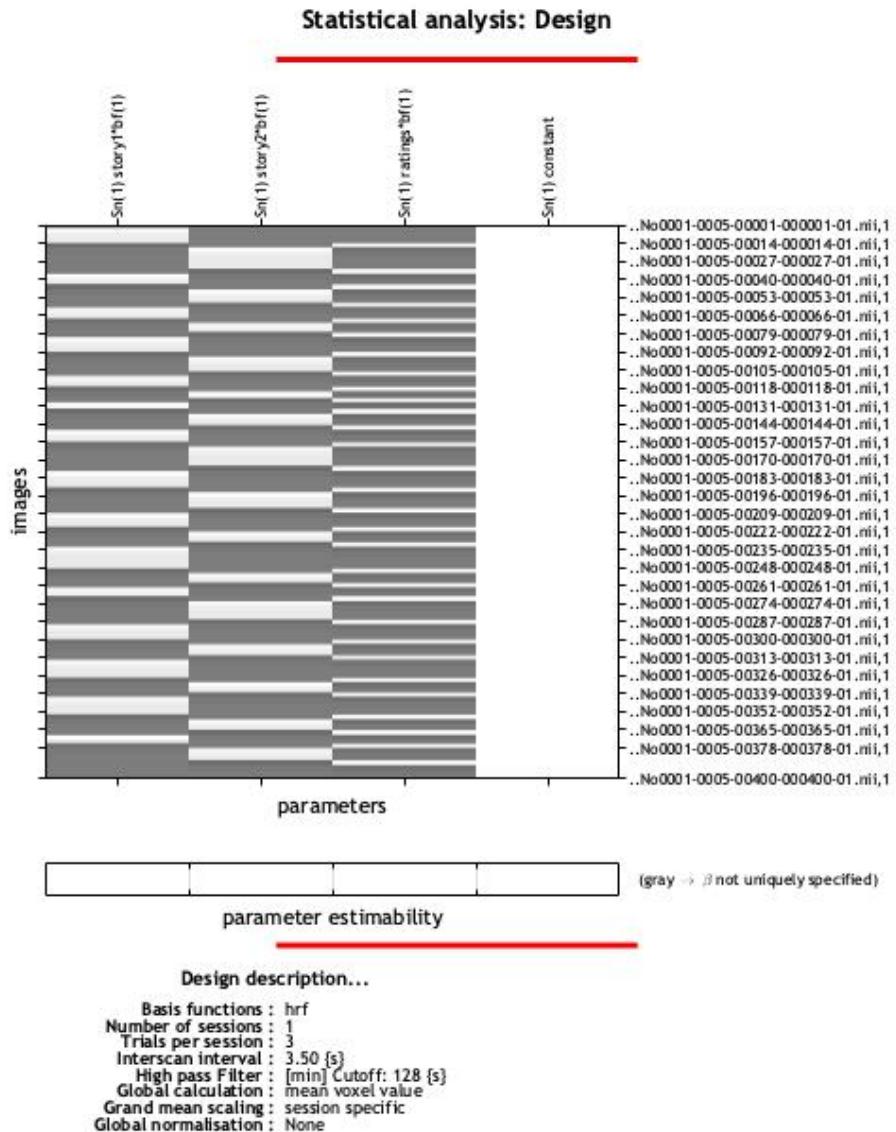
# compare the two vectors using a parametric, Welch two sample t-test
t.test(ratingStory1,ratingStory2)

# according to the t-test, the ratings of the two stories were not significantly different from each other, t(28) = -0.76, p > .05

## 
## Welch Two Sample t-test
##
## data: ratingStory1 and ratingStory2
## t = -0.76031, df = 27.767, p-value = 0.4535
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.9707779  0.9041112
## sample estimates:
## mean of x mean of y
## 3.266667  3.800000
```

2. Create the model in SPM

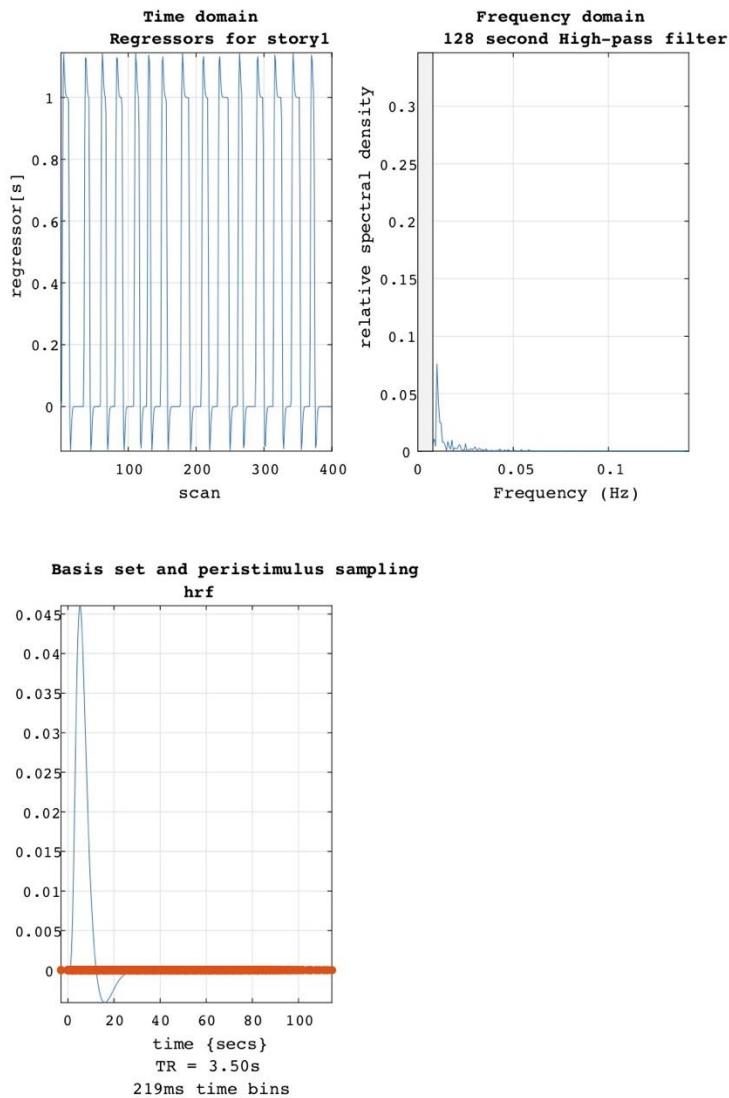
2.a The generated design matrix figure has 4 columns corresponding to onsets for Story 1, Story 2, ratings and a constant.



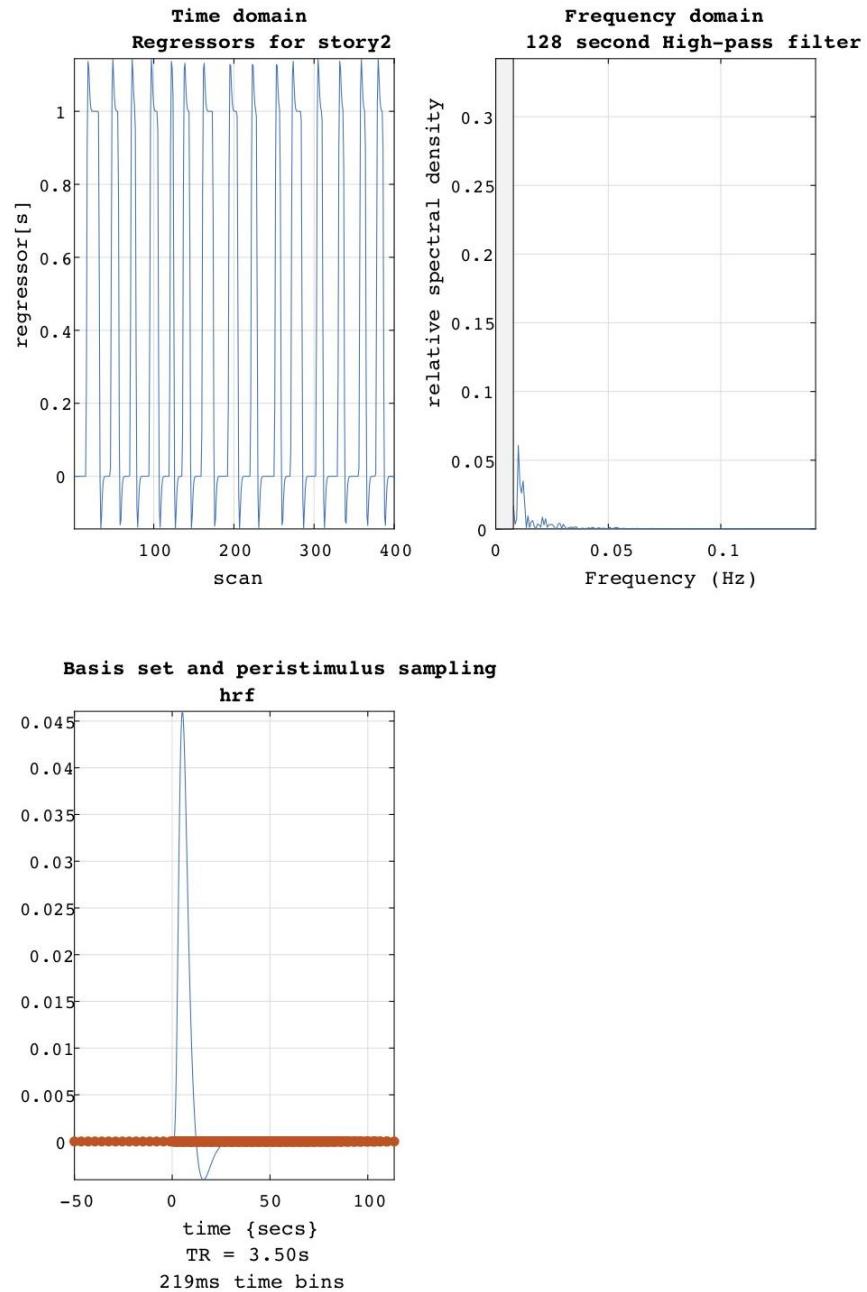
3. Checking the model

3.a Periodogram plots of the Frequency domain for the three conditions

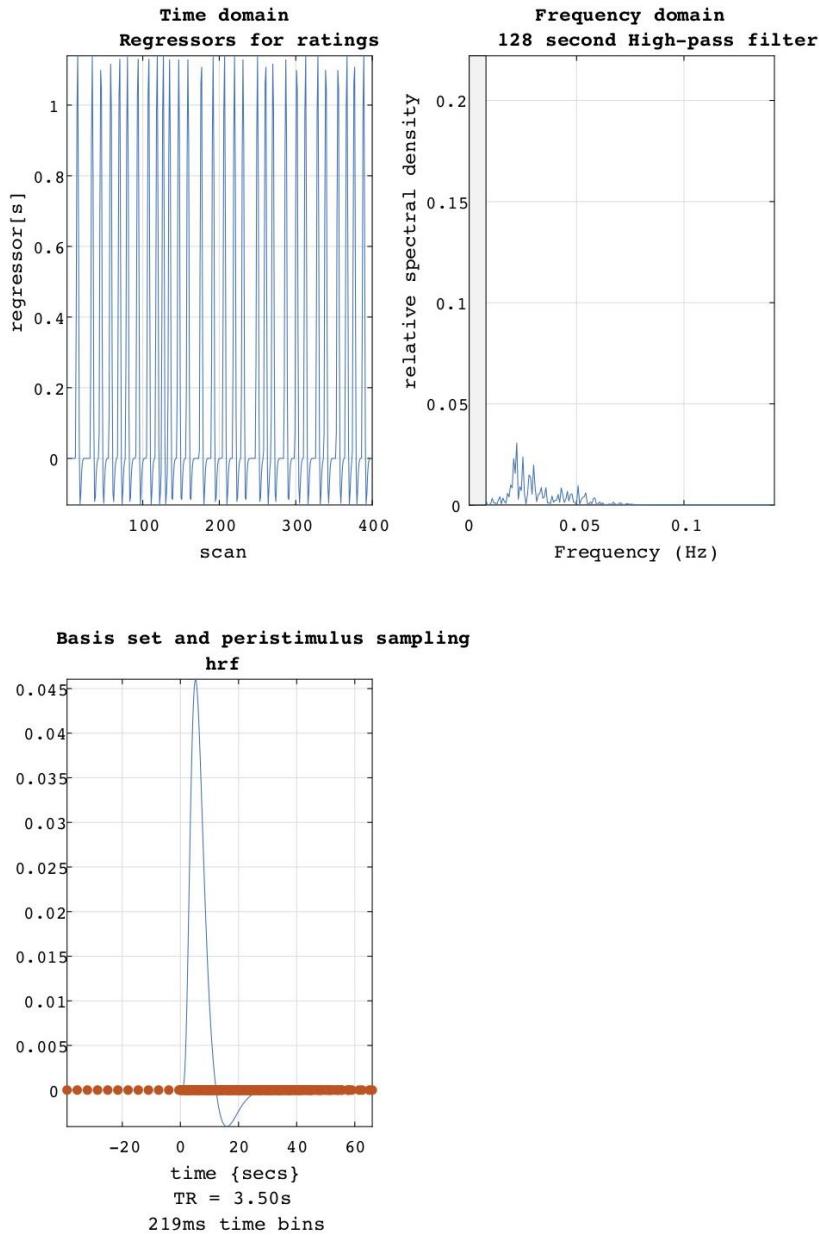
Story 1



Story 2



Ratings

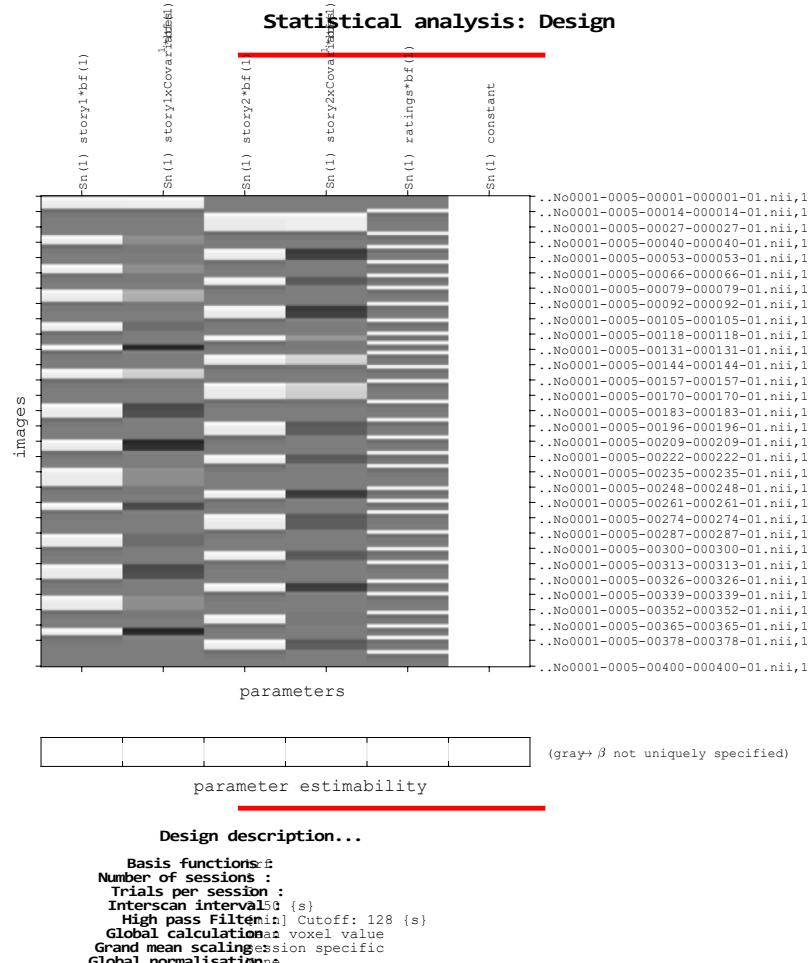


3.b. Eye-ball task: What are the most predominant frequencies for the three condition, as seen from these plots?

For story 1 and 2 the most predominant frequencies are around 0.005 Hz, while the most predominant frequency for ratings is about 0.02 Hz.

4. Adding covariates

4.a. After adding the covariates the generated design matrix figure has 6 columns, with columns 2 and 4 modeling the rating effects.



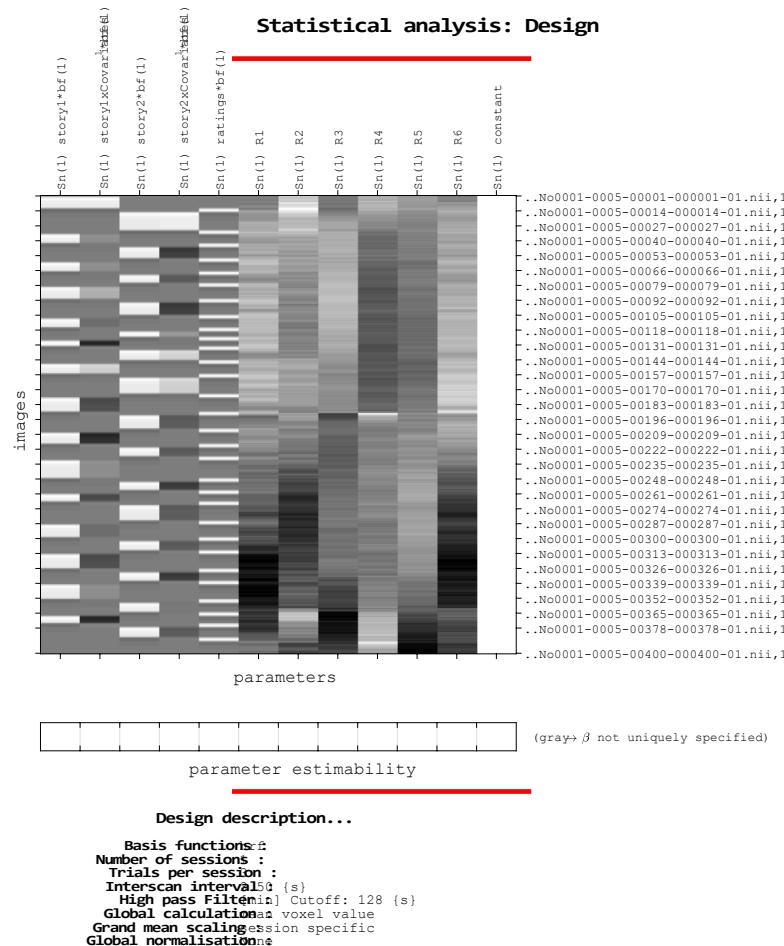
A parametric modulation in SPM is basically an interaction between the modeled response and the mean centered covariate (i.e. where the mean has been subtracted).

4.b. Interpretation task: Why is it important to subtract the mean?

You subtract the mean in order to mean center the data. If the data is mean centered, we are able to draw meaningful inferences from the intercept.

4.c. Model with the motion parameters from the realignment procedure.

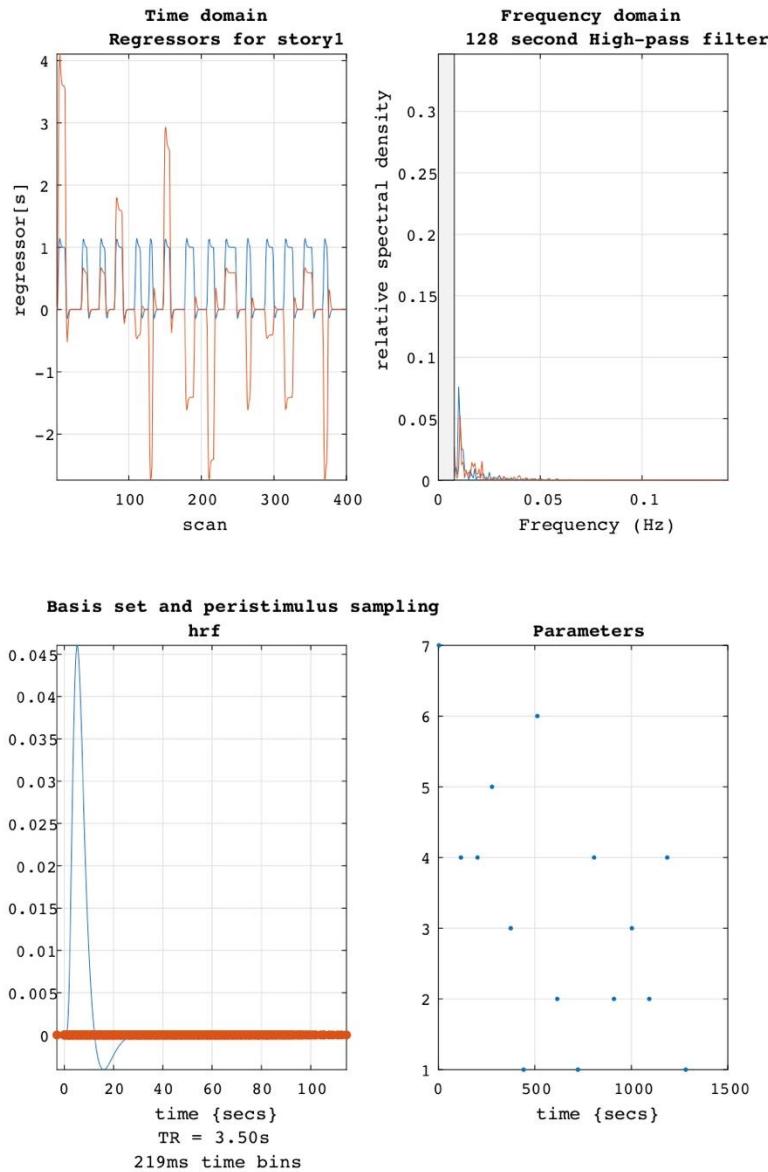
The model now has 12 columns with column 6-11 modeling the motion.



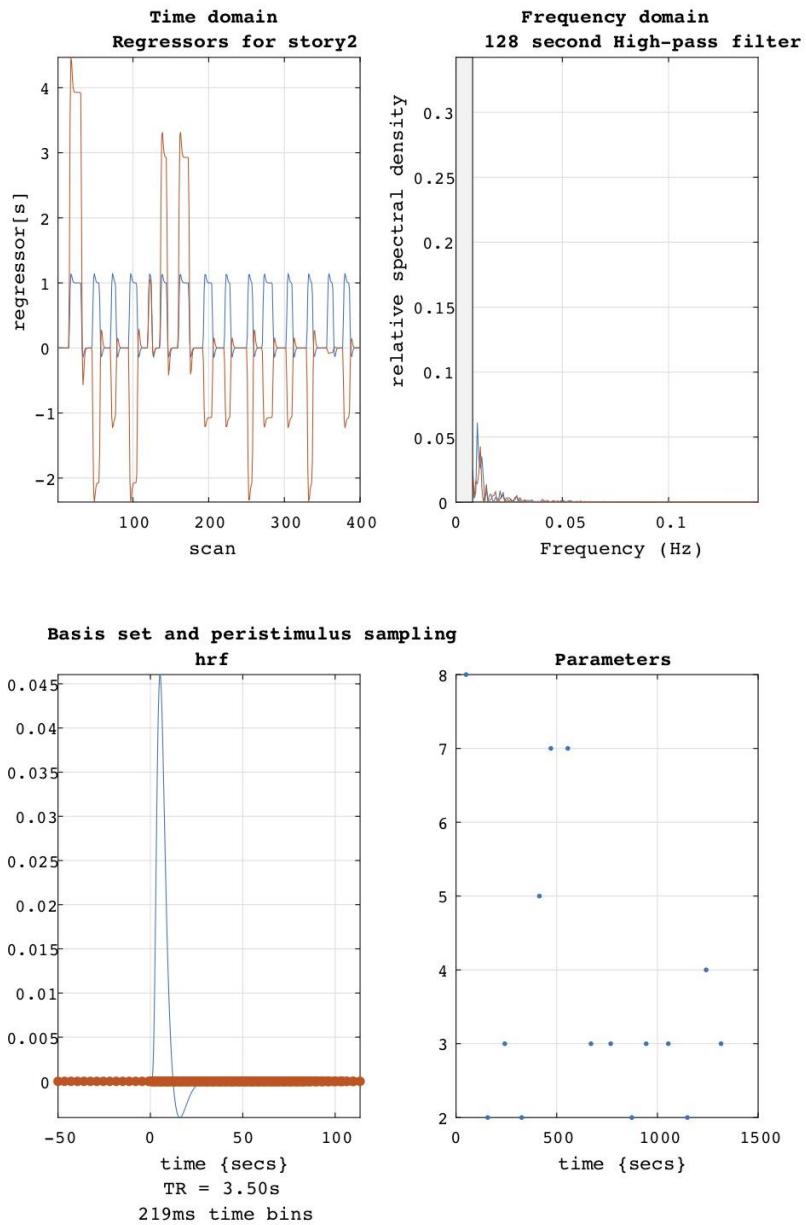
5. Checking the new model

5.a. Frequency domains for the three conditions.

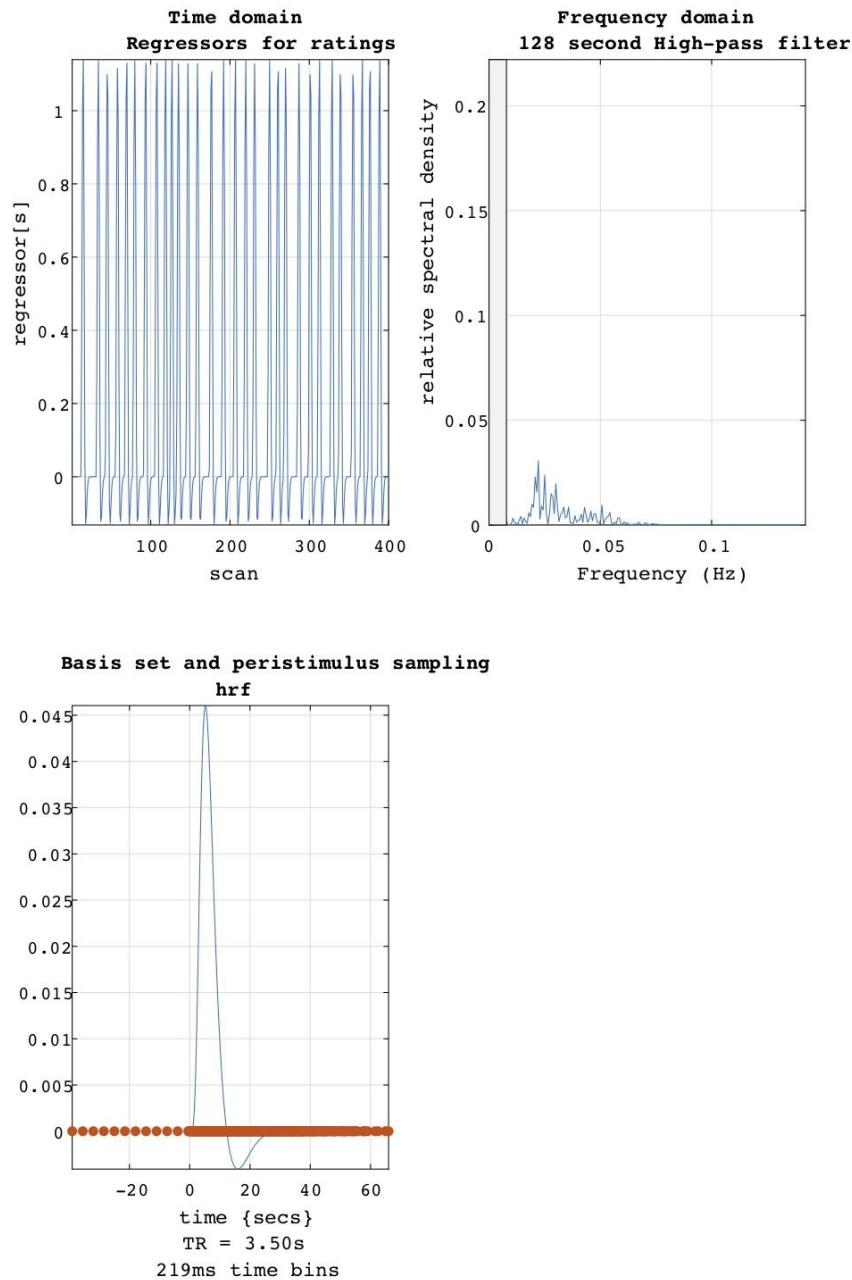
Story 1



Story 2



Ratings



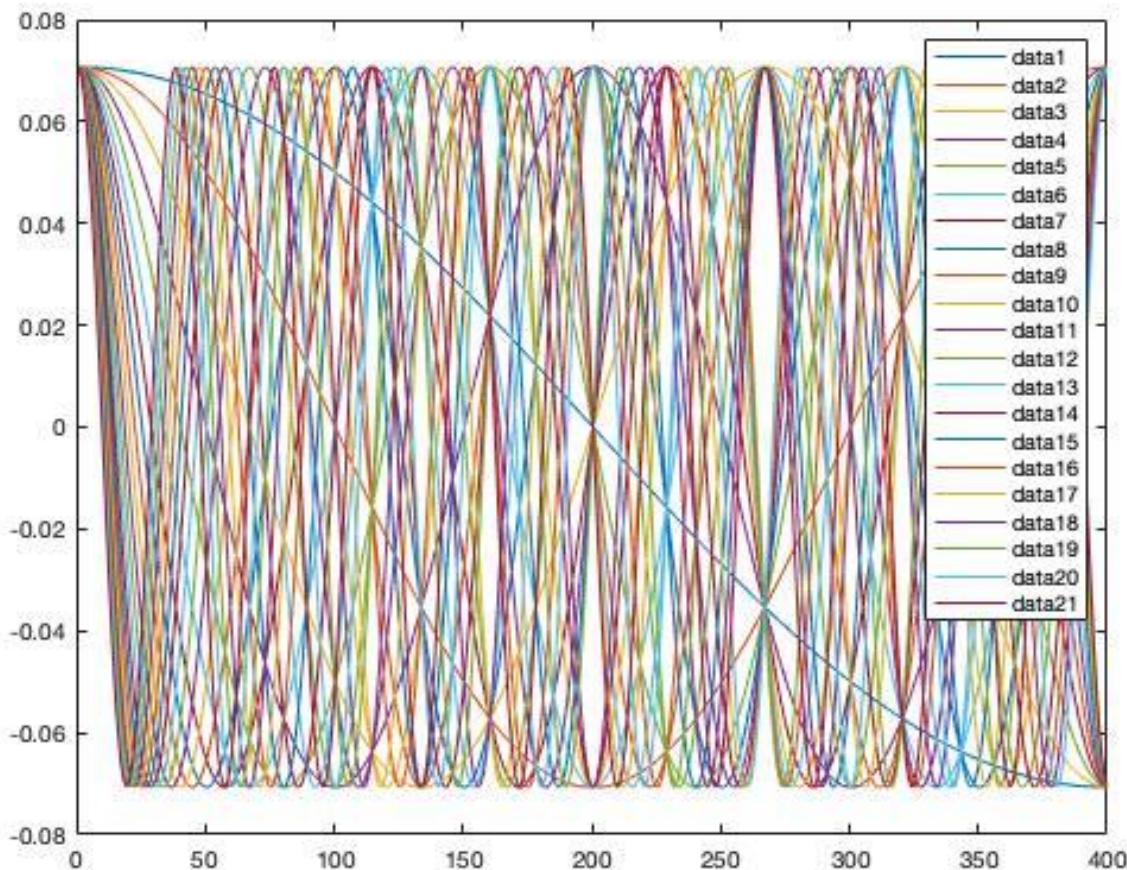
5.b. Eye-balling task: What are the most predominant frequencies for the covariates, as seen from these plots?

The most predominant frequency for story 1 and 2 is at 0.005 Hz, while the predominant frequency is 0.02 Hz for ratings.

5.c1. The lowest frequencies in the design are filtered out using a “high pass” filter in the analysis. This is the part of the spectrum marked by gray in the frequency/density plot. Does this filter seem to affect the covariates?

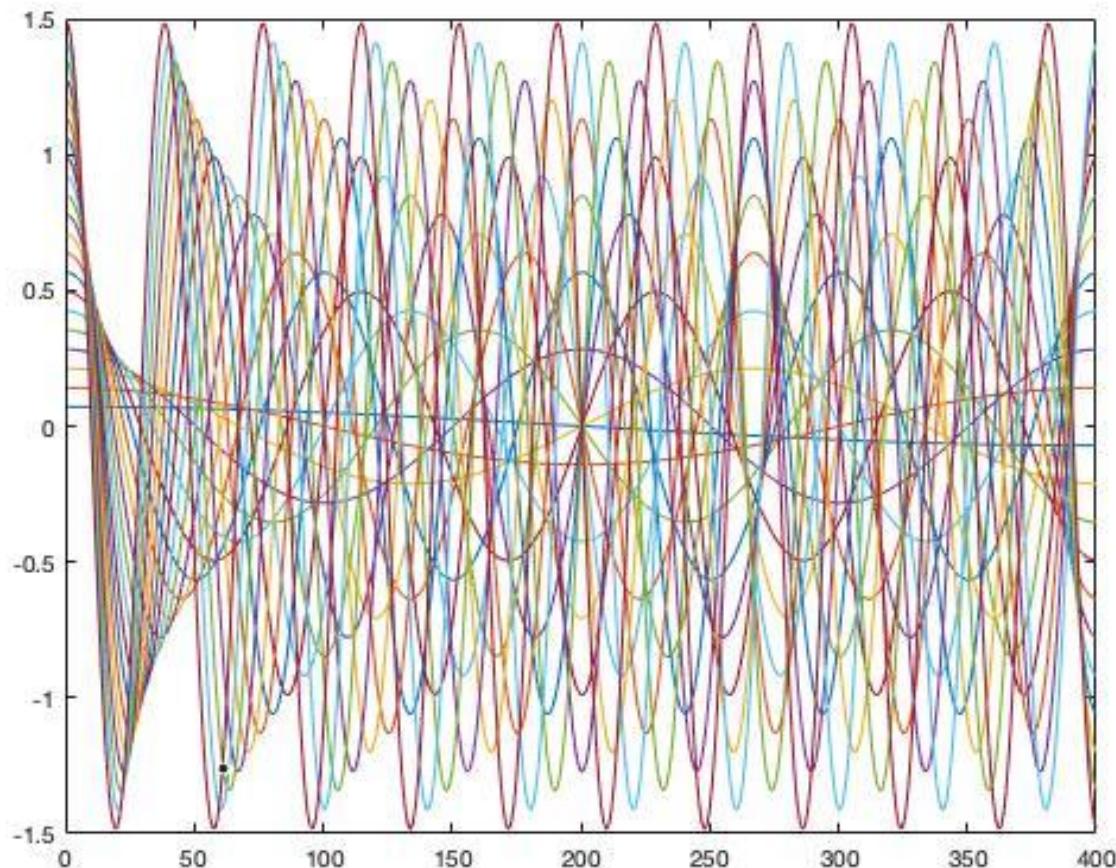
The filter affects the covariates, since it deletes signals of lower frequencies. This can be seen in SPM, where the gray box covers some of the data.

5.c2. The high-pass filter figure.



5.d. The high-pass figure has 21 cosine waves.

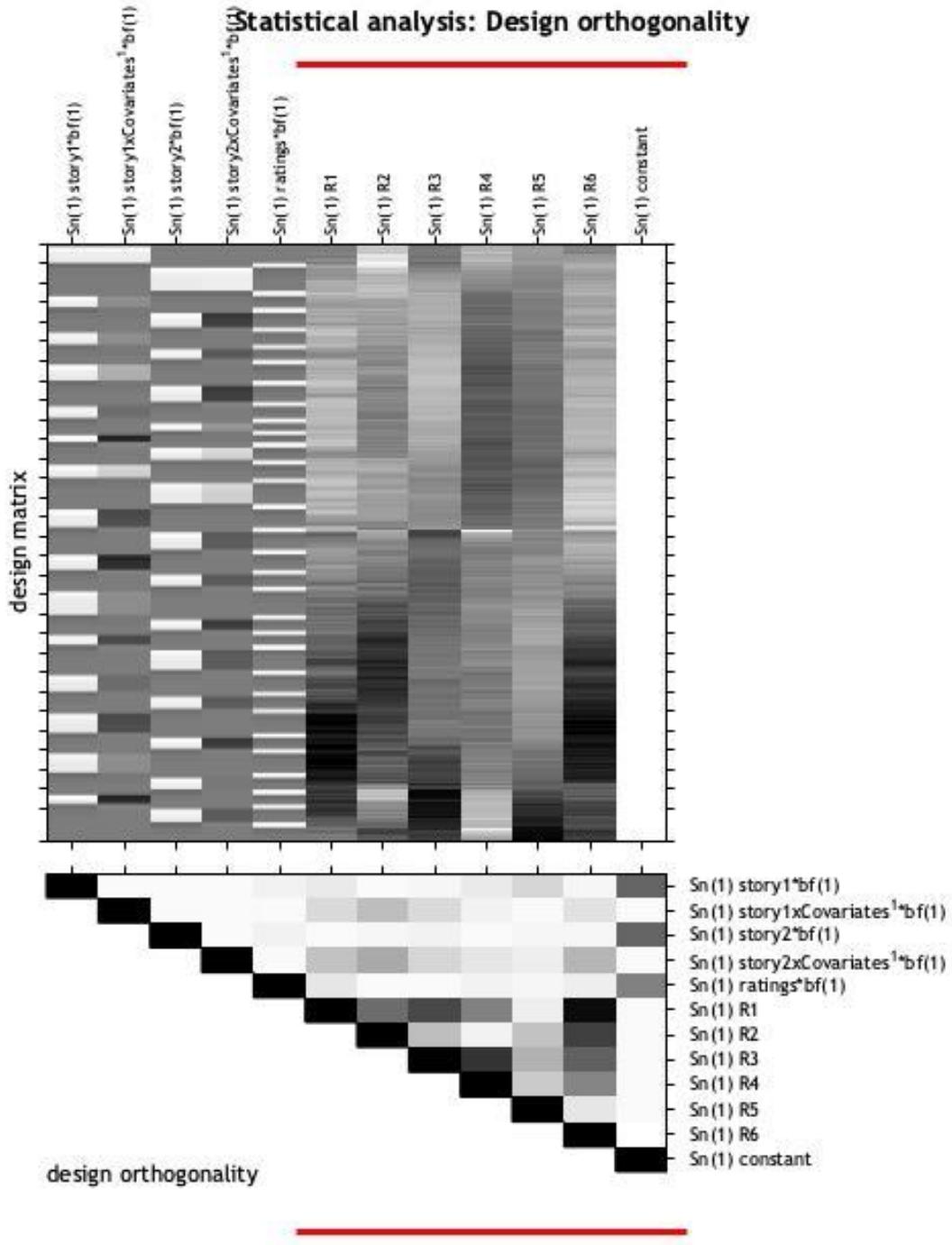
5.e. Hypothetical slow wave signal multiplied with the filter.



5.f. Exploring the design orthogonality and which covariates are the most correlated in the current design.

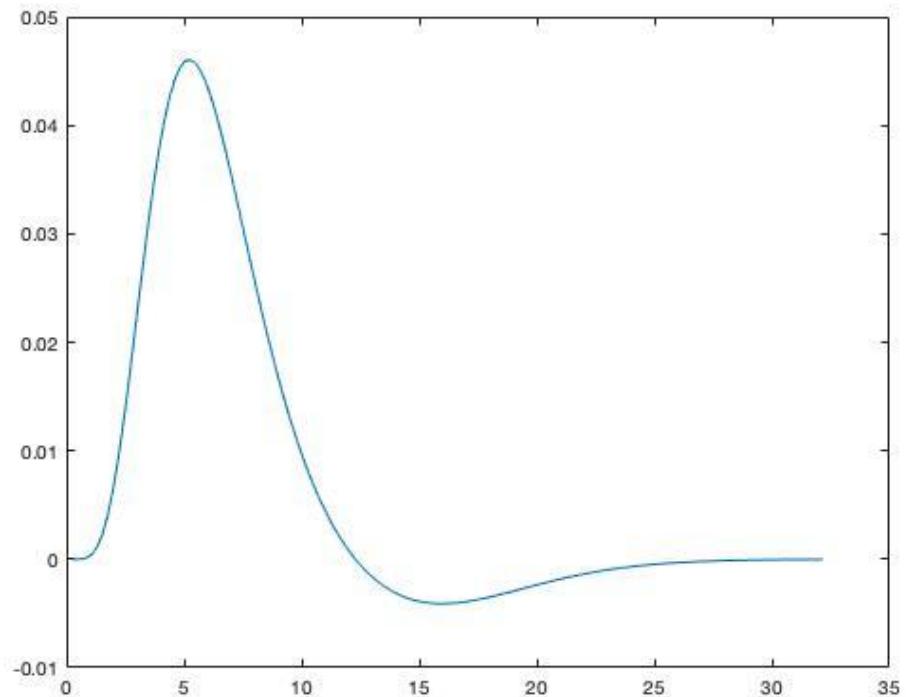
Covariates Sn(1)R6 and Sn(1)R1 are the most correlated as seen in the following heat map.

Statistical analysis: Design orthogonality



Measure : abs. value of cosine of angle between columns of design matrix
Scale :
black - colinear ($\cos=+1/-1$)
white - orthogonal ($\cos=0$)
gray - not orthogonal or colinear

5.g. Plot of the hemodynamic response function (HRF)



Assignment 6

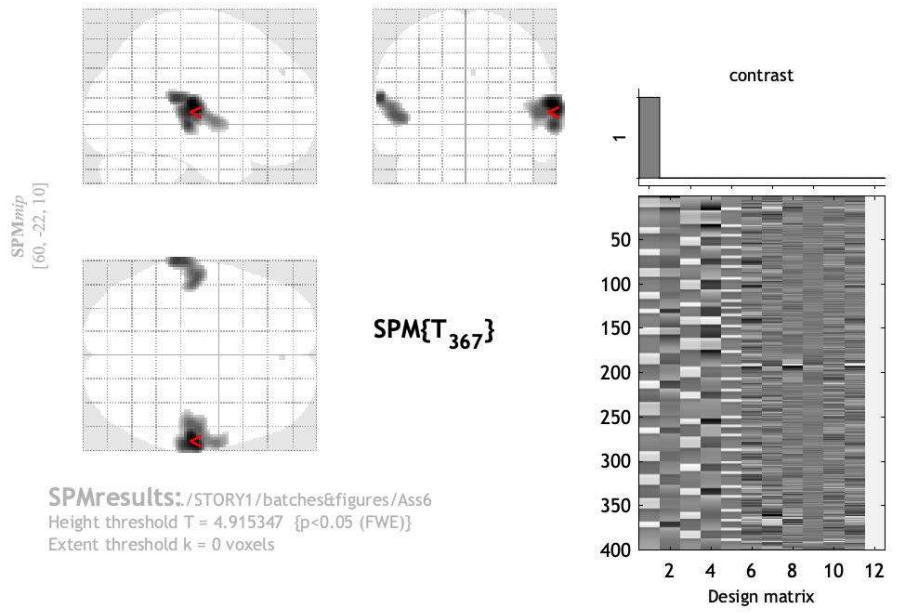
Group 9 - Bella Terragni, Daryna Eismont, Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad

17 mar 2019

T-contrasts

T-contrast for pos_story1 showed significant results with the family-wise error correction for multiple comparisons.

pos_story1



Statistics: p-values adjusted for search volume

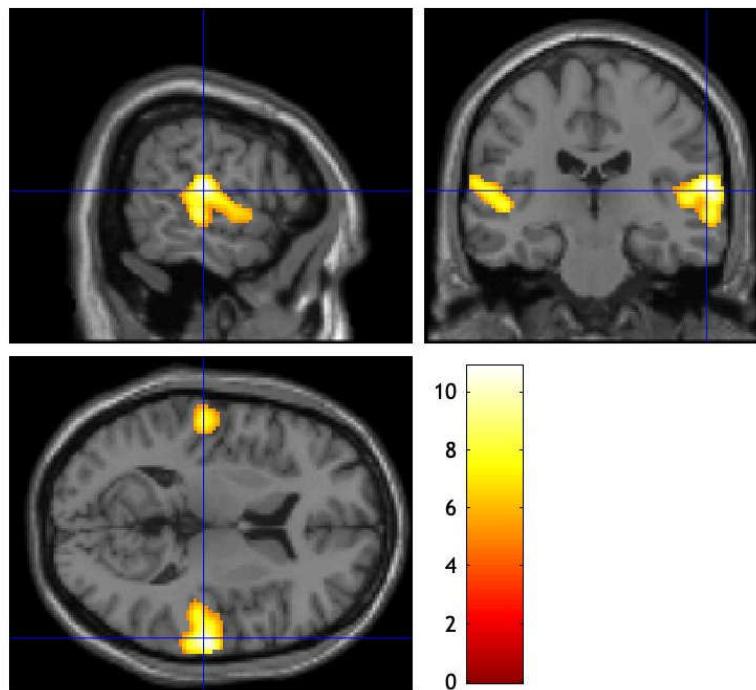
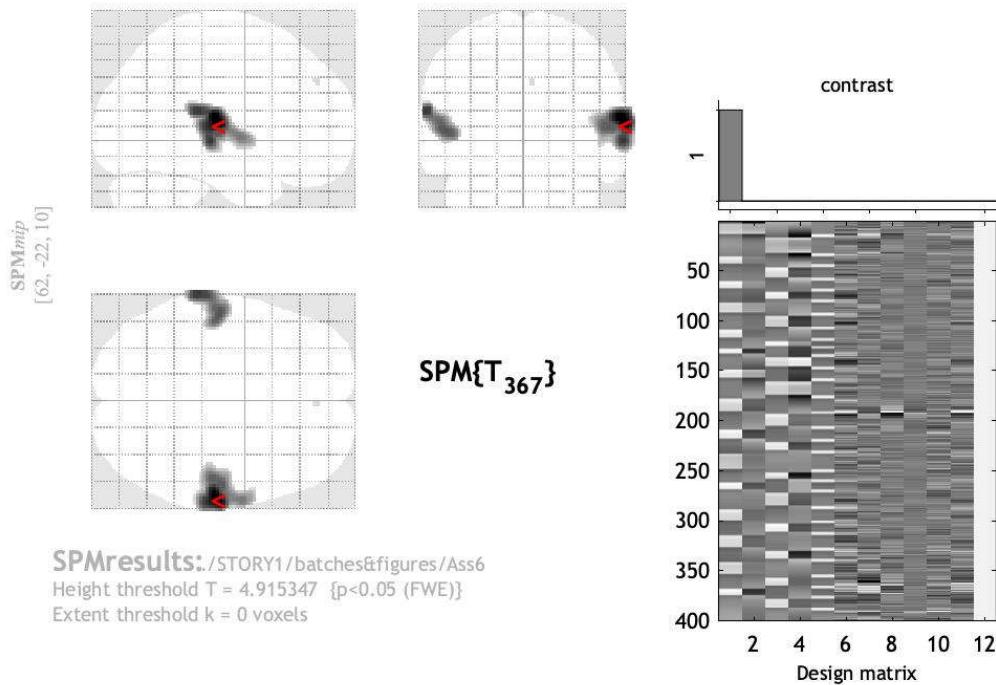
set-level		cluster-level				peak-level				mm mm mm			
p	c	P _{FWE-corr}	q _{FDR-corr}	k _E	P _{uncorr}	P _{FWE-corr}	q _{FDR-corr}	T	(Z _≡)	P _{uncorr}	mm	mm	mm
0.000	3	0.000	0.000	1194	0.000	0.000	0.000	10.84	Inf	0.000	62	-22	12
						0.000	0.000	9.24	Inf	0.000	64	-24	-2
						0.000	0.000	8.33	Inf	0.000	54	-18	6
		0.000	0.000	491	0.000	0.000	0.000	9.10	Inf	0.000	-66	-32	16
		0.016	0.309	3	0.309	0.031	0.620	5.03	4.94	0.000	-52	-18	4

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)
Extent threshold: k = 0 voxels
Expected voxels per cluster, <k> = 3.128
Expected number of clusters, <c> = 0.05
FWEp: 4.915, FDRp: 7.590, FWEc: 3, FDRC: 491

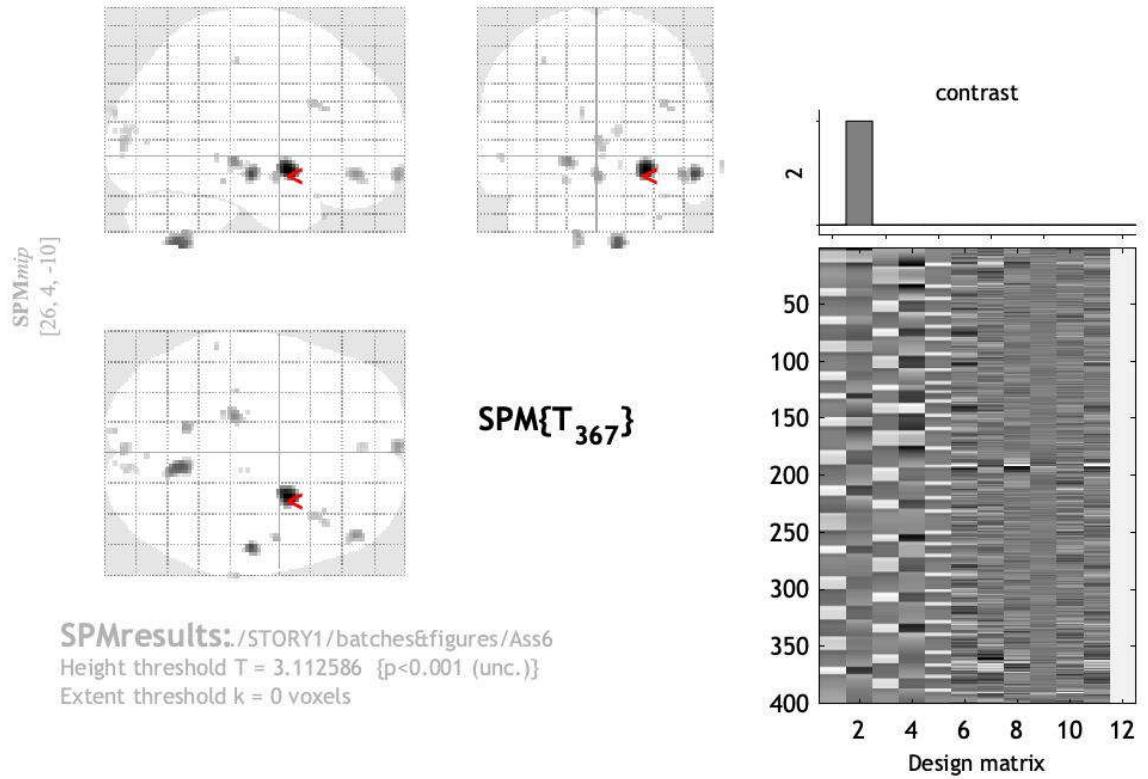
Degrees of freedom = [1.0, 367.0]
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
Volume: 1846984 = 230873 voxels = 1982.9 resels
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

pos_story1



T-contrast for pos_story1raring didn't show significant results with the FWE, but did show some significance with the uncorrected threshold ($p<0.001$)

pos_story1rating



Statistics: p-values adjusted for search volume

set-level	c	cluster-level				peak-level				mm mm mm			
		p	FWE-corr	q _{FDR-corr}	k _E	p _{uncorr}	p _{FWE-corr}	q _{FDR-corr}	T	(Z _≡)	p _{uncorr}	mm	mm
0.137	26	0.144	0.197	102	0.008	0.117	0.175	4.70	4.63	0.000	26	4	-10
		0.820	0.579	37	0.083	0.637	0.700	4.14	4.09	0.000	54	-18	-12
		0.368	0.291	70	0.022	0.773	0.700	4.03	3.98	0.000	10	-60	-50
		0.926	0.579	28	0.127	0.990	0.973	3.67	3.63	0.000	-18	-28	-4
		0.926	0.579	28	0.127	0.995	0.973	3.62	3.58	0.000	48	42	-12
		0.971	0.642	22	0.173	0.998	0.973	3.57	3.54	0.000	-2	66	-14
		0.936	0.579	27	0.134	0.998	0.973	3.56	3.52	0.000	-12	-56	-46
		1.000	0.800	9	0.380	1.000	0.973	3.43	3.40	0.000	50	-4	-14
		0.985	0.661	19	0.203	1.000	0.973	3.34	3.32	0.000	0	-92	6
		0.993	0.698	16	0.242	1.000	0.973	3.33	3.30	0.000	40	24	24
		1.000	0.800	3	0.628	1.000	0.973	3.30	3.27	0.001	-32	-26	50
		1.000	0.800	2	0.701	1.000	0.973	3.28	3.26	0.001	-20	-34	-16
		1.000	0.800	6	0.478	1.000	0.973	3.28	3.26	0.001	-10	-96	22
		1.000	0.800	1	0.800	1.000	0.973	3.27	3.24	0.001	34	24	-36
		1.000	0.800	4	0.569	1.000	0.973	3.26	3.24	0.001	70	-26	-6
		1.000	0.800	3	0.628	1.000	0.973	3.25	3.22	0.001	-56	-38	-2
		1.000	0.800	7	0.441	1.000	0.973	3.24	3.21	0.001	-2	46	-16
		1.000	0.800	8	0.409	1.000	0.973	3.23	3.21	0.001	12	-90	12
		1.000	0.800	1	0.800	1.000	0.973	3.19	3.16	0.001	4	-78	6
		1.000	0.800	1	0.800	1.000	0.973	3.16	3.14	0.001	-20	-20	-14
		1.000	0.800	1	0.800	1.000	0.973	3.13	3.11	0.001	12	-24	-8
		1.000	0.800	1	0.800	1.000	0.973	3.13	3.11	0.001	12	-84	12
		1.000	0.800	2	0.701	1.000	0.973	3.13	3.11	0.001	36	20	-24
		1.000	0.800	1	0.800	1.000	0.973	3.13	3.10	0.001	2	-92	-2
		1.000	0.800	1	0.800	1.000	0.973	3.13	3.10	0.001	8	-96	12
		1.000	0.800	1	0.800	1.000	0.973	3.12	3.10	0.001	12	-20	-6

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.11, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 12.568$

Expected number of clusters, $\langle c \rangle = 20.52$

FWEp: 4.915, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 367.0]

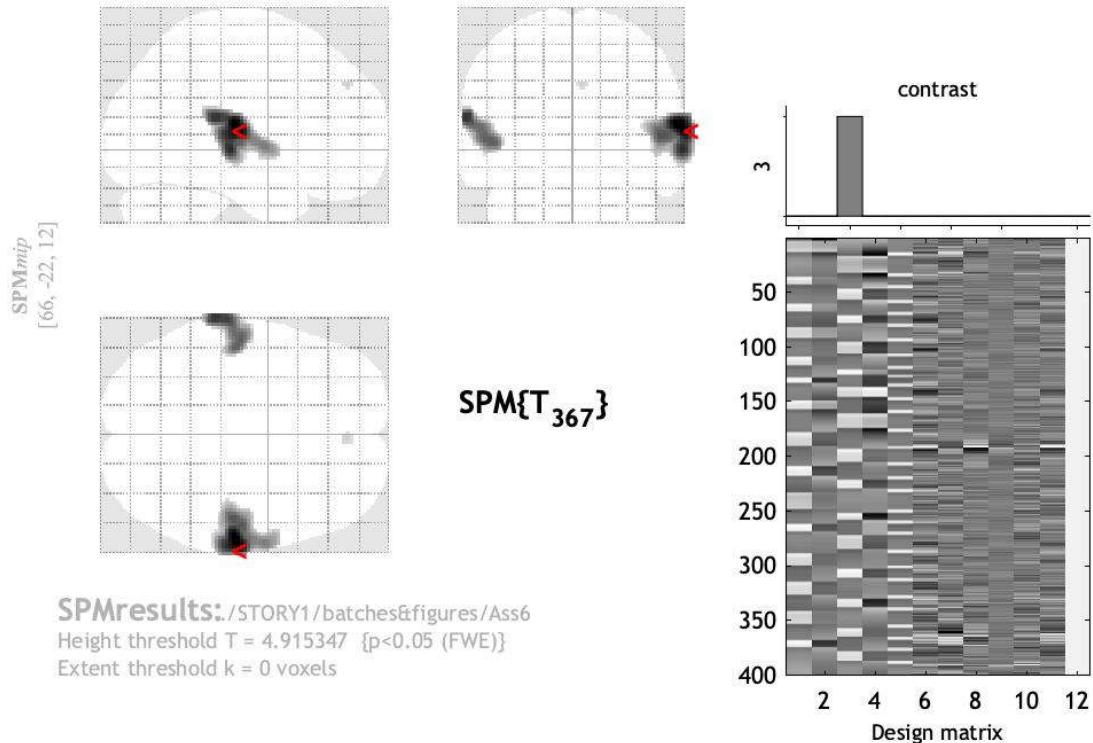
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

T-contrast for pos_story2 showed significant results with the FEW.

pos_story2



Statistics: p-values adjusted for search volume

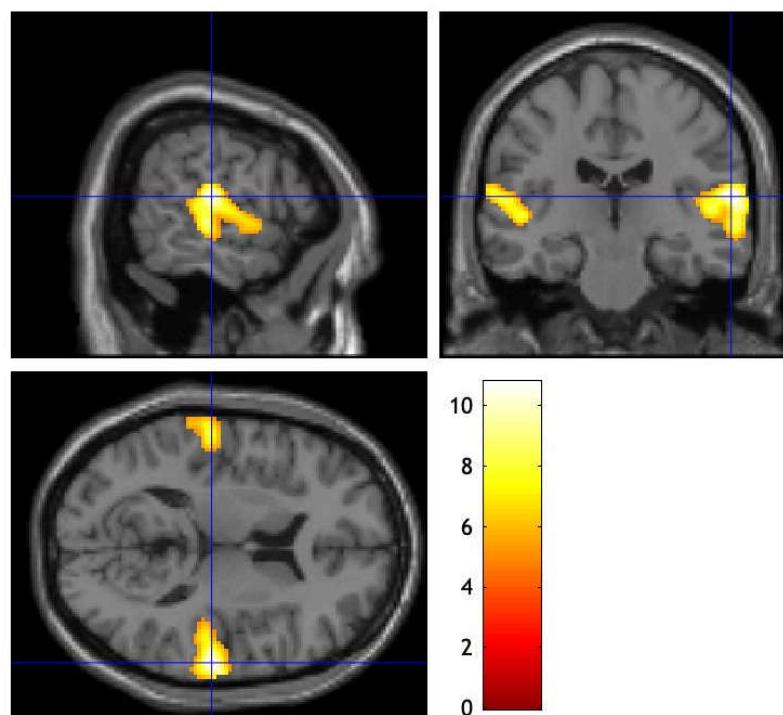
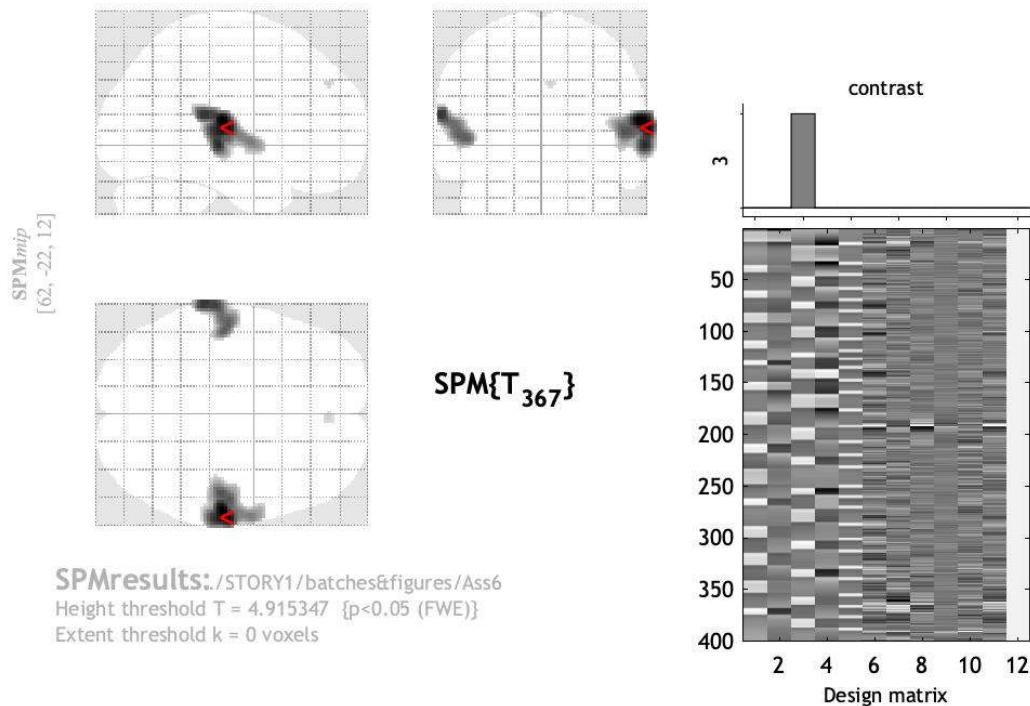
set-level		cluster-level				peak-level					mm mm mm		
p	c	P _{FWE-corr}	q _{FDR-corr}	k _E	p _{uncorr}	P _{FWE-corr}	q _{FDR-corr}	T	(Z ₊)	p _{uncorr}			
0.000	3	0.000	0.000	1239	0.000	0.000	0.000	10.75	Inf	0.000	62	-22	12
						0.000	0.000	9.50	Inf	0.000	64	-24	-2
						0.000	0.000	8.47	Inf	0.000	50	-22	6
0.005	479	0.000	0.000	479	0.000	0.000	0.000	9.21	Inf	0.000	-66	-32	16
						0.000	0.000	8.15	7.80	0.000	-50	-20	2
						0.000	0.000	7.87	7.56	0.000	-56	-22	10
		0.005	0.104	8	0.104	0.013	0.252	5.23	5.14	0.000	4	46	36

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)
 Extent threshold: k = 0 voxels
 Expected voxels per cluster, $\langle k \rangle = 3.128$
 Expected number of clusters, $\langle c \rangle = 0.05$
 FWEp: 4.915, FDRp: 7.377, FWEc: 8, FDRc: 479

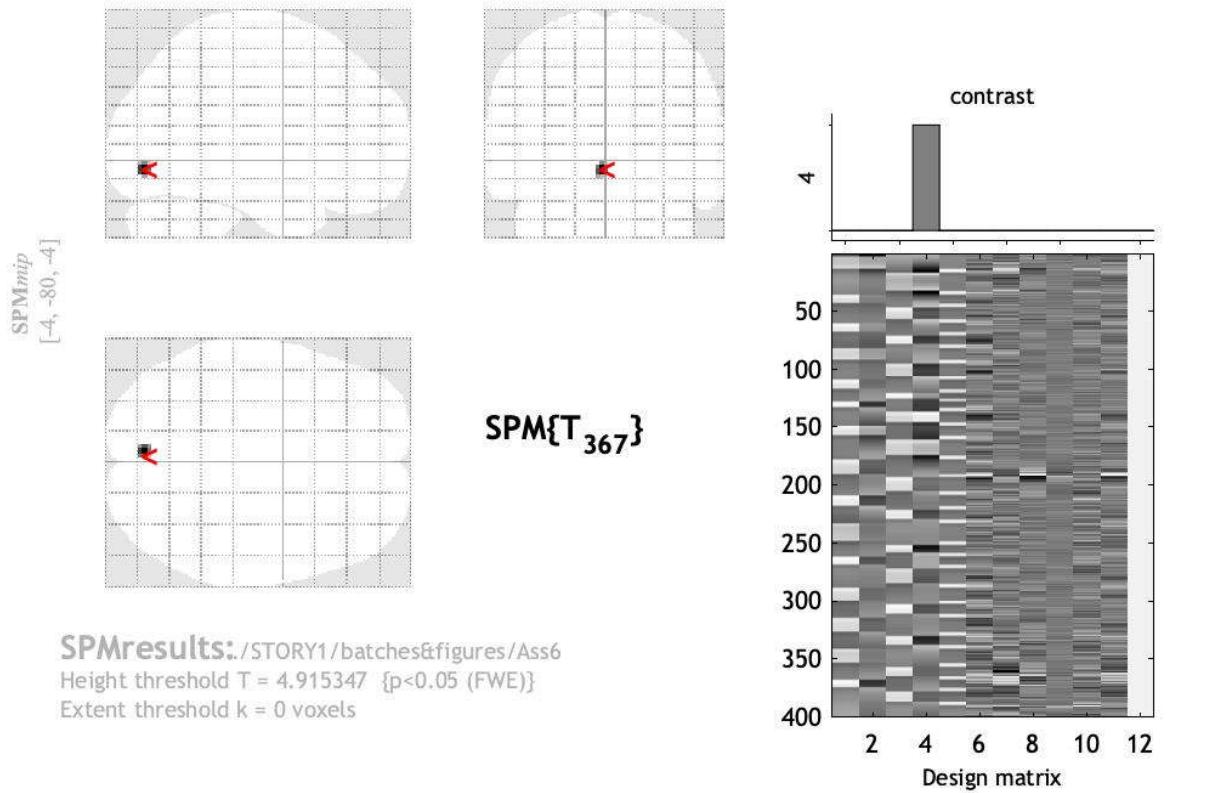
Degrees of freedom = [1.0, 367.0]
 FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
 Volume: 1846984 = 230873 voxels = 1982.9 resels
 Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

pos_story2



T-contrast for pos_story2rating showed significant results with the FEW.

pos_story2rating



Statistics: p-values adjusted for search volume

cluster-level				peak-level					mm mm mm		
$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	k_E	P_{uncorr}	$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	(Z_{all})	P_{uncorr}	mm	mm	mm
0.001	0.010	23	0.010	0.002	0.030	5.69	5.57	0.000	-4	-80	-6

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 3.128$

Expected number of clusters, $\langle c \rangle = 0.05$

FWEp: 4.915, FDRp: 5.691, FWEc: 23, FDRc: 23

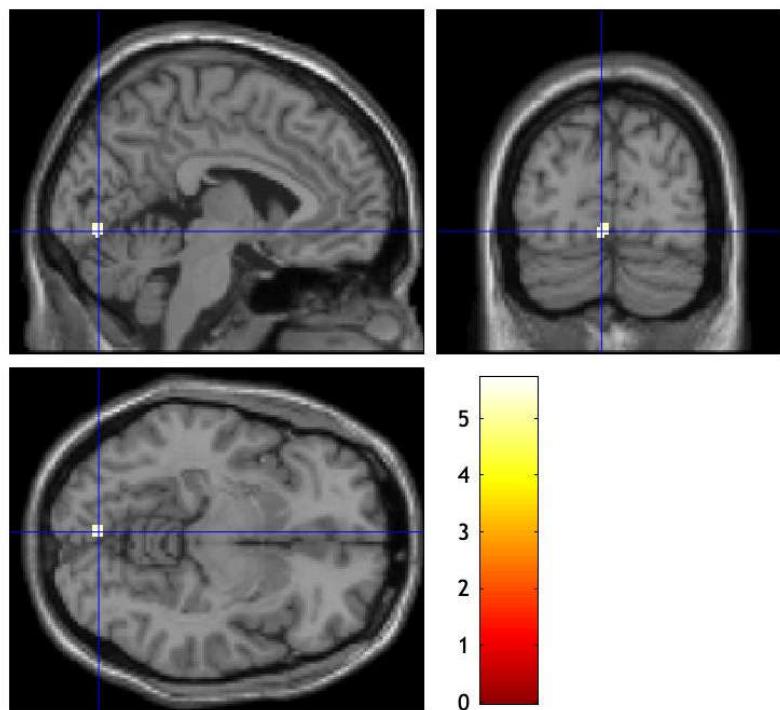
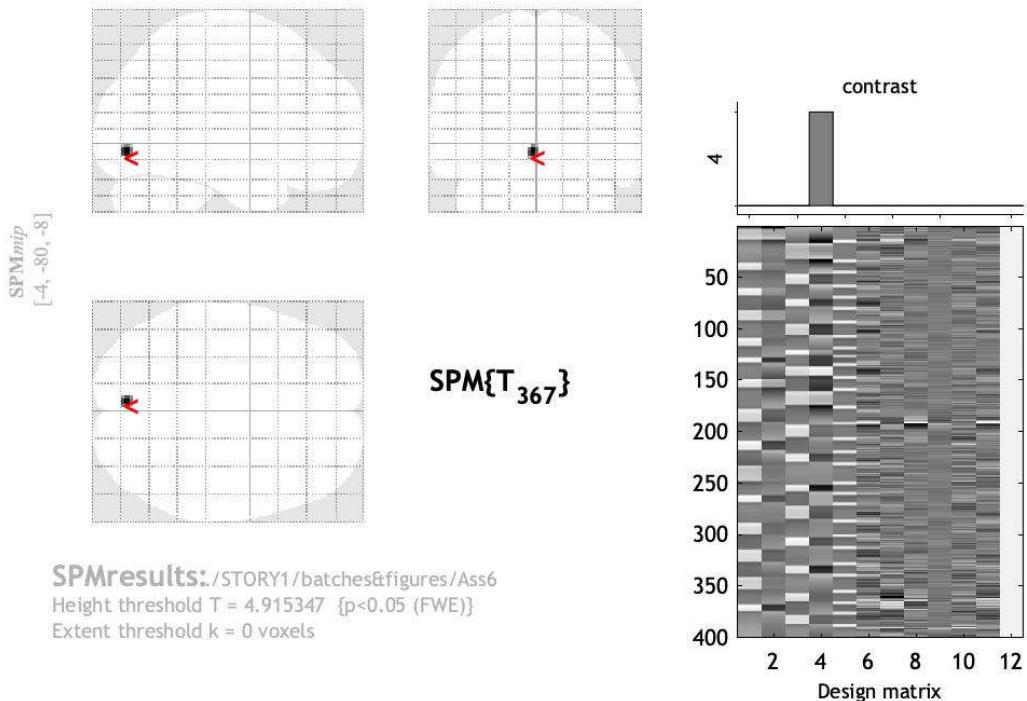
Degrees of freedom = [1.0, 367.0]

FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

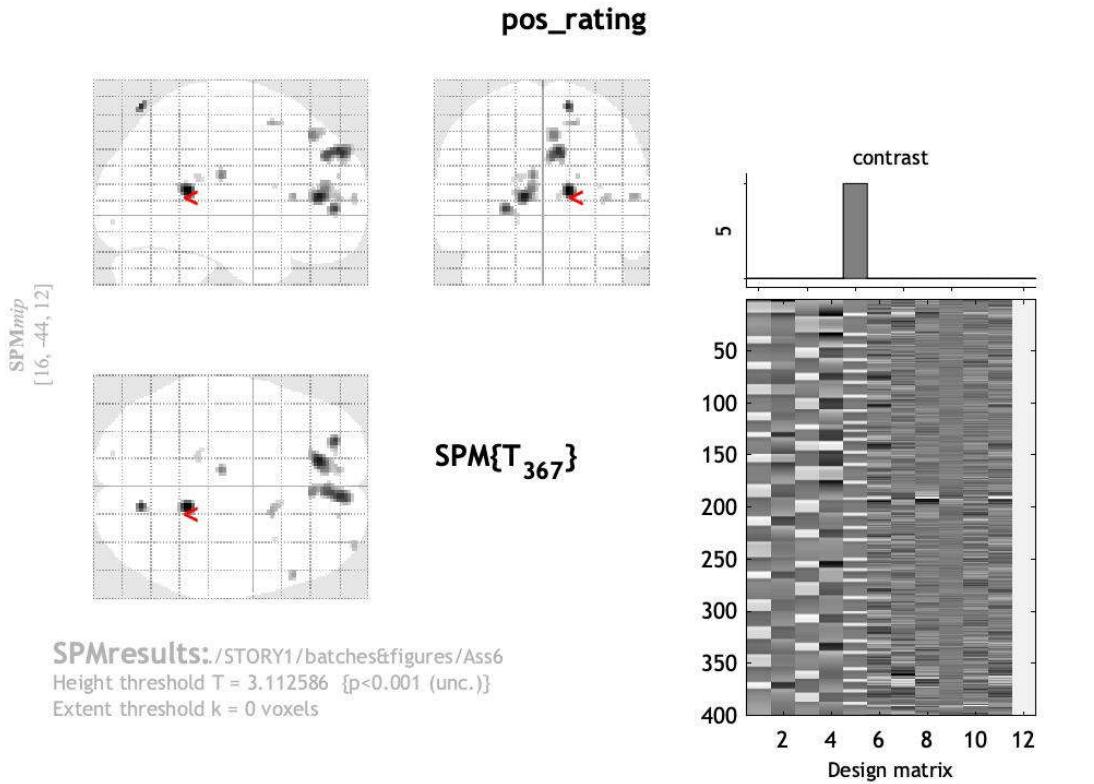
Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

pos_story2rating



T-contrast for pos_rating didn't show significant results with the FWE, but did show some significance with the uncorrected threshold (p<0.001)



Statistics: p-values adjusted for search volume

		set-level				cluster-level				peak-level						
		<i>p</i>	<i>c</i>	<i>p</i> _{FWE-corr}	<i>q</i> _{FDR-corr}	<i>k_E</i>	<i>p</i> _{uncorr}	<i>p</i> _{FWE-corr}	<i>q</i> _{FDR-corr}	<i>T</i>	(<i>Z_≥</i>)	<i>p</i> _{uncorr}	mm	mm	mm	
0.868	16	0.540	0.202	56	0.038	0.159	0.152	4.62	4.55	0.000	14	-44	14			
		0.066	0.027	129	0.003	0.347	0.160	4.38	4.32	0.000	-14	40	10			
		0.999	0.588	11	0.331	0.479	0.160	4.26	4.21	0.000	14	-72	64			
		0.059	0.027	133	0.003	0.517	0.160	4.23	4.18	0.000	8	54	36			
		0.847	0.292	35	0.091	0.661	0.178	4.12	4.07	0.000	-26	50	2			
		0.762	0.280	41	0.070	0.980	0.458	3.72	3.68	0.000	6	34	46			
		0.995	0.588	15	0.257	0.985	0.458	3.70	3.67	0.000	-8	-22	22			
		0.998	0.588	12	0.310	0.999	0.667	3.51	3.48	0.000	56	24	10			
		0.999	0.588	11	0.331	1.000	0.667	3.50	3.46	0.000	18	10	54			
		1.000	0.654	8	0.409	1.000	0.765	3.38	3.35	0.000	38	62	8			
		1.000	0.800	1	0.800	1.000	0.960	3.25	3.22	0.001	20	52	22			
		1.000	0.800	3	0.628	1.000	0.960	3.21	3.18	0.001	8	52	10			
		1.000	0.756	5	0.520	1.000	0.960	3.20	3.18	0.001	-18	-36	22			
		1.000	0.800	1	0.800	1.000	0.984	3.13	3.11	0.001	34	12	16			
		1.000	0.800	1	0.800	1.000	0.984	3.12	3.10	0.001	-18	-90	-6			
		1.000	0.800	1	0.800	1.000	0.984	3.12	3.10	0.001	-20	-34	20			

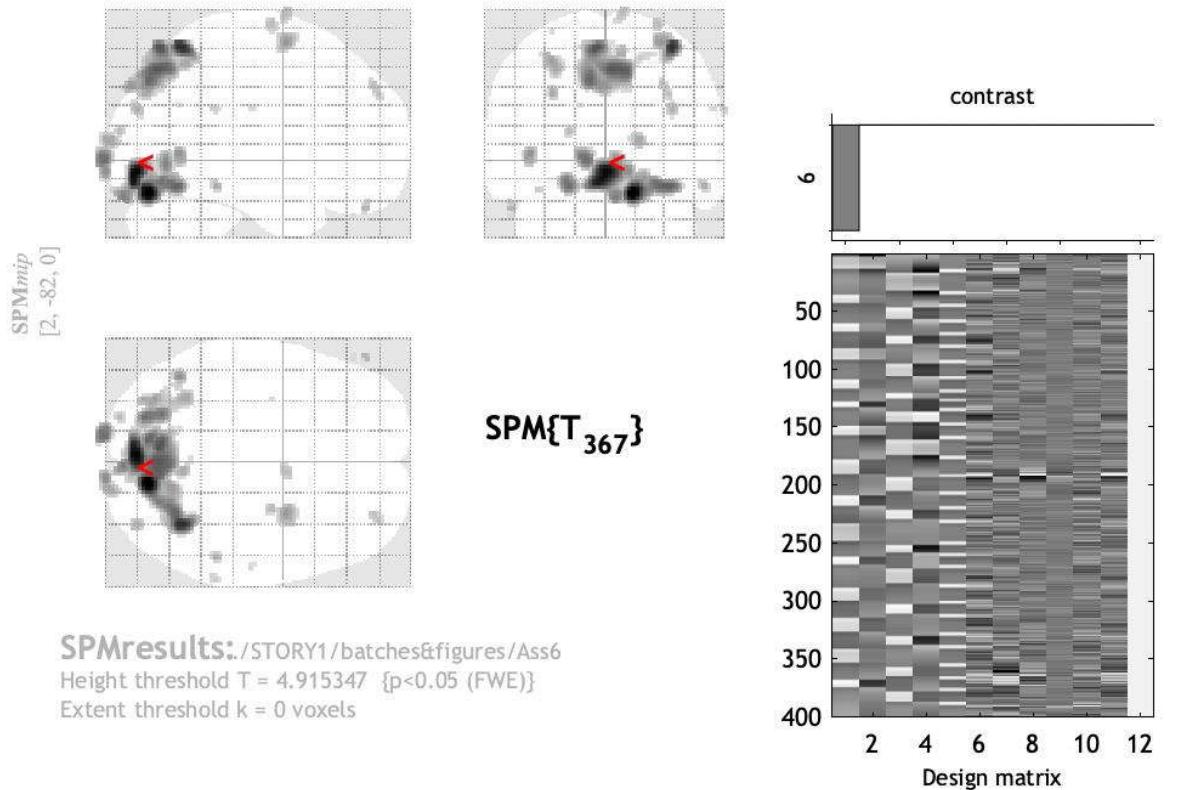
table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.11, p = 0.001 (1.000)
Extent threshold: k = 0 voxels
Expected voxels per cluster, $\langle k \rangle = 12.568$
Expected number of clusters, $\langle c \rangle = 20.52$
FWEp: 4.915, FDRp: Inf, FWEc: Inf, FDRc: 129

Degrees of freedom = [1.0, 367.0]
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
Volume: 1846984 = 230873 voxels = 1982.9 resels
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

T-contrast for neg_story1 showed significant results with the

neg_story1



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level				mm mm mm			
p	c	P _{FWE-corr}	q _{FDR-corr}	k _E	P _{uncorr}	P _{FWE-corr}	q _{FDR-corr}	T	(Z ₊)	P _{uncorr}	mm	mm	mm
0.000	25	0.000	0.000	1260	0.000	0.000	0.000	8.49	Inf	0.000	14	-78	-20
						0.000	0.000	8.28	Inf	0.000	-2	-84	-8
						0.000	0.000	6.82	6.61	0.000	32	-62	-16
		0.000	0.000	159	0.000	0.000	0.000	7.66	7.38	0.000	36	-60	60
		0.000	0.000	1232	0.000	0.000	0.000	7.09	6.86	0.000	-6	-72	60
						0.000	0.000	6.98	6.76	0.000	2	-72	48
						0.000	0.000	6.84	6.64	0.000	10	-74	46
		0.000	0.000	94	0.000	0.000	0.001	6.61	6.42	0.000	-12	-102	-2
		0.000	0.003	43	0.001	0.000	0.008	6.16	6.00	0.000	10	-102	-8
		0.000	0.002	45	0.001	0.000	0.011	6.08	5.93	0.000	-44	-60	2
		0.000	0.000	94	0.000	0.000	0.014	6.01	5.87	0.000	30	0	62
		0.000	0.000	161	0.000	0.001	0.022	5.90	5.76	0.000	-32	-74	52
						0.001	0.036	5.78	5.65	0.000	-32	-58	50
		0.001	0.026	21	0.014	0.001	0.042	5.74	5.62	0.000	-32	-80	34
		0.000	0.002	47	0.001	0.001	0.042	5.73	5.61	0.000	18	-98	10
		0.000	0.008	32	0.003	0.001	0.047	5.70	5.58	0.000	6	-58	56
		0.002	0.050	15	0.032	0.002	0.052	5.67	5.55	0.000	54	-40	-16
		0.000	0.021	24	0.009	0.002	0.056	5.65	5.53	0.000	36	-84	26
		0.006	0.176	7	0.126	0.003	0.074	5.58	5.47	0.000	-56	44	-28
		0.002	0.050	15	0.032	0.004	0.125	5.46	5.35	0.000	28	-8	72
		0.001	0.026	21	0.014	0.005	0.132	5.44	5.33	0.000	8	0	80
		0.002	0.055	14	0.038	0.005	0.132	5.44	5.33	0.000	-22	-94	14
		0.001	0.028	20	0.016	0.006	0.152	5.40	5.29	0.000	34	52	34
		0.008	0.204	6	0.155	0.015	0.355	5.19	5.10	0.000	66	-40	26
		0.010	0.239	5	0.192	0.016	0.360	5.18	5.09	0.000	-40	-60	-18
		0.016	0.367	3	0.309	0.024	0.511	5.09	5.00	0.000	60	-20	26
		0.029	0.568	1	0.568	0.034	0.722	5.01	4.92	0.000	-16	-60	28
		0.021	0.443	2	0.408	0.035	0.722	5.00	4.92	0.000	-54	38	-32

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 3.128$

Expected number of clusters, $\langle c \rangle = 0.05$

FWEp: 4.915, FDRp: 5.700, FWEc: 1, FDRC: 20

Degrees of freedom = [1.0, 367.0]

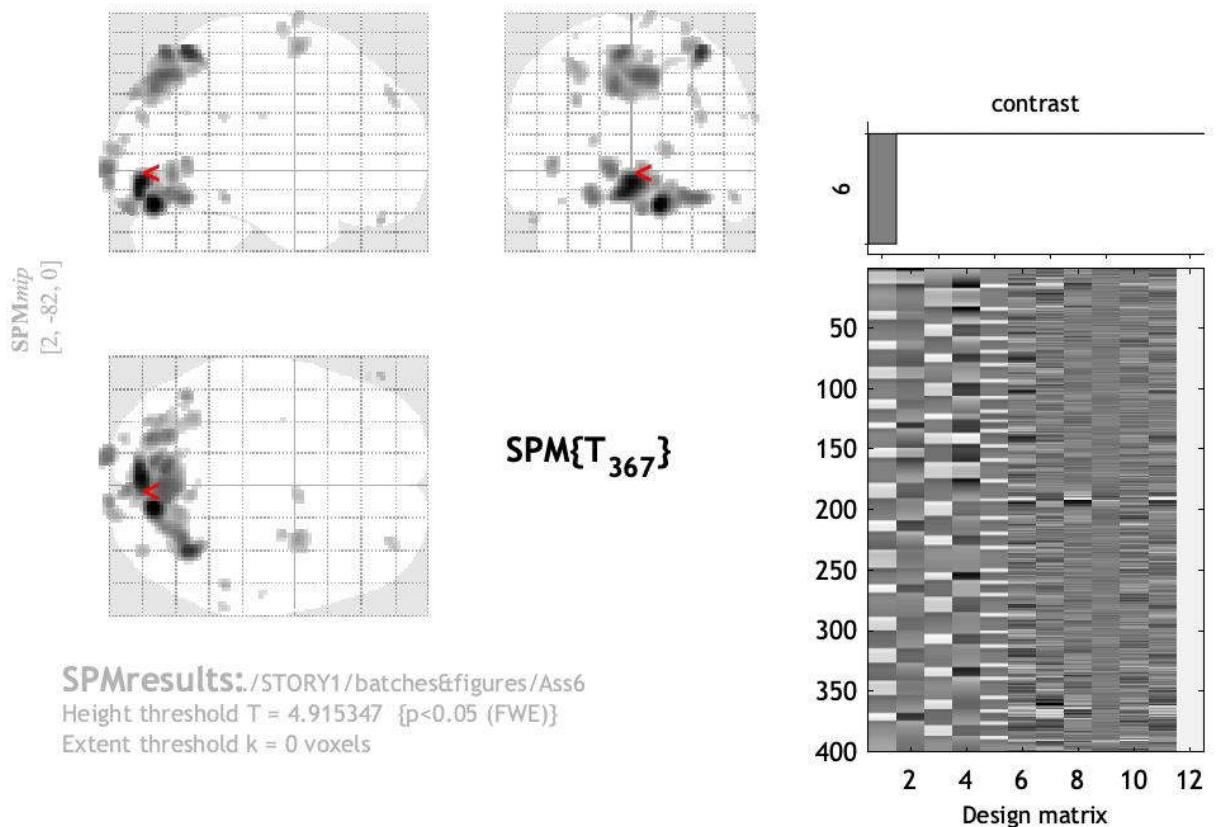
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

Page 1

neg_story1



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	k_E	p_{uncorr}	$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	(Z _≡)	p_{uncorr}	-32	-8	68
0.029		0.568	0.568	1	0.568	0.036	0.722	5.00	4.91	0.000			
0.021		0.443	0.443	2	0.408	0.045	0.890	4.94	4.86	0.000	2	-54	62

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle c \rangle = 3.128$

Expected number of clusters, $\langle c \rangle = 0.05$

FWEp: 4.915, FDRp: 5.700, FWEc: 1, FDRc: 20

Degrees of freedom = [1.0, 367.0]

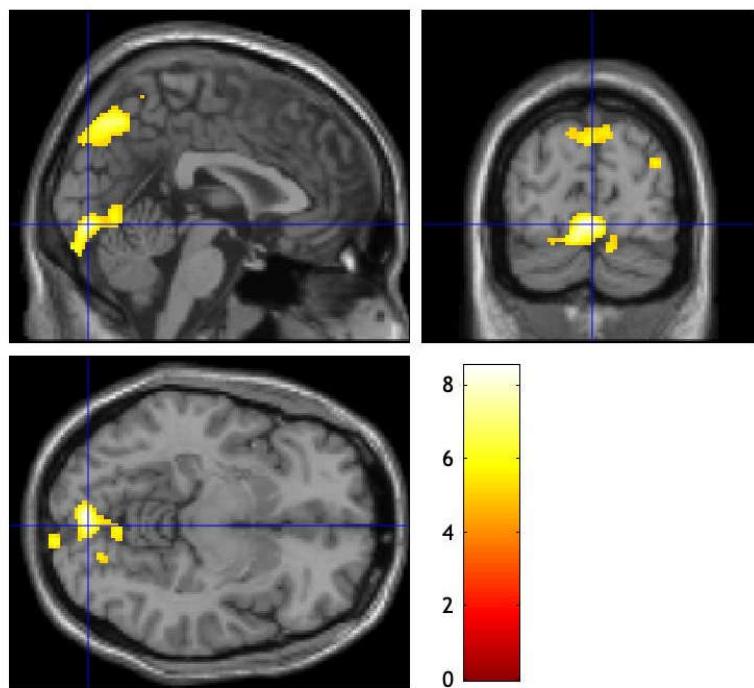
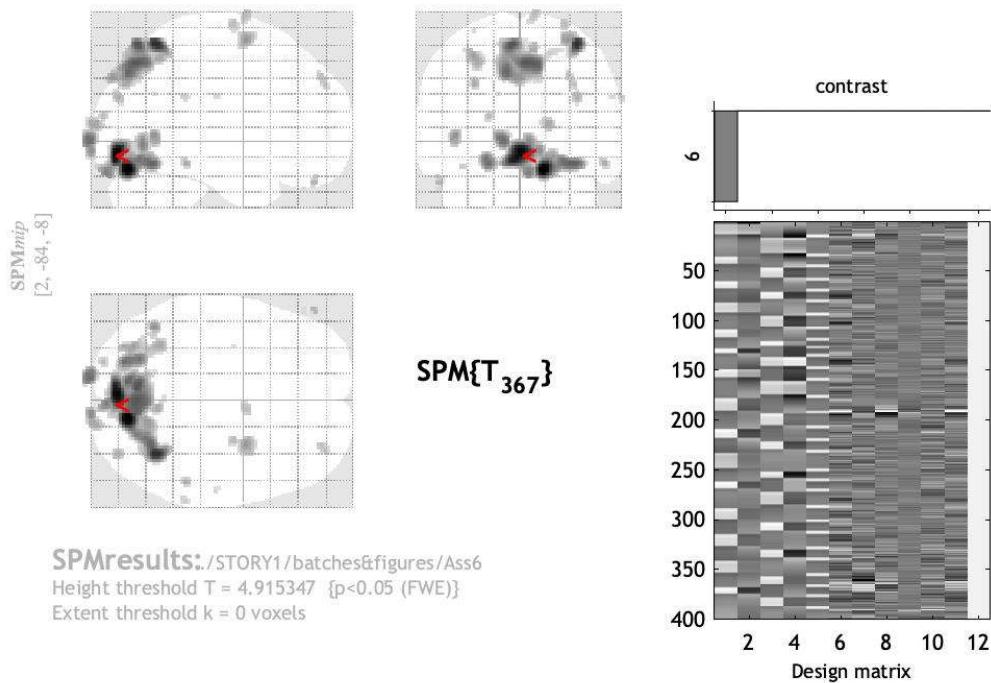
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

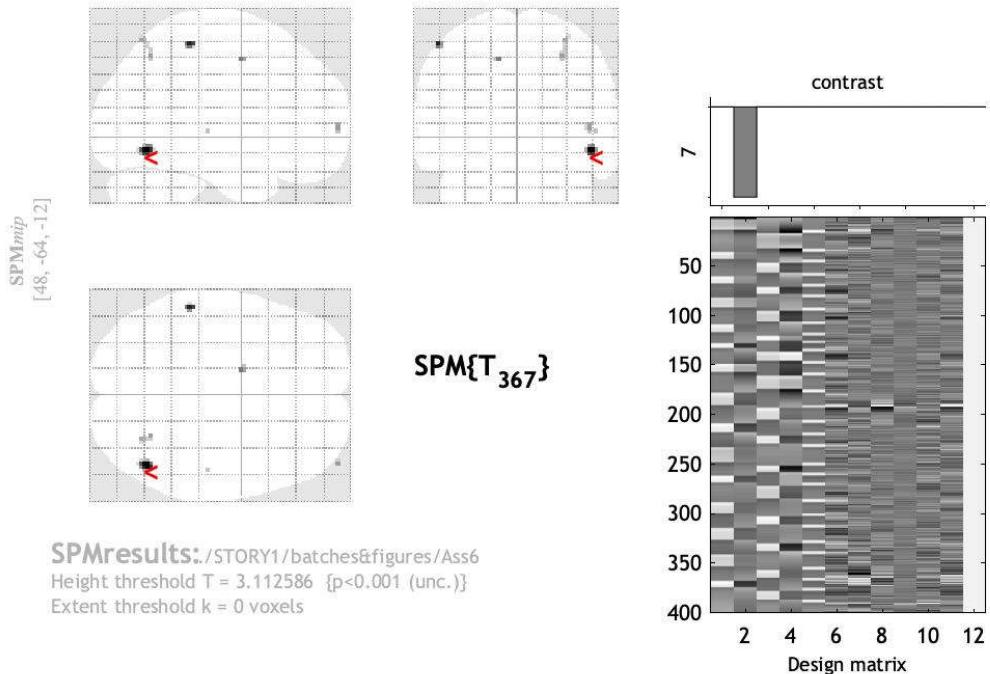
Page 2/2

neg_story1



T-contrast for neg_story1 rating didn't show significant results with the FWE, but did show some significance with the uncorrected threshold (p<0.001)

neg_story1rating



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level						mm	mm	mm
p	c	$P_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	P_{uncorr}	$P_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	P_{uncorr}				
1.000	6	0.884	0.630	32	0.105	0.968	0.855	3.77	3.73	0.000	46	-66	-10	
		1.000	0.754	9	0.380	0.993	0.855	3.64	3.61	0.000	-54	-36	58	
		1.000	0.754	3	0.628	1.000	0.855	3.40	3.37	0.000	-14	-2	48	
		0.999	0.754	11	0.331	1.000	0.855	3.31	3.28	0.001	28	-62	50	
						1.000	0.855	3.28	3.25	0.001	30	-68	60	
		1.000	0.754	5	0.520	1.000	0.855	3.30	3.27	0.001	46	62	4	
		1.000	0.800	1	0.800	1.000	0.855	3.20	3.17	0.001	50	-24	2	

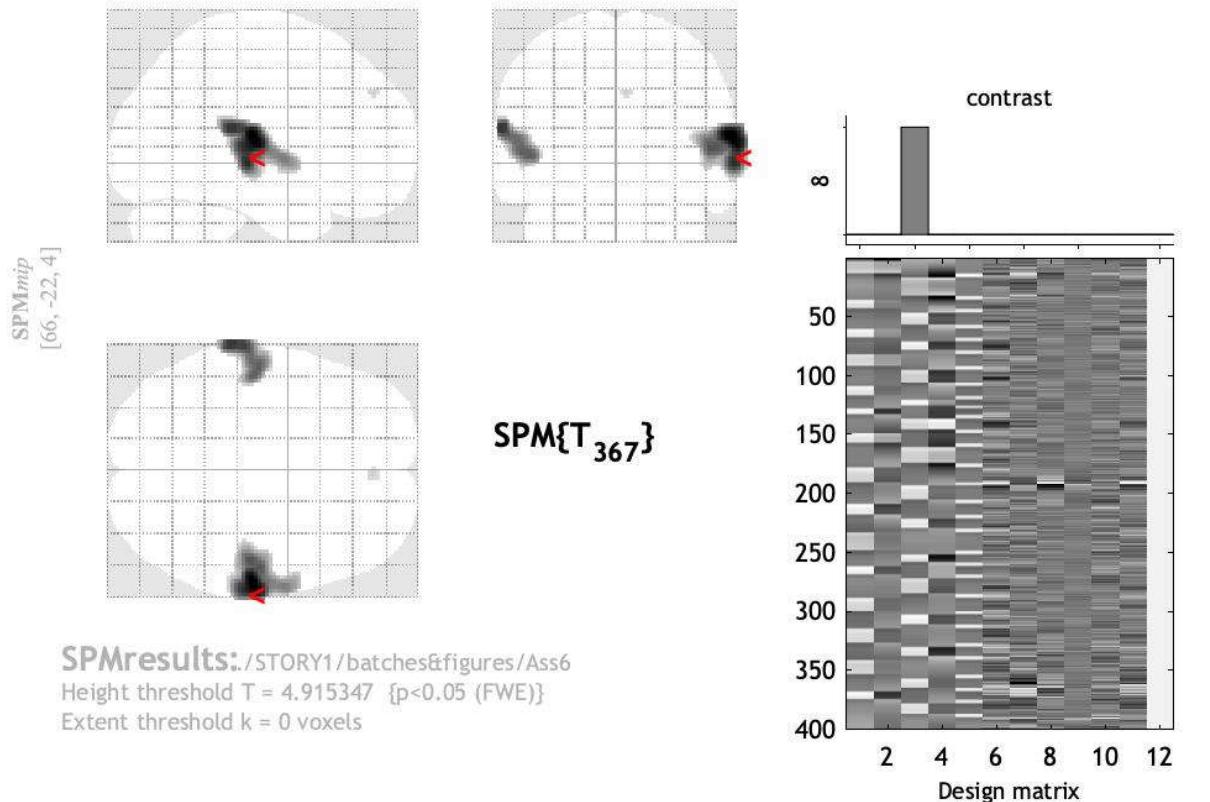
table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.11, p = 0.001 (1.000)
Extent threshold: k = 0 voxels
Expected voxels per cluster, $\langle k \rangle = 12.568$
Expected number of clusters, $\langle c \rangle = 20.52$
FWEp: 4.915, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 367.0]
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
Volume: 1846984 = 230873 voxels = 1982.9 resels
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

T-contrast for neg_story2 showed significant results with the FEW.

neg_story2



Statistics: *p*-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
<i>p</i>	<i>c</i>	<i>p</i> _{FWE-corr}	<i>q</i> _{FDR-corr}	<i>k</i> _E	<i>p</i> _{uncorr}	<i>p</i> _{FWE-corr}	<i>q</i> _{FDR-corr}	<i>T</i>	(<i>Z</i>)	<i>p</i> _{uncorr}	62 -22 12	64 -24 -2	50 -22 6
0.000	3	0.000	0.000	1239	0.000	0.000	0.000	10.75	Inf	0.000	62 -22 12	64 -24 -2	50 -22 6
		0.000	0.000	479	0.000	0.000	0.000	9.21	Inf	0.000	-66 -32 16	-50 -20 2	-56 -22 10
		0.005	0.104	8	0.104	0.013	0.252	5.23	5.14	0.000	4 46 36		

table shows 3 local maxima more than 8.0mm apart

Height threshold: $T = 4.92$, $p = 0.000$ (0.050)

Extent threshold: $k = 0$ voxels

Expected voxels per cluster, $\langle k \rangle = 3.128$

Expected number of clusters, $\langle c \rangle = 0.05$

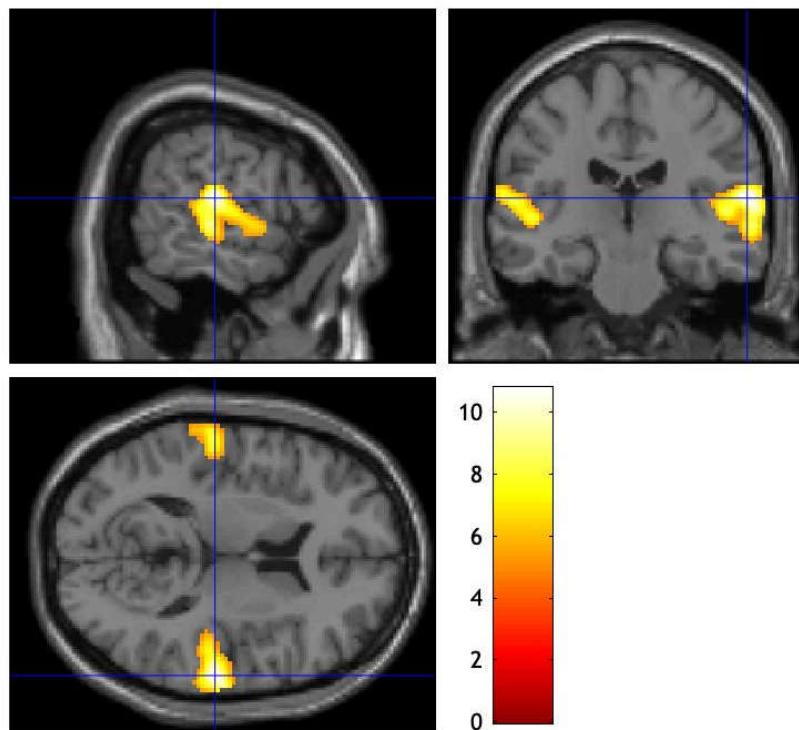
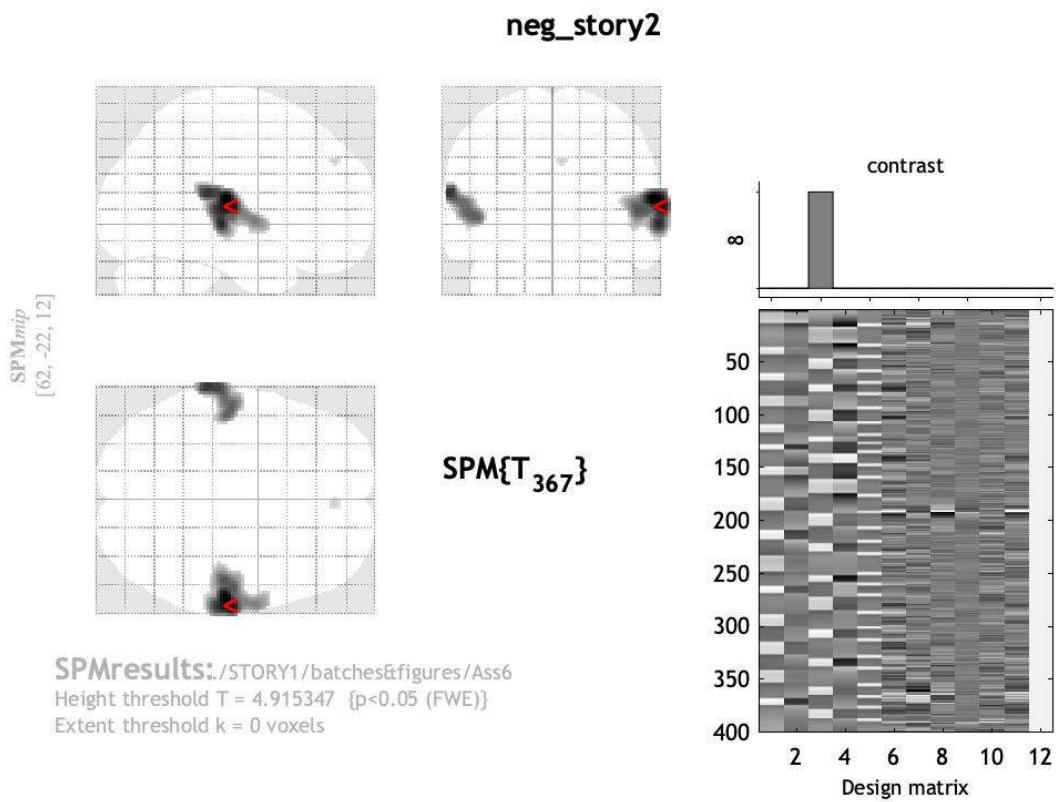
FWEp: 4.915, FDRp: 7.377, FWEc: 8, FDRc: 479

Degrees of freedom = [1.0, 367.0]

FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

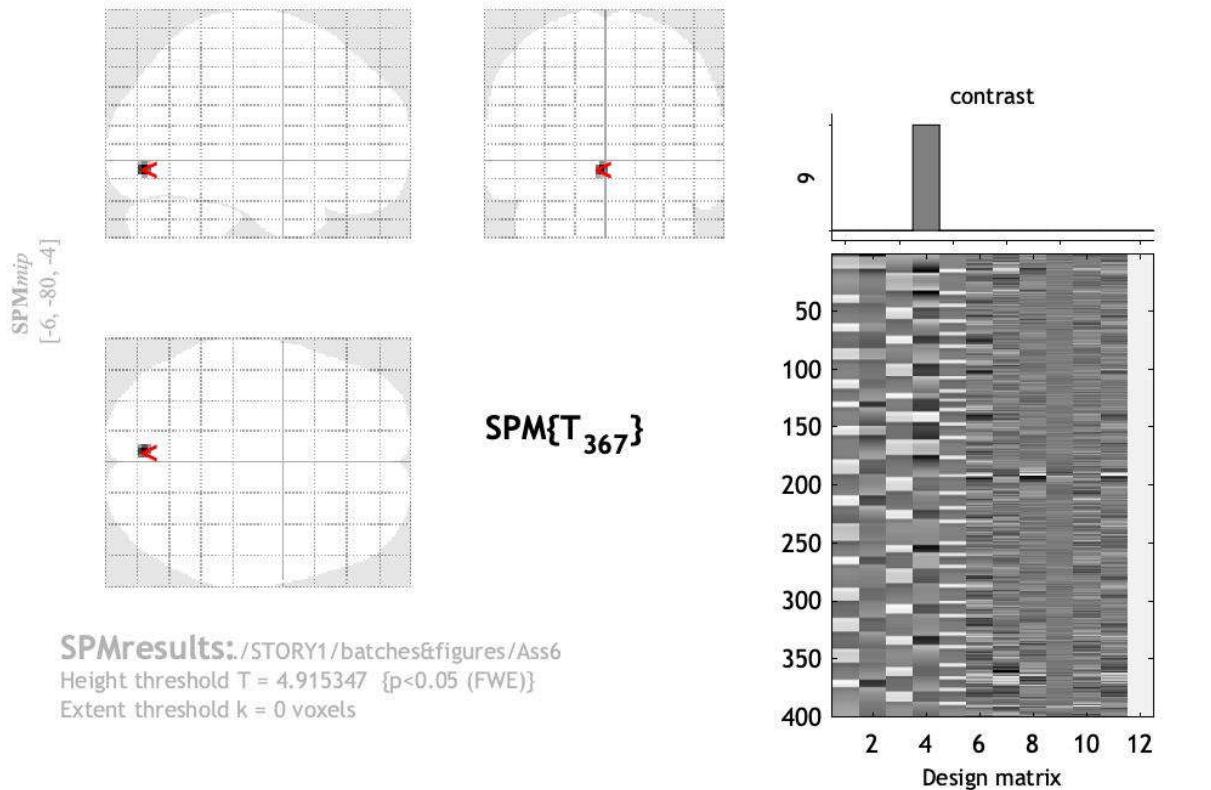
Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)



T-contrast for neg_story2rating showed significant results with the FEW.

neg_story2rating



Statistics: p-values adjusted for search volume

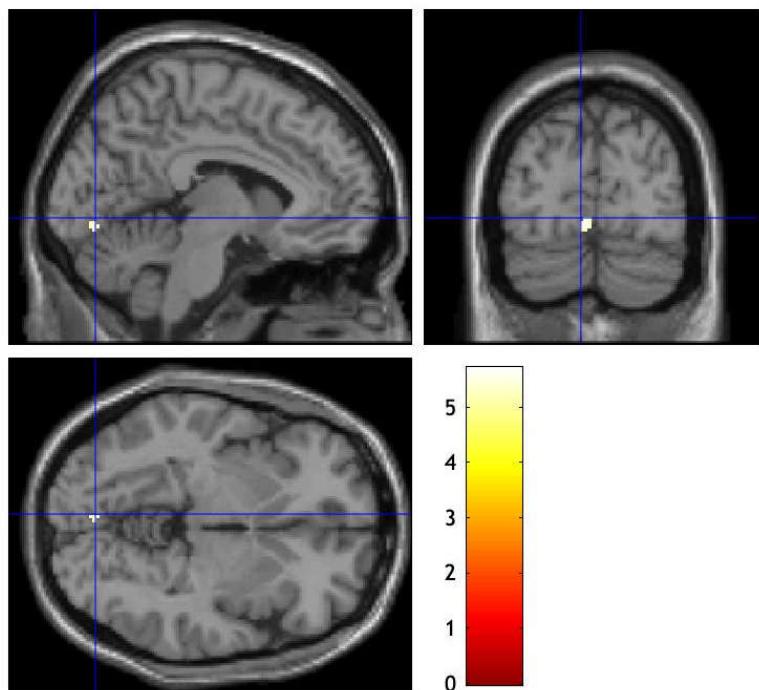
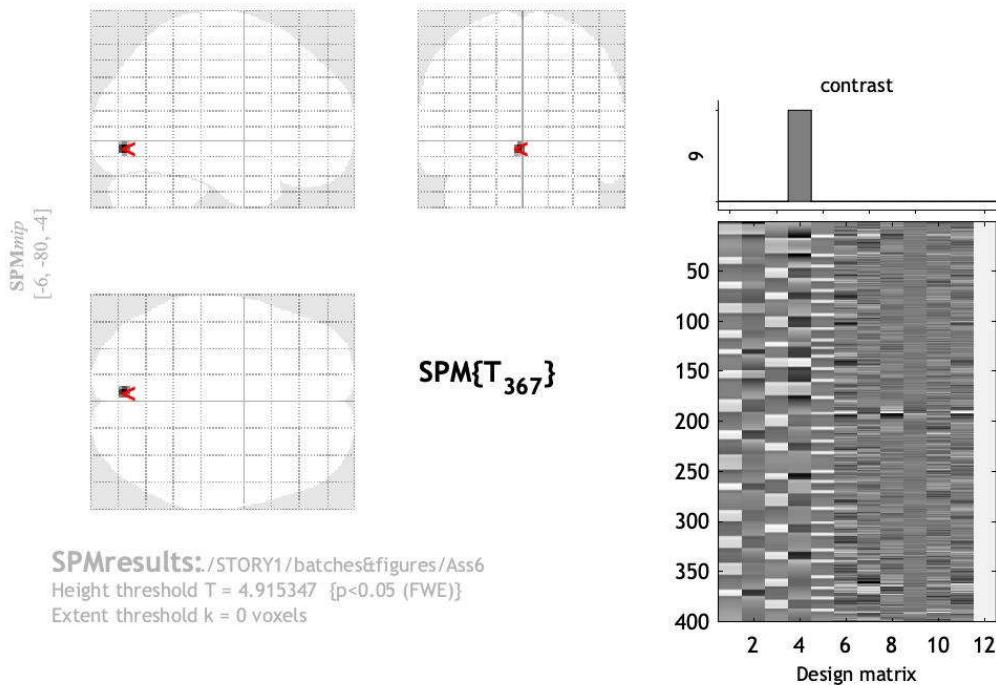
cluster-level				peak-level					mm mm mm		
P _{FWE-corr}	q _{FDR-corr}	k _E	P _{uncorr}	P _{FWE-corr}	q _{FDR-corr}	T	(Z _{max})	P _{uncorr}	-4	-80	-6
0.001	0.010	23	0.010	0.002	0.030	5.69	5.57	0.000			

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 4.92, p = 0.000 (0.050)
 Extent threshold: k = 0 voxels
 Expected voxels per cluster, <k> = 3.128
 Expected number of clusters, <c> = 0.05
 FWEP: 4.915, FDRP: 5.691, FWEC: 23, FDRC: 23

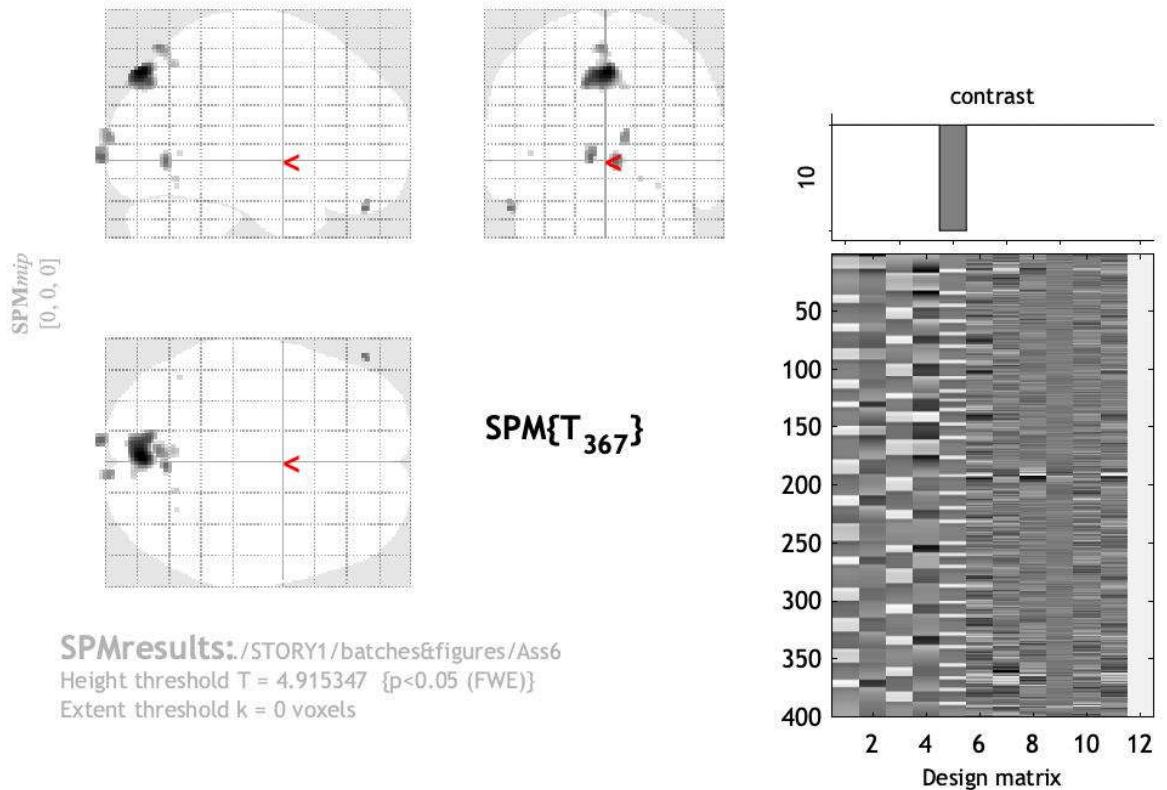
Degrees of freedom = [1.0, 367.0]
 FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
 Volume: 1846984 = 230873 voxels = 1982.9 resels
 Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

neg_story2rating



T-contrast for neg_rating showed significant results with the FEW.

neg_rating



Statistics: *p*-values adjusted for search volume

<i>p</i>	<i>c</i>					<i>P</i> _{FWE-corr}	<i>q</i> _{FDR-corr}	<i>T</i>	(Z _≡)	<i>P</i> _{uncorr}					
0.000	9	0.000	0.000	281	0.000	0.000	0.010	6.40	6.23	0.000	-2	-82	46		
						0.001	0.067	5.90	5.77	0.000	-10	-80	42		
						0.028	0.728	5.05	4.96	0.000	8	-86	38		
						0.002	0.101	5.67	5.55	0.000	-10	-104	0		
						0.002	0.101	5.65	5.53	0.000	4	-68	-2		
						0.002	0.101	5.63	5.51	0.000	-56	44	-28		
						0.003	0.123	5.56	5.44	0.000	-4	-72	60		
						0.017	0.491	5.17	5.07	0.000	-4	-64	50		
						0.005	0.171	5.45	5.34	0.000	10	-100	12		
						0.032	0.760	5.02	4.93	0.000	18	-62	-12		
						0.040	0.870	4.97	4.88	0.000	-44	-60	2		
						0.045	0.898	4.94	4.86	0.000	28	-62	-16		

table shows 3 local maxima more than 8.0mm apart

Height threshold: *T* = 4.92, *p* = 0.000 (0.050)

Extent threshold: *k* = 0 voxels

Expected voxels per cluster, <*k*> = 3.128

Expected number of clusters, <*c*> = 0.05

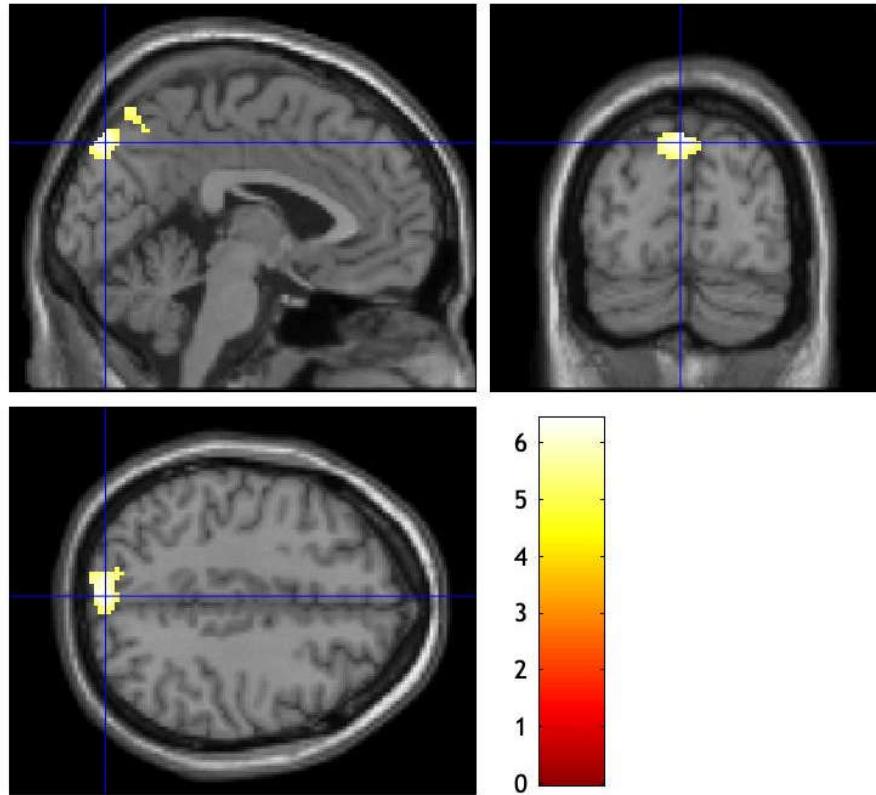
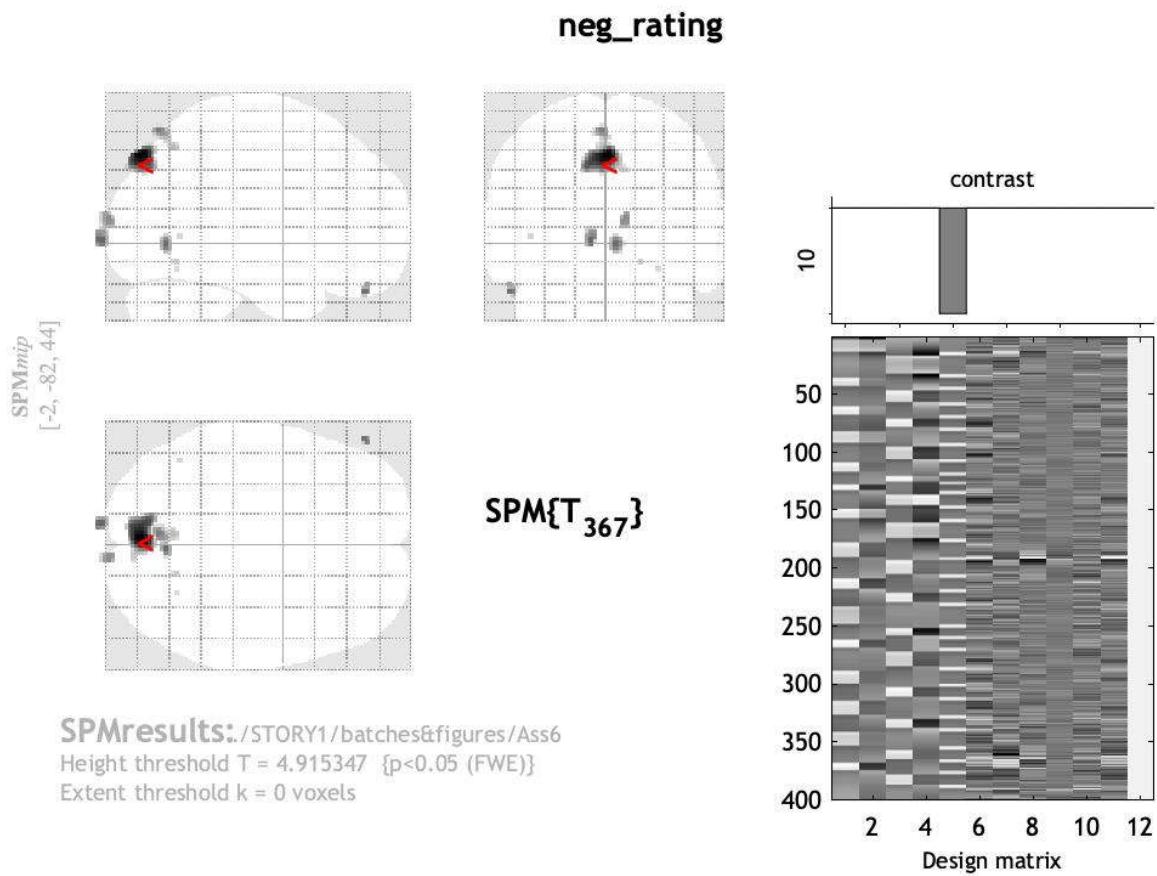
FWEp: 4.915, FDRp: 6.401, FWEc: 1, FDRC: 25

Degrees of freedom = [1.0, 367.0]

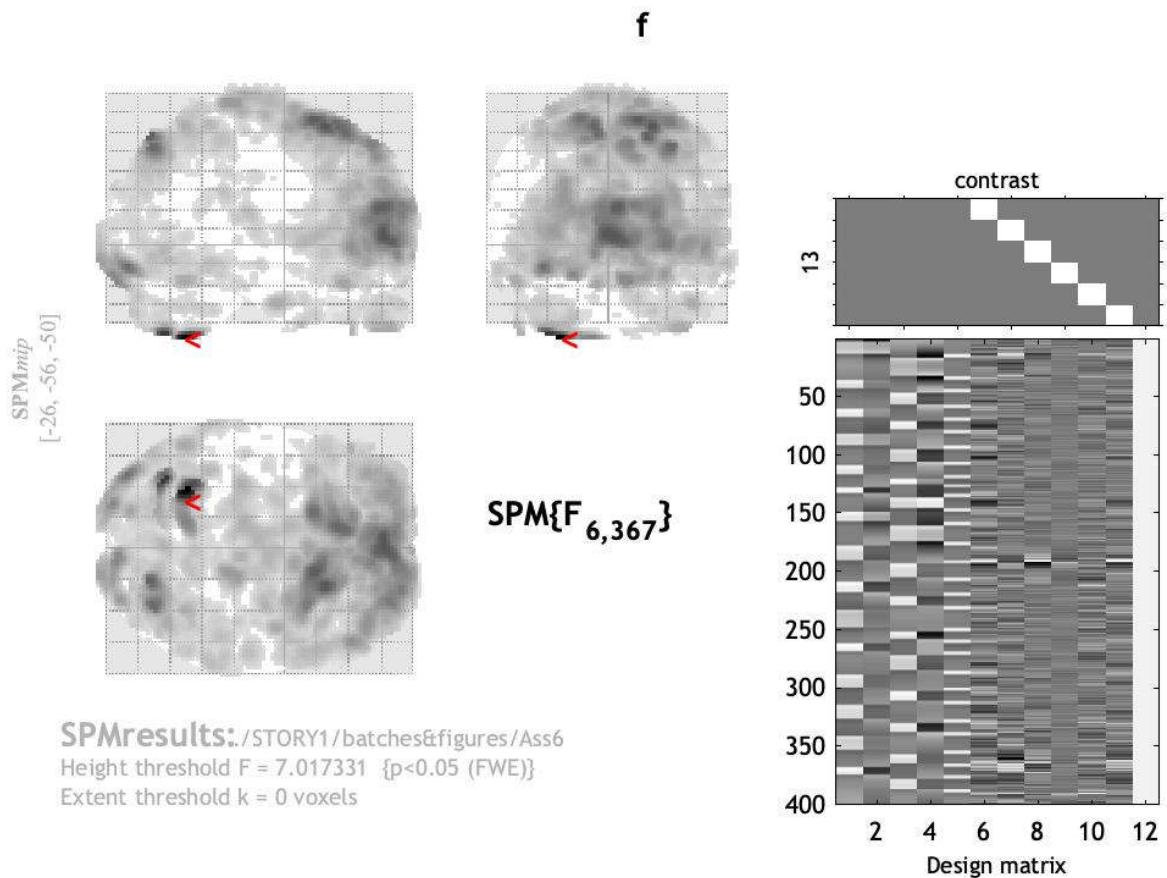
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)



F-contrasts



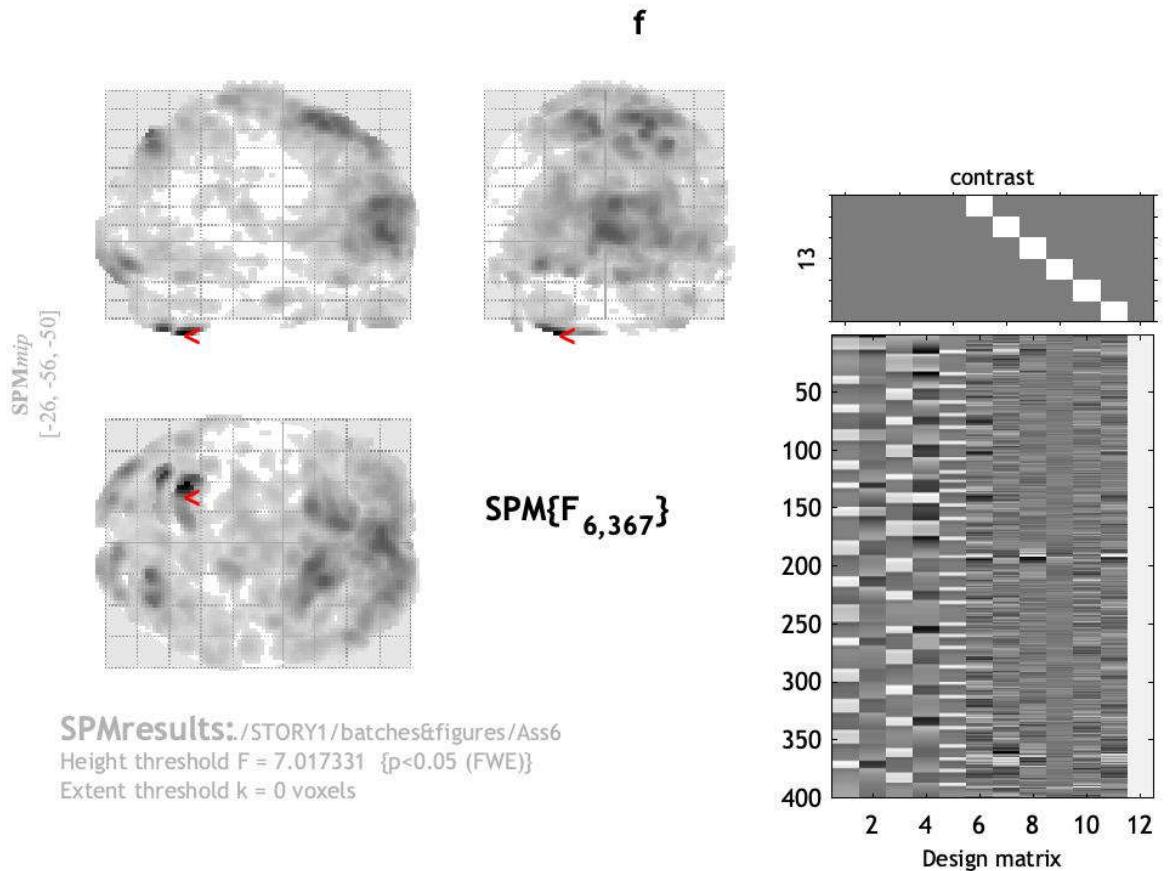
Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	F	(Z_{\equiv})	p_{uncorr}	-30	-56	-52
0.000	71	0.000	0.000	724	0.000	0.000	0.000	56.24	Inf	0.000	-30	-56	-52
						0.000	0.000	41.96	Inf	0.000	-36	-68	-50
						0.000	0.000	26.73	Inf	0.000	-12	-58	-52
						0.000	0.000	36.09	Inf	0.000	22	22	62
						0.000	0.000	34.23	Inf	0.000	16	14	66
						0.000	0.000	30.33	Inf	0.000	-38	-90	-14
						0.000	0.000	25.32	Inf	0.000	-22	-98	-18
						0.000	0.000	19.39	Inf	0.000	-18	-88	-18
						0.000	0.000	25.23	Inf	0.000	10	-96	-22
						0.000	0.000	21.85	Inf	0.000	30	-94	-18
						0.000	0.000	21.03	Inf	0.000	18	-98	-20
						0.000	0.000	18.95	Inf	0.000	-34	56	-48
						0.000	0.000	15.15	Inf	0.000	-24	50	-46
						0.000	0.000	12.19	6.97	0.000	-38	64	-34
						0.000	0.000	18.42	Inf	0.000	46	-6	-34
						0.000	0.000	14.17	7.59	0.000	38	8	-46
						0.000	0.001	10.61	6.41	0.000	46	10	-32
						0.000	0.001	10.18	6.26	0.000	22	50	-38
						0.003	0.070	8.32	5.51	0.000	40	46	-40
						0.000	0.000	16.61	Inf	0.000	-52	36	-50
						0.000	0.000	11.66	6.78	0.000	-52	48	-22
						0.000	0.001	10.31	6.30	0.000	-52	42	-36
						0.000	0.000	15.23	Inf	0.000	44	-68	-10
						0.000	0.000	14.44	7.67	0.000	64	-44	-12
						0.000	0.000	12.14	6.95	0.000	58	-56	-18
						0.000	0.000	14.14	7.58	0.000	58	-22	16

table shows 3 local maxima more than 8.0mm apart

Height threshold: $F = 7.02$, $p = 0.000$ (0.050)
 Extent threshold: $k = 0$ voxels
 Expected voxels per cluster, $\langle k \rangle = 2.028$
 Expected number of clusters, $\langle c \rangle = 0.05$
 FWEP: 7.017, FDRP: 8.493, FWEc: 1, FDRC: 11

Degrees of freedom = [6.0, 367.0]
 FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}
 Volume: 1846984 = 230873 voxels = 1982.9 resels
 Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)
 Page 1



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level				mm mm mm			
p	c	P _{FWE-corr}	q _{FDR-corr}	k _E	P _{uncorr}	P _{FWE-corr}	q _{FDR-corr}	F	(Z ₊)	P _{uncorr}	mm	mm	mm
						0.000	0.000	11.42	6.70	0.000	42	-28	12
						0.000	0.000	10.96	6.54	0.000	42	-18	6
0.000	0.000	224	0.000			0.000	0.000	13.04	7.24	0.000	62	-36	26
						0.000	0.002	10.02	6.19	0.000	52	-32	36
						0.000	0.010	9.27	5.90	0.000	36	-18	34
0.000	0.004	25	0.002			0.000	0.000	12.42	7.04	0.000	4	66	-22
0.000	0.000	304	0.000			0.000	0.000	11.38	6.69	0.000	32	-52	-36
						0.000	0.000	10.72	6.45	0.000	44	-42	-38
						0.000	0.011	9.25	5.89	0.000	52	-62	-34
0.000	0.000	109	0.000			0.000	0.000	11.15	6.61	0.000	-32	16	-40
						0.002	0.055	8.44	5.56	0.000	-44	16	-38
0.000	0.001	32	0.000			0.000	0.000	10.78	6.47	0.000	26	-16	-12
0.001	0.028	14	0.012			0.000	0.001	10.69	6.44	0.000	-10	-98	30
0.000	0.001	33	0.000			0.000	0.001	10.69	6.44	0.000	-50	-24	60
0.000	0.000	52	0.000			0.000	0.001	10.61	6.41	0.000	32	-8	-36
0.000	0.000	45	0.000			0.000	0.003	9.81	6.11	0.000	-16	-58	2
0.000	0.001	34	0.000			0.000	0.004	9.72	6.08	0.000	-42	-12	62
0.000	0.009	20	0.004			0.000	0.008	9.41	5.95	0.000	-44	-12	-38
0.000	0.000	82	0.000			0.000	0.009	9.32	5.92	0.000	36	14	-20
						0.001	0.030	8.73	5.68	0.000	26	4	-16
0.000	0.001	39	0.000			0.000	0.010	9.27	5.90	0.000	12	-76	16
0.000	0.000	47	0.000			0.000	0.013	9.16	5.85	0.000	16	-32	24
						0.007	0.166	7.90	5.33	0.000	16	-22	28
0.000	0.003	27	0.001			0.000	0.014	9.13	5.84	0.000	2	-18	56
0.002	0.068	9	0.038			0.001	0.016	9.05	5.81	0.000	54	36	-38
0.000	0.001	39	0.000			0.001	0.019	8.95	5.77	0.000	40	-38	38
0.000	0.000	77	0.000			0.001	0.019	8.95	5.77	0.000	-2	-44	24
						0.002	0.040	8.59	5.62	0.000	-2	-50	16

table shows 3 local maxima more than 8.0mm apart

Height threshold: F = 7.02, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, <k> = 2.028

Expected number of clusters, <c> = 0.05

FWEp: 7.017, FDRp: 8.493, FWEc: 1, FDRC: 11

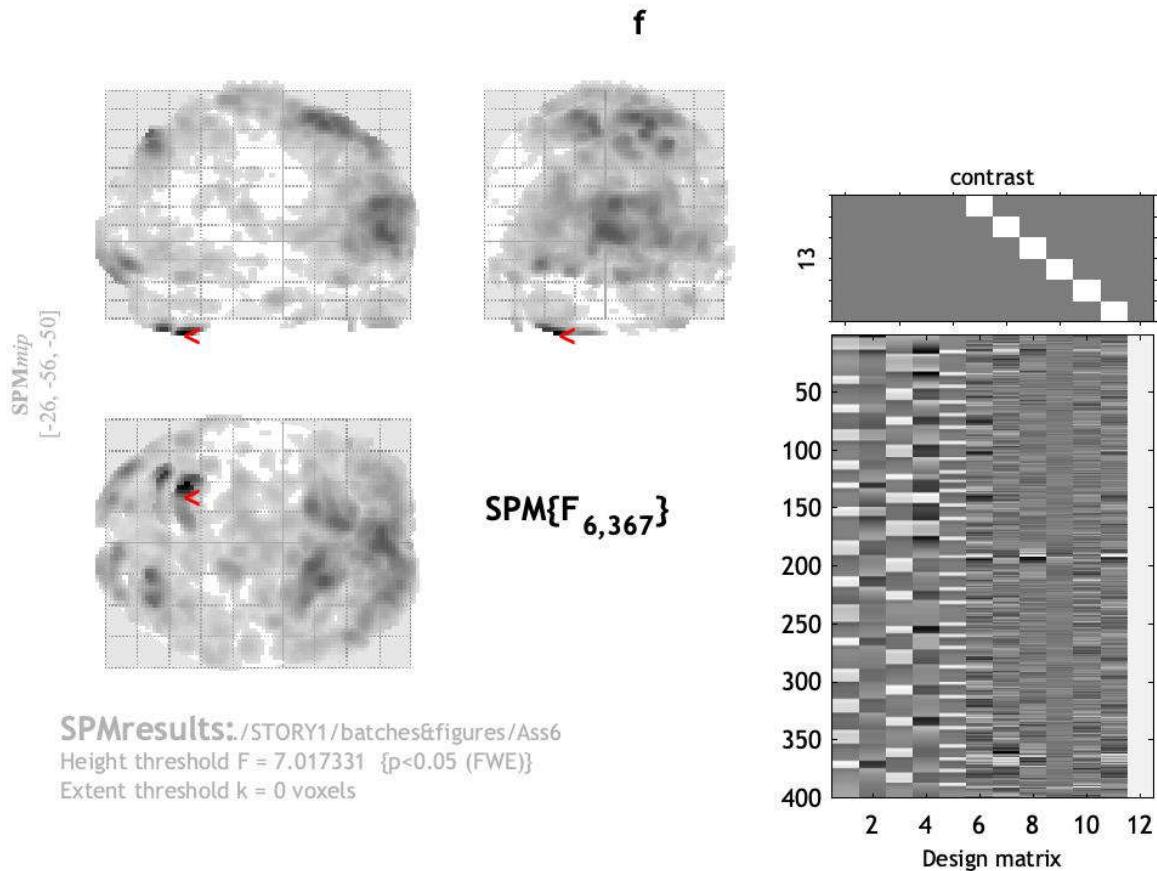
Degrees of freedom = [6.0, 367.0]

FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

Page 2



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level				mm mm mm			
p	c	$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	k_E	P_{uncorr}	$P_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	F	(Z _{max})	P_{uncorr}	mm	mm	mm
		0.000	0.001	32	0.000	0.001	0.023	8.86	5.73	0.000	58	-16	-12
		0.000	0.013	18	0.006	0.001	0.026	8.81	5.71	0.000	62	-32	46
		0.000	0.016	17	0.007	0.001	0.028	8.77	5.70	0.000	-10	-88	-30
		0.001	0.046	11	0.024	0.001	0.028	8.76	5.69	0.000	22	-56	4
		0.000	0.002	31	0.001	0.001	0.038	8.62	5.63	0.000	14	-98	8
		0.003	0.077	8	0.049	0.002	0.046	8.53	5.60	0.000	10	8	-16
		0.004	0.122	6	0.083	0.002	0.047	8.52	5.59	0.000	14	-54	-52
		0.000	0.000	47	0.000	0.002	0.049	8.49	5.58	0.000	-28	-84	20
						0.009	0.201	7.80	5.28	0.000	-36	-84	24
		0.001	0.046	11	0.024	0.002	0.061	8.39	5.54	0.000	68	-28	-22
		0.002	0.068	9	0.038	0.003	0.081	8.25	5.48	0.000	44	-18	-20
		0.003	0.077	8	0.049	0.005	0.113	8.09	5.41	0.000	-42	0	-48
		0.001	0.033	13	0.015	0.005	0.125	8.04	5.39	0.000	-18	-88	28
		0.001	0.046	11	0.024	0.006	0.130	8.02	5.38	0.000	-42	-30	-24
		0.015	0.351	2	0.302	0.007	0.166	7.91	5.33	0.000	-28	24	-24
		0.003	0.077	8	0.049	0.008	0.176	7.87	5.31	0.000	2	24	-6
		0.002	0.068	9	0.038	0.008	0.186	7.84	5.30	0.000	48	-28	0
		0.006	0.160	5	0.110	0.008	0.187	7.84	5.30	0.000	-44	-74	44
		0.011	0.259	3	0.208	0.009	0.190	7.83	5.29	0.000	6	-32	16
		0.004	0.122	6	0.083	0.010	0.229	7.74	5.25	0.000	40	-74	16
		0.001	0.040	12	0.019	0.011	0.245	7.71	5.24	0.000	-30	12	8
		0.011	0.259	3	0.208	0.011	0.251	7.70	5.23	0.000	-58	-12	8
		0.004	0.122	6	0.083	0.012	0.267	7.67	5.22	0.000	16	52	-18
		0.024	0.470	1	0.470	0.014	0.303	7.61	5.19	0.000	52	-78	8
		0.008	0.204	4	0.149	0.018	0.382	7.50	5.14	0.000	-56	28	14
		0.003	0.077	8	0.049	0.018	0.385	7.49	5.14	0.000	26	-34	-24
		0.011	0.259	3	0.208	0.020	0.423	7.45	5.12	0.000	12	-46	-36
		0.011	0.259	3	0.208	0.022	0.464	7.40	5.10	0.000	-52	22	-34

table shows 3 local maxima more than 8.0mm apart

Height threshold: F = 7.02, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 2.028$

Expected number of clusters, $\langle c \rangle = 0.05$

FWeP: 7.017, FDRp: 8.493, FWEc: 1, FDRC: 11

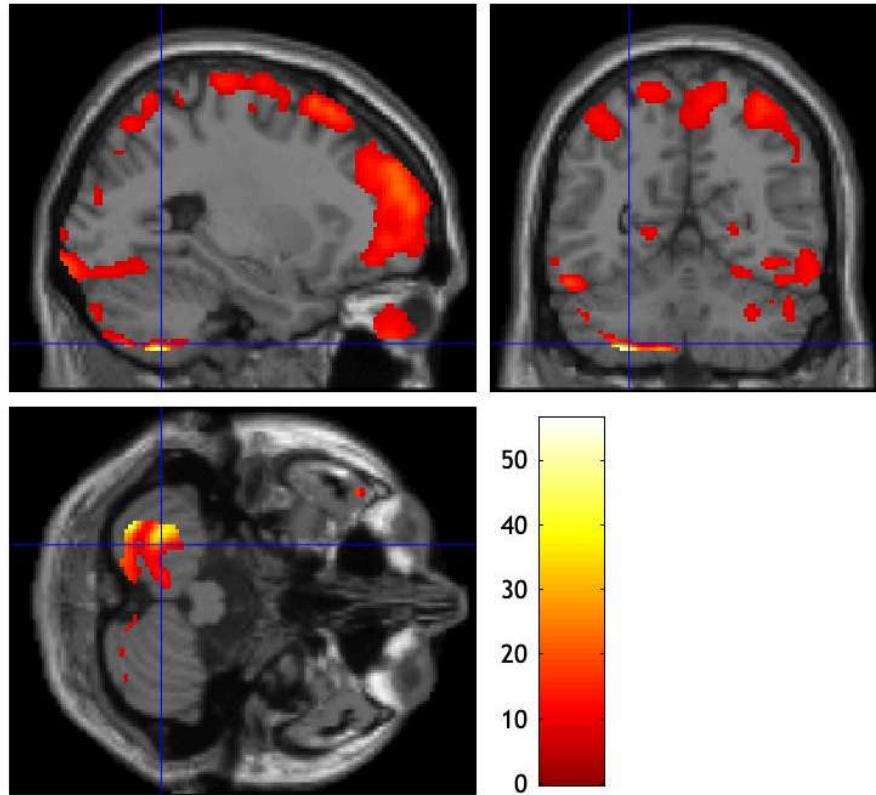
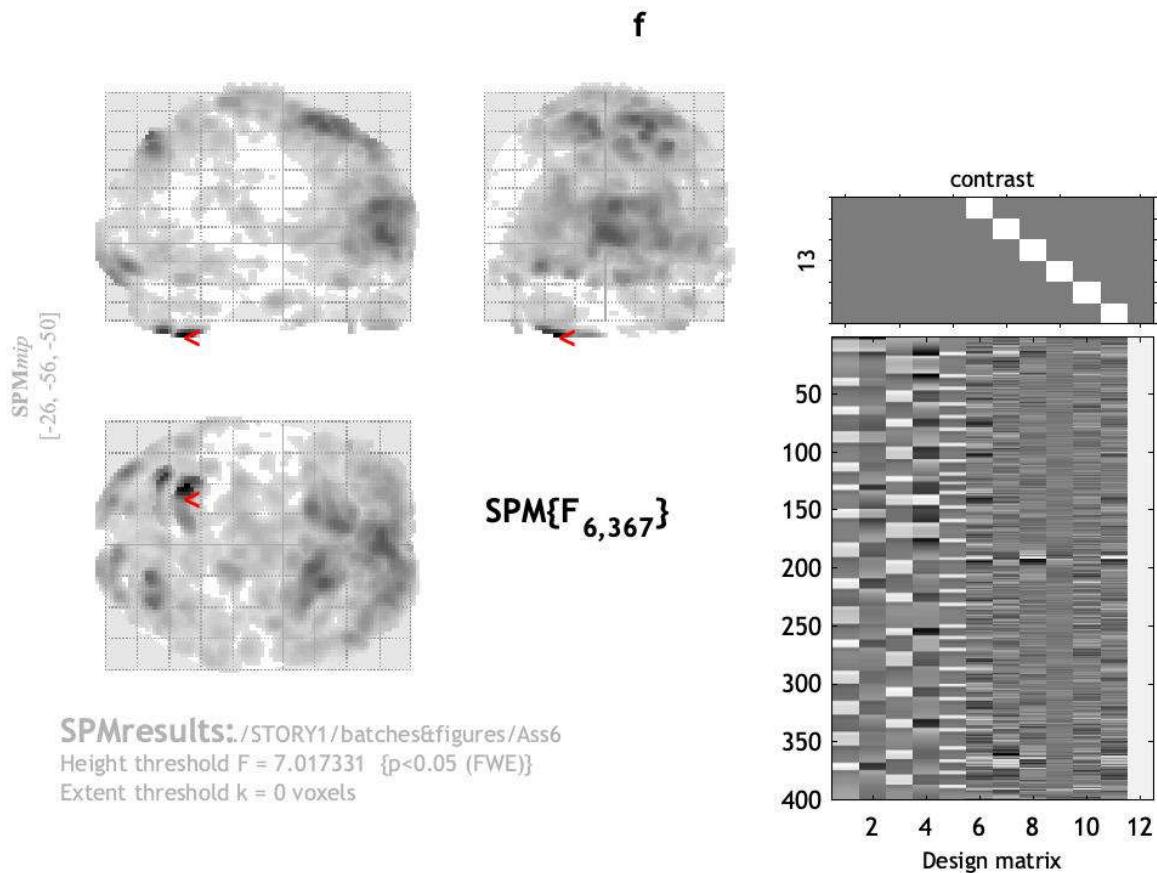
Degrees of freedom = [6.0, 367.0]

FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846984 = 230873 voxels = 1982.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.15 voxels)

Page 3



Voxels included in analysis

$$\text{voxels in total} = 230873$$

If all voxels were independent, the amount of voxels that on average appear to be activated by chance in this analysis would be the uncorrected threshold times the total amount of voxels:

$$\text{voxels in total} \cdot \text{threshold} = 230873 \text{ voxels} \cdot 0.001 \approx 230,873 \text{ voxels}$$

Portfolio 7

Bella Terragni, Daryna Eismont, Peter Thramkrongart and Thea Hølmkjær Kaad

Wed Apr 03 09:35:22 2019

```
##### PART 1 #####
#####
# Emotional Faces Experiment

#0

# set working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Portfolios/7")

# Load Libraries
library(pacman)
p_load(tidyverse,wesanderson,lmerTest, nlme, lme4, pastecs, ggpubr, emmeans)

# Load data from csv into dataframe
face_exp_2016 <- read.csv("face_exp_data_all_160310.csv", sep= ";")
face_exp_2017 <- read.csv("face_exp_all_logs_2017.csv", sep = ";")

# combine the two dataframes into one
face_df <- rbind(face_exp_2016, face_exp_2017)

#1

#1a
# Please explain which factor was between-participants and which were withinparticipants and why.

# frequency is a between-participant factor, since we are analyzing the condition using two separate groups. On the other hand the facial emotion and color are within participant factors, since we are interested in finding out, how each person responds differently to these stimuli.

#1b
# What was the age range of the participants?
# create dataframe with a single row for each participant
age_df <- filter(face_df, no == 1)
```

```

# find mean age from 25 participants
min(age_df$age)

## [1] 19

max(age_df$age)

## [1] 27

mean(age_df$age)

## [1] 21.6

# the mean age is 21.6 with a range from 19-27 yoa.

#2

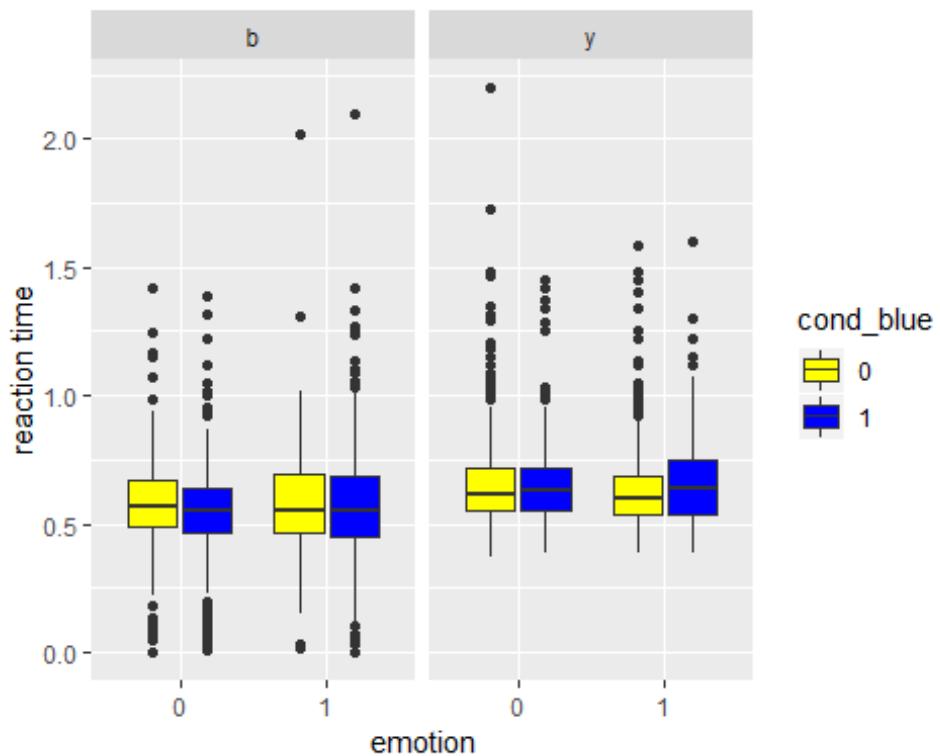
# change conditions into factors
face_df$cond_emo = as.factor(face_df$cond_emo)

face_df$cond_blue = as.factor(face_df$cond_blue)

#2a
# create a box plot displaying the results
boxplot <-
  ggplot(face_df, aes(cond_emo, rt, fill = cond_blue)) + geom_boxplot() + labs(x =
"emotion", y = "reaction time") +
  scale_fill_manual(values = c(
    "yellow",
    "blue",
    "yellow",
    "blue", "yellow",
    "blue", "yellow",
    "blue"
  ))+ facet_wrap(vars(freq))
boxplot

## Warning: Removed 3 rows containing non-finite values (stat_boxplot).

```



#2b

Explain why this plot shows that there is something wrong with the data.

Looking at the box plot there are some very low reaction times that are humanly impossible . This implies that the rt is a mistake response not prompted by the stimuli, but perhaps by reflexes.

#2c

create a subset dataframe containing only correct answers
`correct_df <- filter(face_df, correct_resp == 1)`

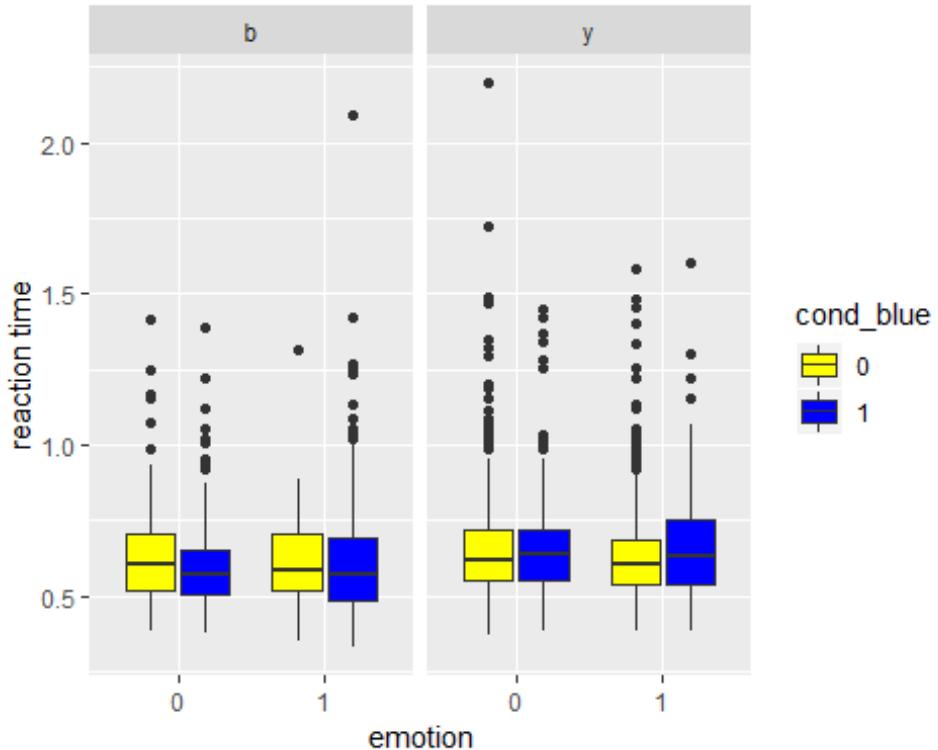
#2d

create a new box plot using only data from correct answers
`boxplot2 <- ggplot(correct_df, aes(cond_emo, rt, fill = cond_blue)) + geom_boxplot() + labs(x = "emotion", y = "reaction time") + scale_fill_manual(values = c("yellow", "blue", "yellow", "blue", "yellow", "blue", "yellow", "blue"))`

```

  "blue"
))+ facet_wrap(vars(freq))
boxplot2

```



Errasing the incorrect trials does delete a Lot of the Low rt anomalies. We will therefore henceforth use the corrected dataframe "correct_df"

```

#2e
# Investigate the three hypotheses based on the descriptive statistics - would you
# expect any of the statistical analyses to be significant based on the descriptive
# stats?

# frequency condition
by(data = correct_df$rt, INDICES = correct_df$freq, FUN = stat.desc)

## correct_df$freq: b
##      nbr.val     nbr.null     nbr.na          min          max
## 8.860000e+02 0.000000e+00 0.000000e+00 3.294620e-01 2.093829e+00
##      range        sum      median        mean       SE.mean
## 1.764367e+00 5.381568e+02 5.840166e-01 6.074004e-01 5.415922e-03
## CI.mean.0.95      var    std.dev   coef.var
## 1.062955e-02 2.598833e-02 1.612090e-01 2.654081e-01
## -----
## correct_df$freq: y

```

```

##      nbr.val     nbr.null     nbr.na       min       max
## 1.319000e+03 0.000000e+00 0.000000e+00 3.679999e-01 2.195470e+00
##      range        sum      median       mean      SE.mean
## 1.827471e+00 8.558316e+02 6.179046e-01 6.488488e-01 4.785901e-03
## CI.mean.0.95    var      std.dev   coef.var
## 9.388815e-03 3.021149e-02 1.738145e-01 2.678814e-01

# The two frequency conditions have a mean of 6.1*e-01 and 6.5*e-01, respectively.
# If we expect the H3 hypothesis to be true, infrequent stimuli will yield longer response times than frequent ones.

The group with the predominantly blue trials seem to have shorter response times than the yellow group in general. We are unsure how to apply this to answer any of the hypothesis.

# color condition
by(data = correct_df$rt, INDICES = correct_df$cond_blue, FUN = stat.desc)

## correct_df$cond_blue: 0
##      nbr.val     nbr.null     nbr.na       min       max
## 1.162000e+03 0.000000e+00 0.000000e+00 3.472731e-01 2.195470e+00
##      range        sum      median       mean      SE.mean
## 1.848197e+00 7.425117e+02 6.021989e-01 6.389946e-01 5.035643e-03
## CI.mean.0.95    var      std.dev   coef.var
## 9.879979e-03 2.946565e-02 1.716556e-01 2.686339e-01
## -----
## correct_df$cond_blue: 1
##      nbr.val     nbr.null     nbr.na       min       max
## 1.043000e+03 0.000000e+00 0.000000e+00 3.294620e-01 2.093829e+00
##      range        sum      median       mean      SE.mean
## 1.764367e+00 6.514766e+02 6.009792e-01 6.246180e-01 5.201635e-03
## CI.mean.0.95    var      std.dev   coef.var
## 1.020687e-02 2.822046e-02 1.679895e-01 2.689475e-01

# The two color conditions have a mean of 6.4*e-01 vs 6.2*e-01, respectively. We expect the H1 hypothesis to be true, since the blue trials (condition 1) seem to have shorter response times than yellow trials (condition 0), according to the aforementioned means. Whether the difference is significant is still doubtful, since the differences in means is relatively small.

# emotion condition
by(data = correct_df$rt, INDICES = correct_df$cond_emo, FUN = stat.desc)

## correct_df$cond_emo: 0
##      nbr.val     nbr.null     nbr.na       min       max
## 1.108000e+03 0.000000e+00 0.000000e+00 3.679999e-01 2.195470e+00
##      range        sum      median       mean      SE.mean

```

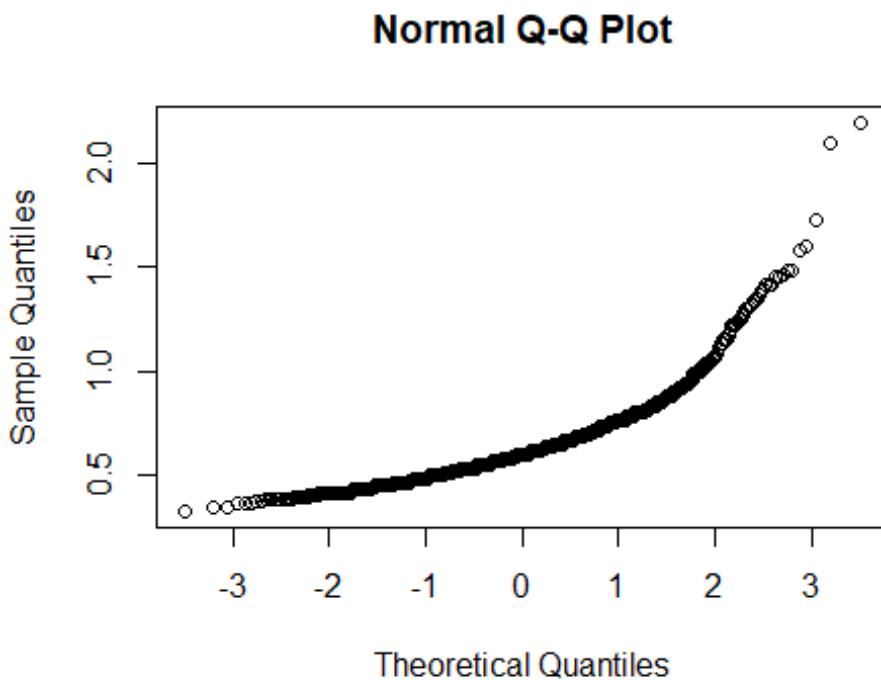
```

## 1.827471e+00 7.039460e+02 6.023024e-01 6.353303e-01 5.109437e-03
## CI.mean.0.95      var     std.dev   coef.var
## 1.002527e-02 2.892583e-02 1.700759e-01 2.676969e-01
## -----
## correct_df$cond_emo: 1
##   nbr.val    nbr.null     nbr.na       min       max
## 1.097000e+03 0.000000e+00 0.000000e+00 3.294620e-01 2.093829e+00
##   range        sum     median       mean      SE.mean
## 1.764367e+00 6.900424e+02 6.009371e-01 6.290268e-01 5.133648e-03
## CI.mean.0.95      var     std.dev   coef.var
## 1.007289e-02 2.891072e-02 1.700315e-01 2.703089e-01

# The two emotive conditions have slightly different means at 6.4*e-01 and 6.3*e-0
1, respectively. The neutral condition (condition 0) has a longer response time th
an the fearful condition (condition 1). This is in agreement with the H2 hypothesi
s, that response time is shorter for fearful faces. The difference between means a
re not particularly large, though, so the significance of the difference remains a
question.

#2f
# explore if the reaction time is normally distributed using a qq-plot
qqnorm(correct_df$rt)

```



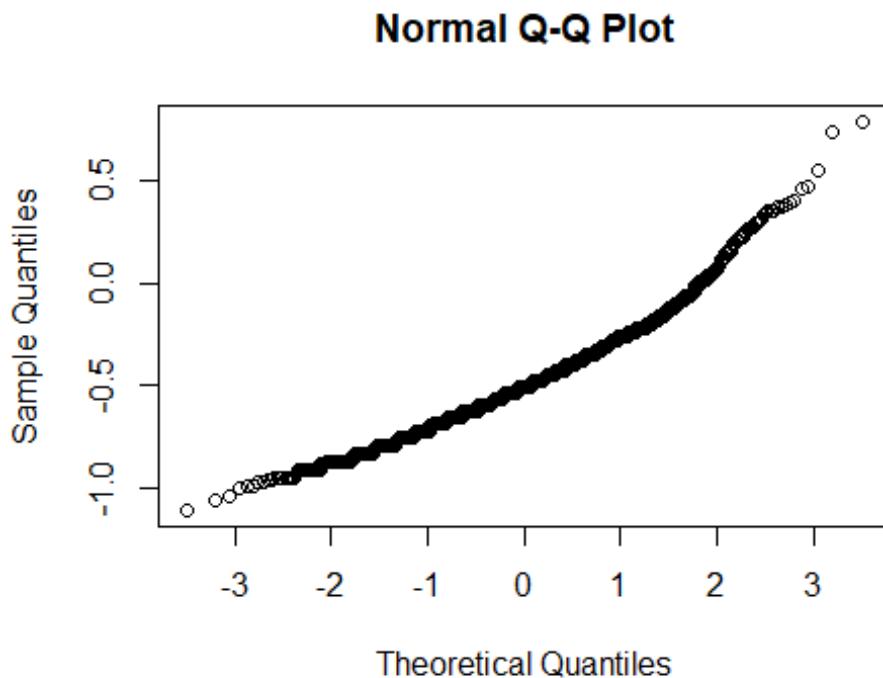
```

# the reaction time is not perfectly normally distributed, which is apparent, since the line in the qq-plot isn't linear.

#2g
# ensure that rt is numeric
correct_df$rt <- as.numeric(correct_df$rt)
# Log transform the reaction time data
correct_df$logtrans <- log(correct_df$rt)

#2h
# Use a qq-plot to explore if the transformed data appear more normal than the untransformed
qqnorm(correct_df$logtrans)

```



```

# the Log transformed data conforms more closely to the assumptions of linearity in a qq-plot.

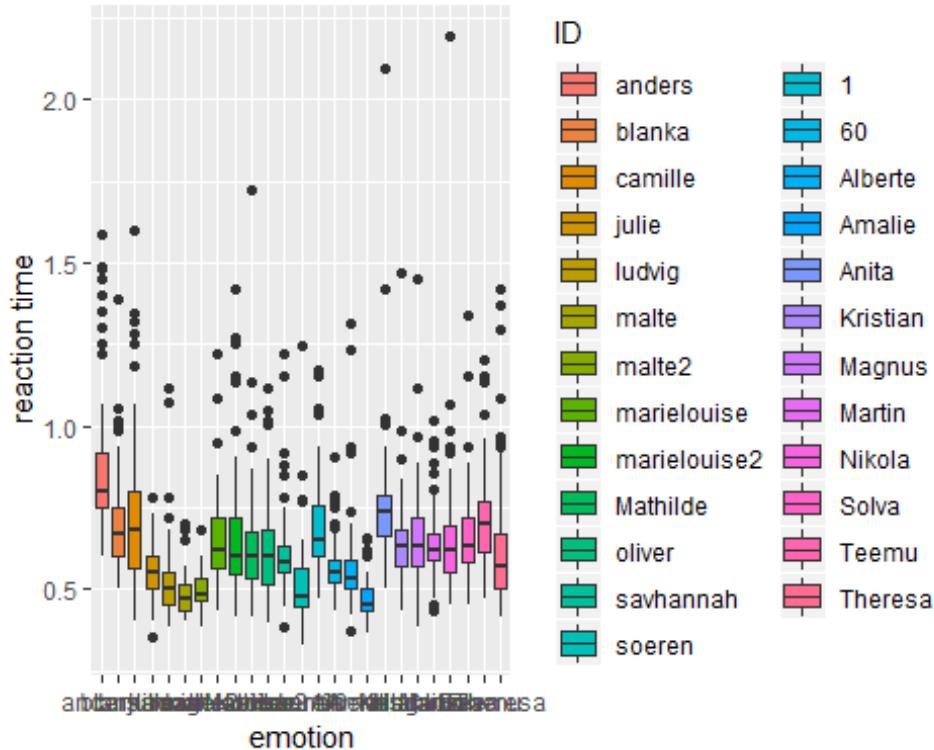
```

```

#2i
# Make a plot that explores the response times for participants, individually, using a box-plot. Does anybody stick out as unusual?
boxplot3 <-
  ggplot(correct_df, aes(ID, rt, fill = ID)) + geom_boxplot() + labs(x = "emotion")

```

```
, y = "reaction time")
boxplot3
```



Looking at the boxplot, there are some outliers in the positive directions. These could be due to practical problems in the experiment. We can see that individuals differ in their range of reaction time, which indicates to us that applying with in-participant measures could improve our model.

#3

#3a

Make mixed effects model where you predict reaction time using the three factors as fixed effects, and include random intercepts for each participant

mixed model with interactions

```
model <- lme(rt ~ freq*cond_emo*cond_blue, random=~1|ID, data = correct_df, method = "ML", na.action = na.exclude)
summary(model)
```

Linear mixed-effects model fit by maximum likelihood

Data: correct_df

AIC BIC logLik

-2137.346 -2080.361 1078.673

```

## 
## Random effects:
##   Formula: ~1 | ID
##             (Intercept) Residual
## StdDev:    0.0843167 0.1455144
##
## Fixed effects: rt ~ freq * cond_emo * cond_blue
##                               Value Std.Error DF t-value p-value
## (Intercept)              0.6225921 0.02833777 2174 21.970398 0.0000
## freqy                   0.0320796 0.03688862   23  0.869633 0.3935
## cond_emo1                -0.0256658 0.01740680 2174 -1.474468 0.1405
## cond_blue1               -0.0414825 0.01492333 2174 -2.779709 0.0055
## freqy:cond_emo1          0.0040625 0.01998750 2174  0.203250 0.8390
## freqy:cond_blue1         0.0430997 0.01915627 2174  2.249902 0.0246
## cond_emo1:cond_blue1    0.0455863 0.02106492 2174  2.164084 0.0306
## freqy:cond_emo1:cond_blue1 -0.0232987 0.02709941 2174 -0.859749 0.3900
## Correlation:
##                               (Intr) freqy  cnd_m1 cnd_b1 frqy:cnd_m1
## freqy                   -0.768
## cond_emo1                -0.311  0.239
## cond_blue1                -0.363  0.279  0.589
## freqy:cond_emo1           0.271 -0.273 -0.871 -0.513
## freqy:cond_blue1          0.283 -0.285 -0.459 -0.779  0.525
## cond_emo1:cond_blue1     0.256 -0.197 -0.826 -0.707  0.719
## freqy:cond_emo1:cond_blue1 -0.199  0.201  0.642  0.550 -0.737
##                               frqy:cnd_b1 c_1:_1
## freqy
## cond_emo1
## cond_blue1
## freqy:cond_emo1
## freqy:cond_blue1
## cond_emo1:cond_blue1      0.551
## freqy:cond_emo1:cond_blue1 -0.706      -0.777
##
## Standardized Within-Group Residuals:
##      Min        Q1        Med        Q3        Max
## -2.1325397 -0.5515993 -0.1511939  0.3101766 10.5555973
##
## Number of Observations: 2205
## Number of Groups: 25

#3b
# Report the t-statistics using summary()
coefficients(summary(model))

##                               Value Std.Error DF t-value
## (Intercept)              0.622592062 0.02833777 2174 21.9703980

```

```

## freq                                0.032079555 0.03688862   23  0.8696329
## cond_emo1                            -0.025665767 0.01740680 2174 -1.4744681
## cond_blue1                           -0.041482503 0.01492333 2174 -2.7797090
## freq:cond_emo1                      0.004062464 0.01998750 2174  0.2032502
## freq:cond_blue1                      0.043099740 0.01915627 2174  2.2499020
## cond_emo1:cond_blue1                 0.045586261 0.02106492 2174  2.1640843
## freq:cond_emo1:cond_blue1           -0.023298702 0.02709941 2174 -0.8597493
##                                         p-value
## (Intercept)                         8.785989e-97
## freq                                 3.934864e-01
## cond_emo1                            1.405004e-01
## cond_blue1                           5.487591e-03
## freq:cond_emo1                      8.389585e-01
## freq:cond_blue1                      2.455481e-02
## cond_emo1:cond_blue1                3.056646e-02
## freq:cond_emo1:cond_blue1          3.900221e-01

# the results of the t-test show a non-significant effect of frequency  $t(23) = 0.2$ 
#  $03$ ,  $p\text{-value} > .3$ , a non-significant effect of emotion  $t(2174) = -1.47$ ,  $p\text{-value} > .1$ ,
# and a significant effect of color  $t(2174) = -2.78$ ,  $p\text{-value} < .01$ .
Investigating the interaction effects, there seems to be an insignificant effect between color condition and emotion  $t(2174) = 2.25$ ,  $p\text{-value} > 0.5$ . On the other hand, we found a significant interaction effect of frequency and color condition  $t(2174) = 2.25$ ,  $p\text{-value} < .05$ , as well as a significant interaction effect between emotive condition and color condition,  $t(2174) = 2.16$ ,  $p\text{-value} < .05$ . The three-way interaction frequency:emotion:color yielded insignificant results,  $t(2174) = -0.860$ ,  $p\text{-value} > 0.1$ .

```

```

#3c
# Report the F-statistics using anova() and a type='I' analysis
anova(model, type = "sequential")

##                                         numDF denDF  F-value p-value
## (Intercept)                         1     2174 1325.8126 <.0001
## freq                               1      23    2.1949  0.1520
## cond_emo                            1     2174    1.0146  0.3139
## cond_blue                           1     2174    0.0016  0.9685
## freq:cond_emo                      1     2174    2.4620  0.1168
## freq:cond_blue                      1     2174    5.3329  0.0210
## cond_emo:cond_blue                  1     2174    5.6532  0.0175
## freq:cond_emo:cond_blue            1     2174    0.7392  0.3900

# According to the analysis, which is of "type 1", and therefore sequential, there
# are no significant effects of frequency ( $F(1, 23) = 2.195$ ,  $p\text{-value} > .1$ ), emotion
# ( $F(1, 2174) = 1.01$ ,  $p\text{-value} > .3$ ) or color ( $F(1, 2174) = 0.002$ ,  $p\text{-value} > .9$ ).
# There was likewise no significant interaction effect between frequency and emotive
# condition  $F(1, 2174) = 2.46$ ,  $p\text{-value} > .1$ .

```

On the other hand, the analysis found a significant interaction effect between frequency and color condition, $F(1, 2174) = 5.33$, $p\text{-value} < .05$, as well as emotional condition and color condition, $F(1, 2174) = 5.65$, $p\text{-value} < .05$.

Lastly, the three-way interaction effect between color, frequency and emotion was insignificant, $F(1, 2174) = 0.739$, $p\text{-value} > .3$.

#3d

```
# Report the F-statistics using anova() and type='marginal'  
anova(model, type = "marginal")
```

```
##                                     numDF denDF  F-value p-value  
## (Intercept)                  1    2174 482.6984 <.0001  
## freq                      1     23   0.7563  0.3935  
## cond_emo                   1    2174   2.1741  0.1405  
## cond_blue                  1    2174   7.7268  0.0055  
## freq:cond_emo              1    2174   0.0413  0.8390  
## freq:cond_blue              1    2174   5.0621  0.0246  
## cond_emo:cond_blue         1    2174   4.6833  0.0306  
## freq:cond_emo:cond_blue    1    2174   0.7392  0.3900
```

According to the analysis, which is of “type 3”, and therefore marginal, there are no significant effects of frequency ($F(1, 23) = 0.75$, $p\text{-value} > .3$) and emotion ($F(1, 2174) = 2.17$, $p\text{-value} > .1$). The color condition yielded significant results ($F(1, 2174) = 7.72$, $p\text{-value} < .05$). There was no significant interaction effect between frequency and emotional condition $F(1, 2174) = 0.04$, $p\text{-value} > .8$.

On the other hand, the analysis found a significant interaction effect between frequency and color condition, $F(1, 2174) = 5.06$, $p\text{-value} < .05$, as well as emotional condition and color condition, $F(1, 2174) = 4.68$, $p\text{-value} < .05$.

Lastly, the three-way interaction effect between color, frequency and emotion was insignificant, $F(1, 2174) = 0.739$, $p\text{-value} > .3$.

We see a difference between the results of the sequential and marginal anova. This is due to sequential or type 1 analysis attributing all shared variance to the first covariate, while the marginal or type 3 analysis assumes no shared variance, and is therefore not dependent on the order.

#3e

```
# Make a new model including a random slope from trial number
```

```
# create mixed model with interactions containing ID as random intercept, while trial number is added as random slope  
model2 <- lme(rt ~ freq*cond_emo*cond_blue, random = ~ no|ID, data = correct_df, method = "ML", na.action = na.exclude)  
summary(model2)
```

```

## Linear mixed-effects model fit by maximum likelihood
## Data: correct_df
##      AIC      BIC logLik
## -2155.601 -2087.219 1089.8
##
## Random effects:
## Formula: ~no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev     Corr
## (Intercept) 0.0854302431 (Intr)
## no          0.0007812995 -0.246
## Residual    0.1438851695
##
## Fixed effects: rt ~ freq * cond_emo * cond_blue
##                         Value Std.Error DF t-value p-value
## (Intercept)            0.6166732 0.02805559 2174 21.980399 0.0000
## freqy                 0.0394512 0.03649854   23  1.080898 0.2909
## cond_emo1              -0.0257103 0.01725856 2174 -1.489714 0.1364
## cond_blue1             -0.0408069 0.01479698 2174 -2.757786 0.0059
## freqy:cond_emo1        0.0034605 0.01982423 2174  0.174557 0.8614
## freqy:cond_blue1       0.0419984 0.01900181 2174  2.210231 0.0272
## cond_emo1:cond_blue1   0.0473595 0.02087944 2174  2.268236 0.0234
## freqy:cond_emo1:cond_blue1 -0.0261979 0.02688034 2174 -0.974613 0.3299
## Correlation:
##                         (Intr) freqy  cnd_m1 cnd_b1 frqy:cnd_m1
## freqy                  -0.769
## cond_emo1              -0.310  0.238
## cond_blue1             -0.366  0.282  0.587
## freqy:cond_emo1        0.270 -0.273 -0.871 -0.511
## freqy:cond_blue1       0.285 -0.288 -0.457 -0.779  0.524
## cond_emo1:cond_blue1   0.258 -0.198 -0.825 -0.706  0.718
## freqy:cond_emo1:cond_blue1 -0.200  0.203  0.641  0.548 -0.736
##                         frqy:cnd_b1 c_1:_1
## freqy
## cond_emo1
## cond_blue1
## freqy:cond_emo1
## freqy:cond_blue1
## cond_emo1:cond_blue1      0.550
## freqy:cond_emo1:cond_blue1 -0.705      -0.777
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -2.0061538 -0.5565794 -0.1530110  0.3038404 10.4244362
##
## Number of Observations: 2205
## Number of Groups: 25

```

```

# report t-statistics using summary() as seen in 3b
coefficients(summary(model2))

##                                     Value Std.Error DF   t-value
## (Intercept)                 0.61667317 0.02805559 2174 21.9803992
## freqy                      0.03945120 0.03649854  23  1.0808982
## cond_emo1                  -0.02571032 0.01725856 2174 -1.4897139
## cond_blue1                 -0.04080690 0.01479698 2174 -2.7577865
## freqy:cond_emo1            0.00346046 0.01982423 2174  0.1745571
## freqy:cond_blue1           0.04199839 0.01900181 2174  2.2102308
## cond_emo1:cond_blue1       0.04735950 0.02087944 2174  2.2682365
## freqy:cond_emo1:cond_blue1 -0.02619792 0.02688034 2174 -0.9746129
##                                     p-value
## (Intercept)                7.337174e-97
## freqy                     2.909463e-01
## cond_emo1                  1.364445e-01
## cond_blue1                 5.868095e-03
## freqy:cond_emo1            8.614439e-01
## freqy:cond_blue1           2.719296e-02
## cond_emo1:cond_blue1       2.341249e-02
## freqy:cond_emo1:cond_blue1 3.298607e-01

# the results of the t-test show a non-significant effect of frequency, t(23) = 1.08, p-value > .05, a non-significant effect of emotion, t(2174) = -1.48, p-value > .05, and a significant effect of color, t(2174) = -2.75, p-value < .01.
# Investigating the interaction effects, there seems to be a significant interaction effect of frequency and color condition t(2174) = 0.174, p-value < .05, as well as a significant interaction effect between emotive condition and color condition t(2174) = 2.27, p-value < .05. There were no interaction effects between frequency and emotion, t(2174) = 0.17, p-value > .5. The threefold interaction frequency:emotion:color yielded insignificant results, t(2174) = -0.97, p-value > .3.

# the inclusion of the random slope did not change the significance of the covariates, but the p-values differ slightly from those in 3b. The model does not seem to be improved according to this analysis.

# The random slope takes into account that people will react differently to the effect of time on the experiment. This helps remove noisy variance due to individual differences in e.g. in their need for novelty or how their tiredness progresses.

#3f
# Make a model comparison of model 3.a and 3.e using anova().
anova(model, model2)

##      Model df     AIC     BIC logLik   Test L.Ratio p-value
## model     1 10 -2137.346 -2080.361 1078.673
## model2    2 12 -2155.601 -2087.219 1089.800 1 vs 2 22.25427 <.0001

```

according to the anova where we compare the model with and without random slope, the incorporation of random slope significantly improves the model, Likelihood-Ratio(2) = 22.3, p-value < .0001.

#3g

Response times are correlated in time which goes against the assumption of independence. We will therefore try to include a so-called auto-regressive component in the model.

```
model3 <- lme(rt ~ freq*cond_emo*cond_blue, random = ~ no | ID, cor=corAR1(), form=~1 | ID), data = correct_df, method = "ML", na.action = na.exclude)
summary(model3)

## Linear mixed-effects model fit by maximum likelihood
## Data: correct_df
##          AIC      BIC      logLik
## -2178.676 -2104.595 1102.338
##
## Random effects:
## Formula: ~no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev     Corr
## (Intercept) 0.0849782730 (Intr)
## no          0.0007389268 -0.232
## Residual    0.1445441591
##
## Correlation Structure: AR(1)
## Formula: ~1 | ID
## Parameter estimate(s):
##   Phi
## 0.1234472
## Fixed effects: rt ~ freq * cond_emo * cond_blue
##                  Value Std.Error DF t-value p-value
## (Intercept) 0.6182015 0.02807205 2174 22.021959 0.0000
## freqy       0.0415594 0.03655247  23  1.136980 0.2673
## cond_emo1  -0.0275362 0.01697363 2174 -1.622294 0.1049
## cond_blue1 -0.0398698 0.01453090 2174 -2.743791 0.0061
## freqy:cond_emo1 0.0025412 0.01951340 2174  0.130230 0.8964
## freqy:cond_blue1 0.0367991 0.01871400 2174  1.966393 0.0494
## cond_emo1:cond_blue1 0.0487635 0.02052286 2174  2.376058 0.0176
## freqy:cond_emo1:cond_blue1 -0.0245981 0.02642545 2174 -0.930849 0.3520
## Correlation:
##                  (Intr) freqy  cnd_m1 cnd_b1 frqy:cnd_m1
## freqy            -0.768
## cond_emo1        -0.304  0.234
## cond_blue1       -0.358  0.275  0.583
## freqy:cond_emo1  0.265 -0.268 -0.870 -0.507
## freqy:cond_blue1 0.278 -0.281 -0.453 -0.776  0.521
```

```

## cond_emo1:cond_blue1      0.252 -0.193 -0.822 -0.706  0.715
## freqy:cond_emo1:cond_blue1 -0.196  0.199  0.638  0.548 -0.734
##                               frqy:cnd_b1 c_1:_1
## freqy
## cond_emo1
## cond_blue1
## freqy:cond_emo1
## freqy:cond_blue1
## cond_emo1:cond_blue1      0.548
## freqy:cond_emo1:cond_blue1 -0.706      -0.777
##
## Standardized Within-Group Residuals:
##      Min       Q1       Med       Q3       Max
## -2.0185321 -0.5603134 -0.1591011  0.3036637 10.3505386
##
## Number of Observations: 2205
## Number of Groups: 25

anova(model, model2, model3)

##          Model df     AIC     BIC   logLik   Test L.Ratio p-value
## model      1 10 -2137.346 -2080.361 1078.673
## model2     2 12 -2155.601 -2087.219 1089.800 1 vs 2 22.25427 <.0001
## model3     3 13 -2178.676 -2104.595 1102.338 2 vs 3 25.07505 <.0001

# the model is improved by adding the auto-regressive component. Model 3 is the best fitting model according to the anova comparison analysis, Likelihood-Ratio(1) = 25.1, p-value < .0001.

#4

#4a
# Comprehension question. If you were to report these results, which model would you use and why?

# When considering which model to report, we want to use a model that balances between extensiveness and simplicity. Extensiveness ensures that we model all confounds or covariates, while simplicity ensures that our results are interpretable. We believe that adding random slope, and an auto-regressive component resulted in significant, sensible improvements that make the model 3 the most preferable.

#4b
# Throughout part 3 of this exercise we made several models to choose from. What is the problem of this strategy?

# One could argue that we are p-hacking, since we are investigating many different models. Since there is a 5 % risk of a false positive, the more models we test out

```

, the bigger chance that one of them is a false positive. This is especially problematic, if we chose to report only the best one of our models, without exposing the discarded models.

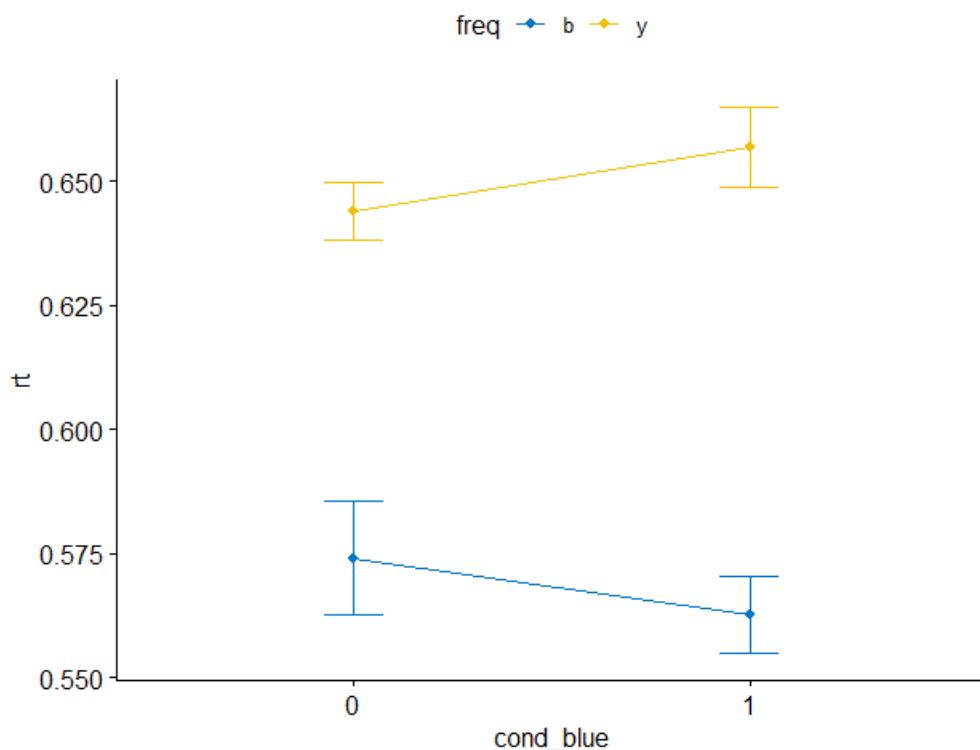
#4c

Write a few lines, briefly stating the results of the experiment in relation to the hypotheses, using the model you decided upon in 4.a.

The results from model 3 support hypothesis 1, that the blue trials have a shorter response time than the yellow trials, because of the negative slope, $\beta = -0.03$, $t(2174) = -2.74$, $p - \text{value} < .01$.

The results also support hypothesis 2, that the fearful faces will yield a shorter response time than neutral faces, because of the negative slope, $\beta = -0.02$, $t(2174) = -1.62$, $p - \text{value} < .05$.

```
ggline(face_df, x = "cond_blue", y = "rt", col='freq',
       add = c("mean_se", "dodge"), palette = "jco")
```



Looking at the plot, we can see that infrequent stimuli yield longer response times than frequent stimuli, since 0 is yellow and 1 is blue. The group with a high blue frequency were quicker at reacting to blue stimuli than yellow stimuli, and the opposite is true for the high yellow frequency group. These results are also evident in the stats, $\beta = -0.04$, $t(2174) 1.97$, $p - \text{value} < .05$.

```
#####
##### PART 2 #####
#####

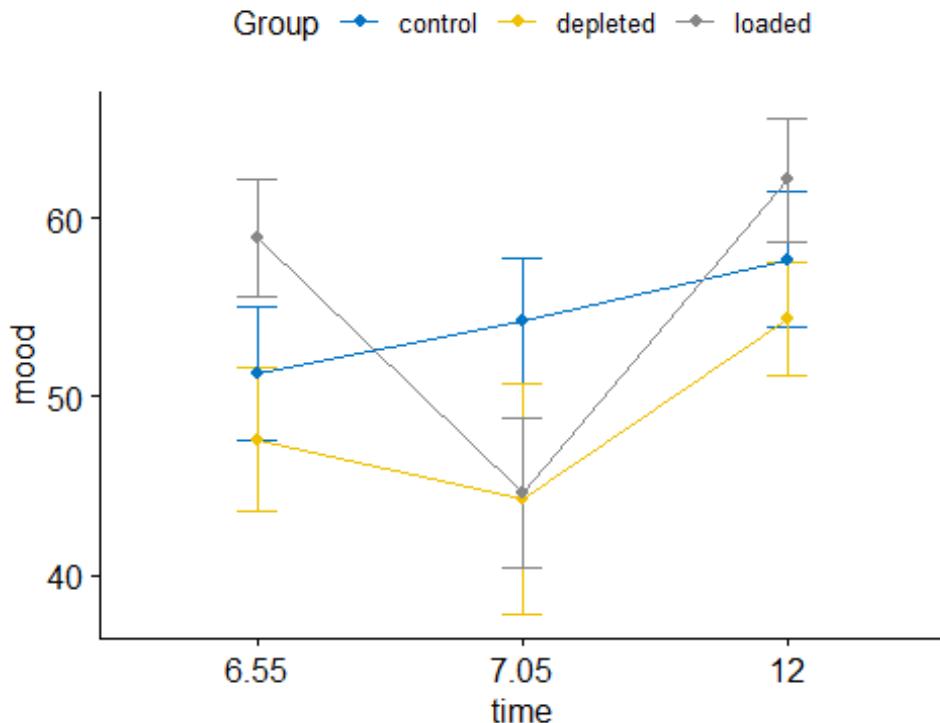
# Tryptophan Depletion Study

#5

#5a
# Load data from csv file into R as a dataframe
trypt_long <- read.csv(file = 'trypt_long.csv', header = TRUE, sep = ",")

# make ID number and time point factorial instead of continuous
trypt_long$ID<-as.factor(trypt_long$ID)
trypt_long$time<-as.factor(trypt_long$time)

# use ggline to make a Line plot displaying time on the x-axis and mood on the y-axis
ggline(trypt_long, x = "time", y = "mood", col='Group',
       add = c("mean_se", "dodge"), palette = "jco")
```



```
# Relevel to make the reference group "Loaded"
trypt_long$Group <- relevel(trypt_long$Group, 'loaded')

# Relevel to make the reference time "7.05"
trypt_long$time <- relevel(trypt_long$time, '7.05')
```

```

# Make mixed effects model with Group and time as fixed effects and ID as random effect
trypt_model <- lmerTest::lmer(mood~Group*time+(1|ID), data = trypt_long)

# Get summary statistics
trypt_res<-summary(trypt_model)

# Apply Bonferroni correction for multiple comparisons to p-values (9 tests) and round a bit (5 decimals)
trypt_res$coefficients2 <- matrix(round(c(trypt_res$coefficients,trypt_res$coefficients[,5]*9),
                                         digits=5),ncol=6)

# Add names to the new results matrix
colnames(trypt_res$coefficients2) <- c(colnames(trypt_res$coefficients), 'p(bonf)')
rownames(trypt_res$coefficients2) <- c(rownames(trypt_res$coefficients))

# Print model coefficients
trypt_res$coefficients2

##                                     Estimate Std. Error      df t value Pr(>|t|) 
## (Intercept)                 44.57333  3.87088 79.82486 11.51503 0.00000
## Groupcontrol                  9.66784  5.31081 79.82486  1.82041 0.07244
## Groupdepleted                -0.32788  5.95114 79.82486 -0.05510 0.95620
## time6.55                     14.28667  3.86452 80.00000  3.69688 0.00040
## time12                        17.58000  3.86452 80.00000  4.54908 0.00002
## Groupcontrol:time6.55       -17.23961  5.30207 80.00000 -3.25148 0.00168
## Groupdepleted:time6.55      -10.95939  5.94136 80.00000 -1.84459 0.06880
## Groupcontrol:time12          -14.12706  5.30207 80.00000 -2.66444 0.00932
## Groupdepleted:time12          -7.47091  5.94136 80.00000 -1.25744 0.21225
##                                     p(bonf)
## (Intercept)                 0.00000
## Groupcontrol                  0.65200
## Groupdepleted                  8.60580
## time6.55                      0.00359
## time12                         0.00017
## Groupcontrol:time6.55        0.01513
## Groupdepleted:time6.55        0.61918
## Groupcontrol:time12           0.08392
## Groupdepleted:time12           1.91027

# get comprehensible pairwise interactions (uncorrected for multiple comparisons)
lsm = emmeans(trypt_model, ~ Group*time)
contrast(lsm, interaction = "pairwise")

##   Group_pairwise    time_pairwise estimate    SE df t.ratio p.value
## loaded - control     7.05 - 6.55    -17.240 5.30 80 -3.251  0.0017

```

```

## loaded - depleted 7.05 - 6.55    -10.959 5.94 80 -1.845 0.0688
## control - depleted 7.05 - 6.55    6.280 5.79 80  1.084 0.2815
## loaded - control 7.05 - 12      -14.127 5.30 80 -2.664 0.0093
## loaded - depleted 7.05 - 12      -7.471 5.94 80 -1.257 0.2123
## control - depleted 7.05 - 12      6.656 5.79 80  1.149 0.2539
## loaded - control 6.55 - 12       3.113 5.30 80  0.587 0.5588
## loaded - depleted 6.55 - 12       3.488 5.94 80  0.587 0.5588
## control - depleted 6.55 - 12      0.376 5.79 80  0.065 0.9484

```

#5b

Report and discuss the findings. What do they mean? How do they relate to the hypotheses?

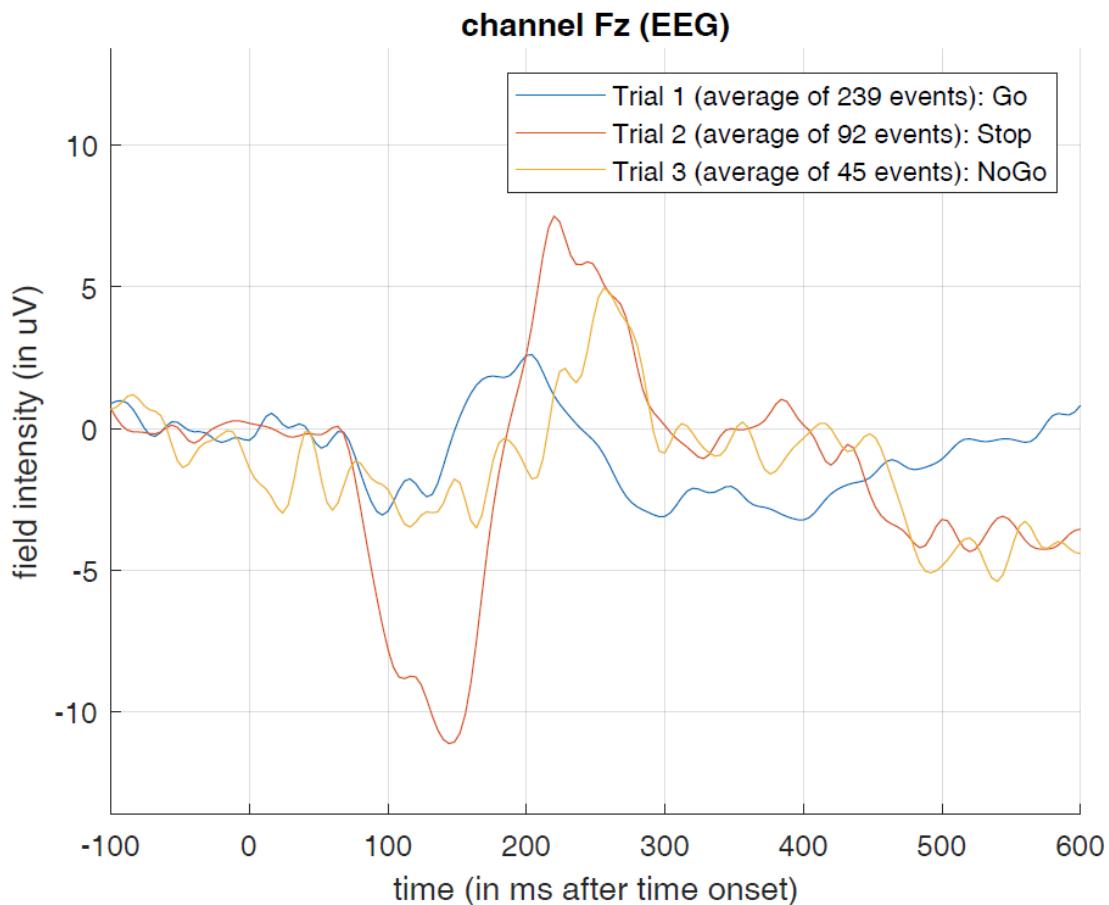
H1: Being depleted of tryptophan is hypothesized to lead to alterations of mood. The results from the analysis show that there is not a significant difference between the baseline (Loaded) and the tryptophan-depleted group's mood, $\beta = -0.33$, $t(79.82486) = -0.05$, $p - \text{value} > 0.9$, bonferroni $p - \text{value} > .6$.

H2: forcing yourself to eat a nasty powder at 7:00 will alter the mood. This is tested by looking at the interaction effects between Groupcontrol:time6.55, where we compare the mood of the Loaded and control group at 6:55 and 7:05. These results suggest a significant result, $\beta = -17.23$, $t(80) = -3.25$, $p - \text{value} < .002$, bonferroni $p - \text{value} < .02$.

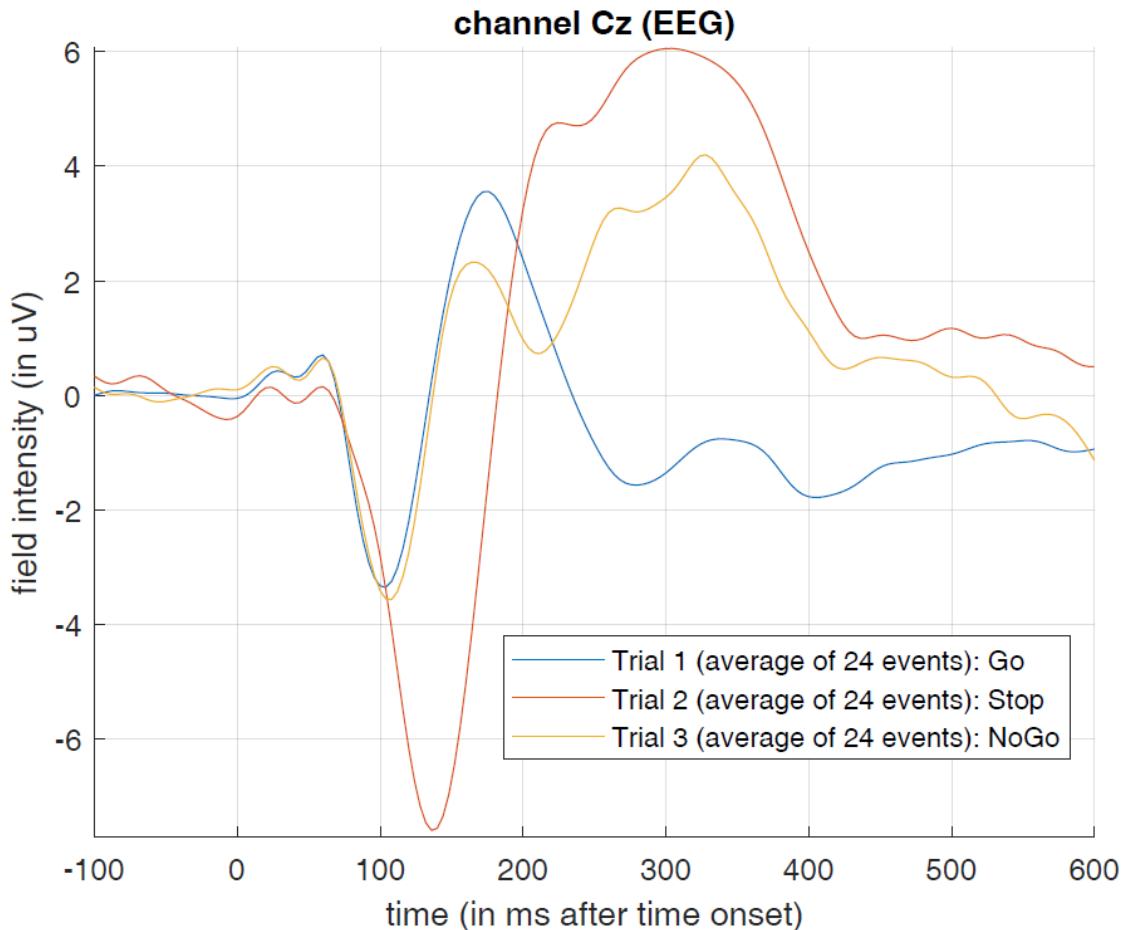
H3: becoming hungry at 12:00 alters mood. This is tested by examining people's mood at 12 pm. The results show a significant decrease in mood, $\beta = 17.58$, $t(80) = 4.55$, $p - \text{value} < .001$, bonferroni $p - \text{value} < .001$.

Assignment 8

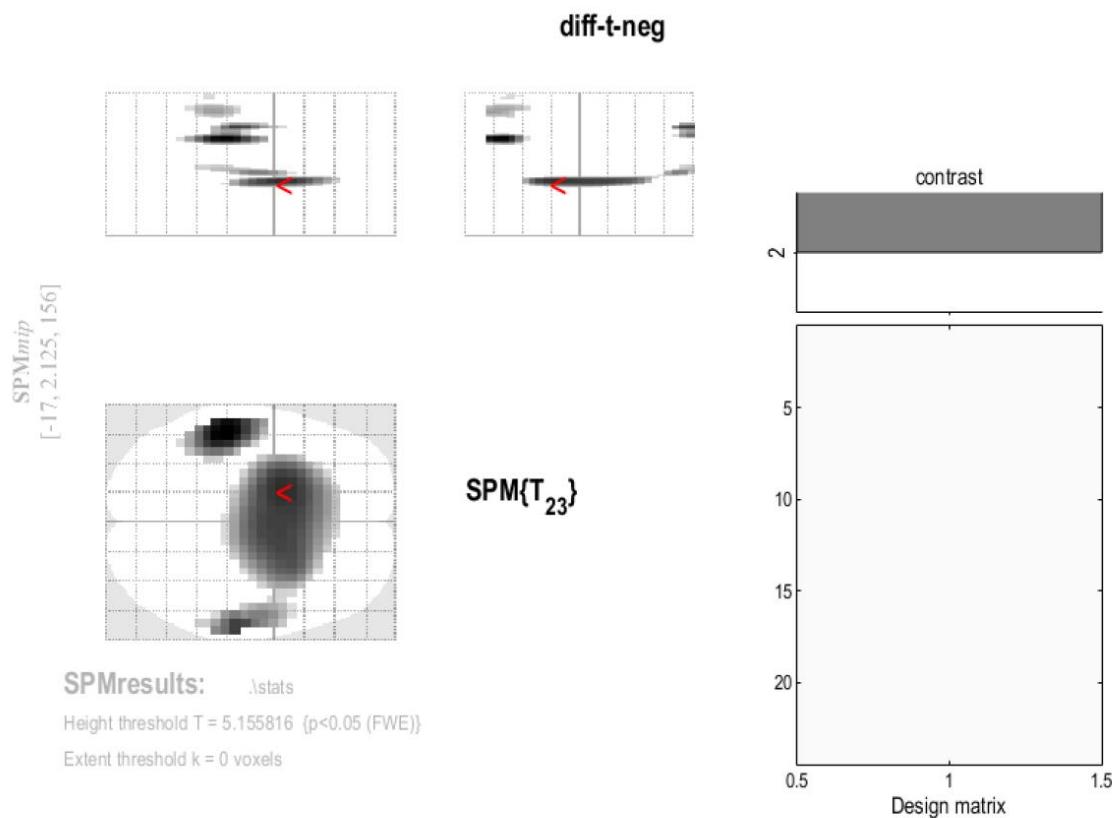
Group 9 - Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad, Daryna Eismont, and Bella Terragni



Above plot shows one participant's brain response at Fz to the three different conditions (Go, Stop and NoGo). In this ERP plot, we can see a very negative effect of Stop at N150, followed by a positive peak at P225. These two peaks might be indicators of the mismatch negativity effect at N200 and attention effect at P300, although this is not clear-cut.



Above plot shows the averaged ERP's at each of the three conditions. Comparing this plot with the earlier plot from a single participant, we see that the effects have gotten more prominent as noise has been reduced. There is an effect of the Go condition at N100 (N1) and P175 (P2), which might be the N1-P2 complex. There is a likewise effect of NoGo, although NoGo also elicits a P300 effect, which is probably due to the direction of attention in order to inhibit response. There is an effect of Stop at N140 as well as a dispersed effect around P200-P350. This might be two positive peaks interacting, one of them being P300.



Statistics: p-values adjusted for search volume

set-level	c	cluster-level				peak-level				mm mm ms			
		p _{FWE-corr}	q _{FDR-corr}	k _E	p _{uncorr}	p _{FWE-corr}	q _{FDR-corr}	T	(Z _≡)	p _{uncorr}	mm	mm	ms
0.000	5	0.000	0.012	382	0.005	0.000	0.014	8.53	5.67	0.000	-51	-30	368
		0.000	0.000	1870	0.000	0.000	0.027	7.78	5.40	0.000	-17	2	156
		0.002	0.053	175	0.042	0.001	0.030	7.50	5.28	0.000	64	-25	428
		0.001	0.030	249	0.018	0.008	0.188	6.12	4.67	0.000	-55	-36	496
		0.034	0.667	8	0.667	0.044	0.875	5.22	4.20	0.000	-43	-41	580

There seems to be a problem with our design matrix, since it does not show any gray-scale graded cells.

Our group would appreciate a little guidance in order to produce a functional design matrix and make more sense of our results, if possible. Many thanks 😊

Assignment 9

Thea Hølmkjær Kaad, Peter Mikkelsen Thramkrongart og Bella Terragni
Experimental Methods 2

1.a Report order of conditions

The order of the conditions is:

{'blue_neutral'} {'blue_fear'} {'yellow_neutral'} {'yellow_fear'} {'incorrect'}

Not all files contains a condition for incorrect answers. This is probably due to these people never giving the wrong answer, and thus not needing the column to tally up these scores.

1.b Comprehension question

The hypotheses of the experiment are as follows:

H1: An activation is expected in the Ocipital Face Area (OFA) and Fusiform Face Area (FFA) across all stimuli.

H2: Fearful faces will yield a greater response than neutral images. This is hypothesised to originate in emotional regions, e.g. the amygdalae, but may also lead to a modulation of visual areas, e.g. OFA and FFA.

H3: The index finger (blue) trials will lead to a smaller BOLD response in the motor cortex than middle finger (yellow) trials, due to the hand being more adapted to using the index finger.

H4: Infrequent stimuli will yield stronger BOLD response than frequent in brain areas relevant for perception of faces and motor responses.

The contrasts we use to analyze the FMRI scans:

All pos: The all positive contrast is [1 1 1 1] This contrast tests for H1, namely, what brain areas are activated.

blue_vs_yellow: [1 1 -1 -1] H3 Does blue produce a stronger BOLD signal than yellow faces?

fearful > neutral: [-1 1 -1 1] H2 Do fearful expression produce stronger response than neutral?

interaction [-1 1 1 -1] H4 This contrast examines an interaction effect, that is, whether the blue fearful and yellow neutral conditions make some areas more active, while blue neutral and yellow fearful make these areas less active.

yellow_vs_blue: [-1 -1 1 1] H3 Does yellow produce a stronger BOLD signal than blue faces?

interaction2: [1 -1 -1 1] H4 This contrast examines an interaction effect, that is, whether the blue neutral and yellow fearful make some areas more active, while blue fearful and yellow neutral make these areas less active.

fearful<neutral [1 -1 1 -1] H2 test whether neutral faces results in more activation than fearful faces.

1.d Checking preprocessing

The scan images from one participant, participant 9, seem not to have covered the entirety of his or her brain, leaving part of the prefrontal cortex black. It is generally quite problematic to keep such a participant's data in the statistical analysis because of two reasons. One, spm will completely disregard the part of the brain in all participants which is missing and thus not show statistical significant activation, even if it is there. Second, it is in the interest of the researcher to have as big of a data set as possible. If you delete data, the statistical results lose power and might not be significant. On the other hand, it is in general questionable to remove data, merely because it does not suit you as a researcher. It could be viewed as p-hacking or fraudulence, if you remove data in order to weigh the odds in the favor of your own experiment. In this case, however, we are examining the structural images before analysis, which enables us to be unbiased in our judgement. Furthermore, the brain regions we expect to analyze are not located in the prefrontal cortex. Due to these deliberations, we chose to keep the participant's data in our analyses, but it is a case by case judgement.

2.a Investigate the design matrix

Looking at the design matrix it is obvious to see that she has received more blue stimuli than yellow, since there are more white than black in the blue columns. She has two incorrect responses.

It is normally a good idea to model incorrect responses separately. This is due to the fact that it can be hard to predict, why an error has occurred. Furthermore, when a participant makes mistakes, it could be due to many different factors and these factors could contort our data. Therefore, it is also pertinent to model the incorrect answers in order to gain a better insight into how these answers might affect the data.

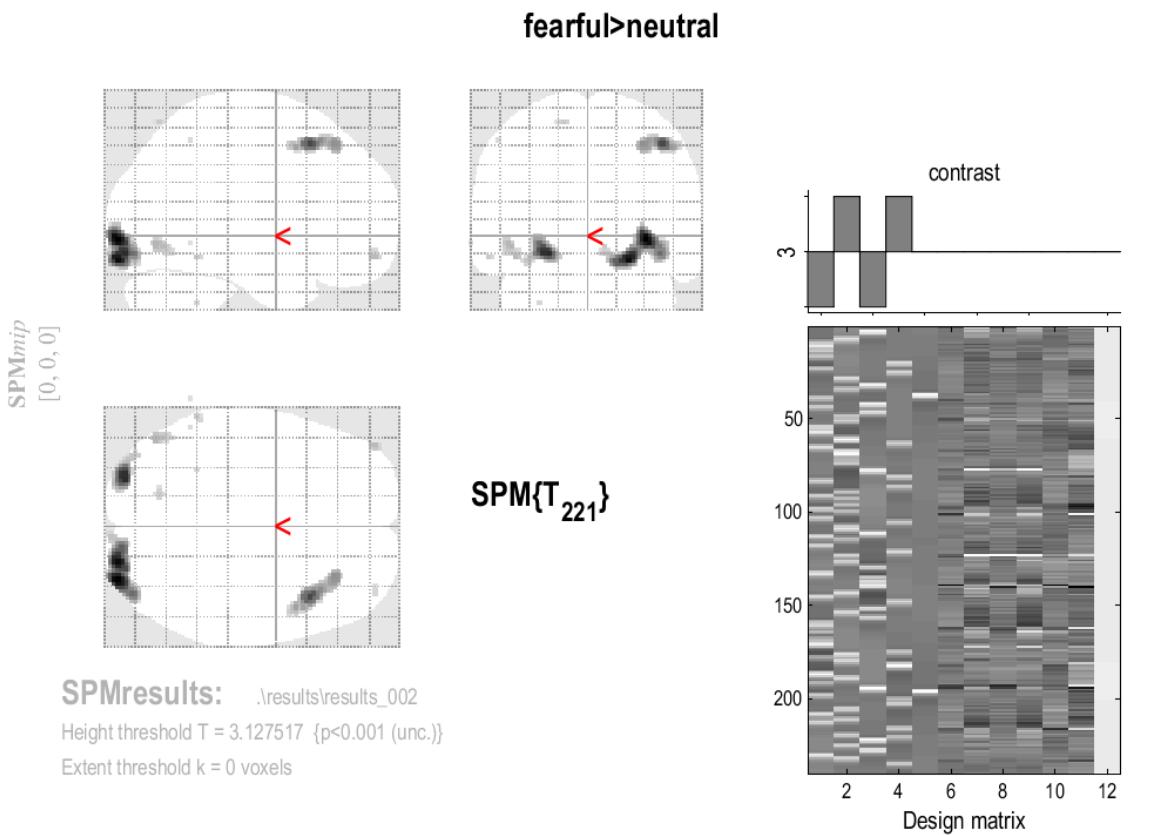
2.b Investigate the results

In the following, the statistical results and overlays for participant 2 will be printed together with a couple of descriptive sentences.

We see that there is evidence for hypothesis H2, since fearful faces elicit bigger response than neutral. H1 is also fulfilled, since the face-related areas (OFA and FFA in particular) are activated across conditions

When it comes to H3 and the effect of which finger, elicits the greatest activation, blue/index finger creates a greater BOLD response than yellow in seven different clusters (cluster size > 9 voxels); but yellow elicits a greater response than blue in five different areas (cluster size > 9 voxels). These results were obtained using a non-corrected p-value at 0.001 significance. Utilizing familywise error correction, we find no significant activation. These results somewhat contradict the H2 hypothesis, that the blue condition would elicit a smaller response, since the index finger is generally more used than the middle finger. According to literature, the brain will activate less the more experienced it becomes with handling a task, since the movements or steps become automatized and procedural. In the blue over yellow condition, there is a small cluster along the central sulcus, which might be due to an activation spike in the motor cortex because of the use of a finger.

When looking for an interaction effect, and therefore evidence for the hypothesis H4, we see significant activation for the interaction effect 2 in several places of the brain. There are six clusters with a voxel size 10 or above, which are significant.



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	mm	mm	mm
0.944	11	0.000	0.000	396	0.000	0.025	0.033	5.10	4.95	0.000	32	-94	-4
						0.047	0.033	4.94	4.80	0.000	22	-94	-16
						0.499	0.190	4.22	4.14	0.000	14	-92	-16
		0.030	0.007	180	0.002	0.134	0.066	4.66	4.54	0.000	-24	-90	-12
						1.000	0.865	3.27	3.22	0.001	-38	-82	-16
		0.026	0.007	187	0.002	0.283	0.114	4.43	4.33	0.000	42	18	50
						0.932	0.411	3.80	3.74	0.000	30	32	50
						0.983	0.508	3.67	3.61	0.000	48	10	48
		0.822	0.283	39	0.103	0.999	0.730	3.49	3.44	0.000	-46	-72	-8
		0.998	0.821	11	0.376	0.999	0.730	3.45	3.40	0.000	-44	56	-14
		1.000	0.821	6	0.521	1.000	0.841	3.31	3.27	0.001	-16	-70	62
		1.000	0.821	3	0.663	1.000	0.841	3.30	3.26	0.001	-60	-48	-14
		1.000	0.821	3	0.663	1.000	0.906	3.22	3.18	0.001	40	56	-14
		1.000	0.821	1	0.821	1.000	0.906	3.21	3.17	0.001	-30	-48	-40
		1.000	0.821	1	0.821	1.000	0.924	3.18	3.14	0.001	-54	-52	52
		1.000	0.821	1	0.821	1.000	0.993	3.13	3.09	0.001	-70	-52	-6

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.13, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 15.128$

Expected number of clusters, $\langle c \rangle = 16.73$

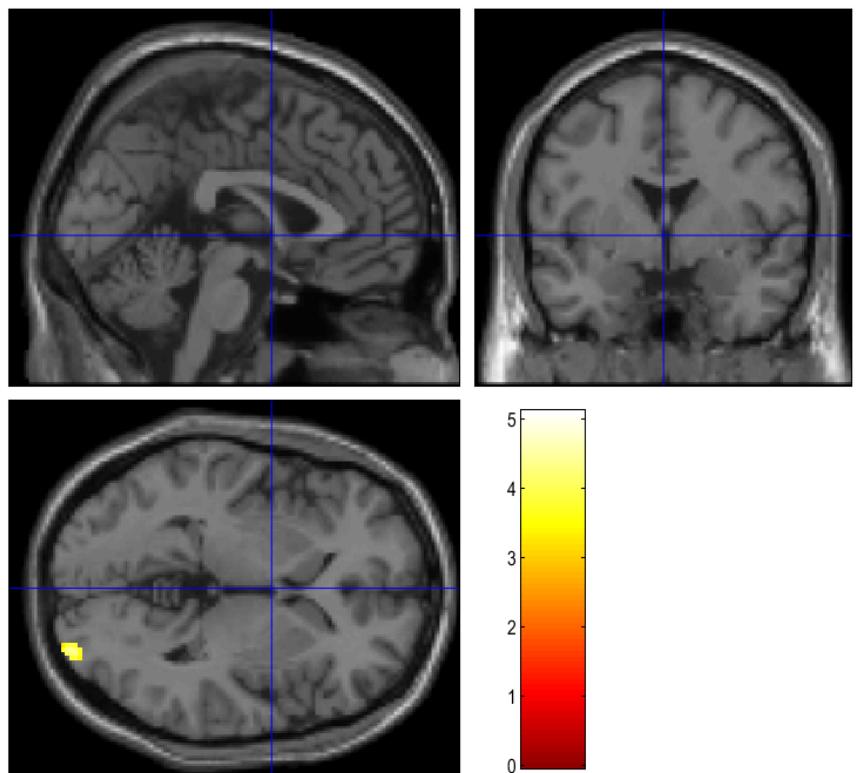
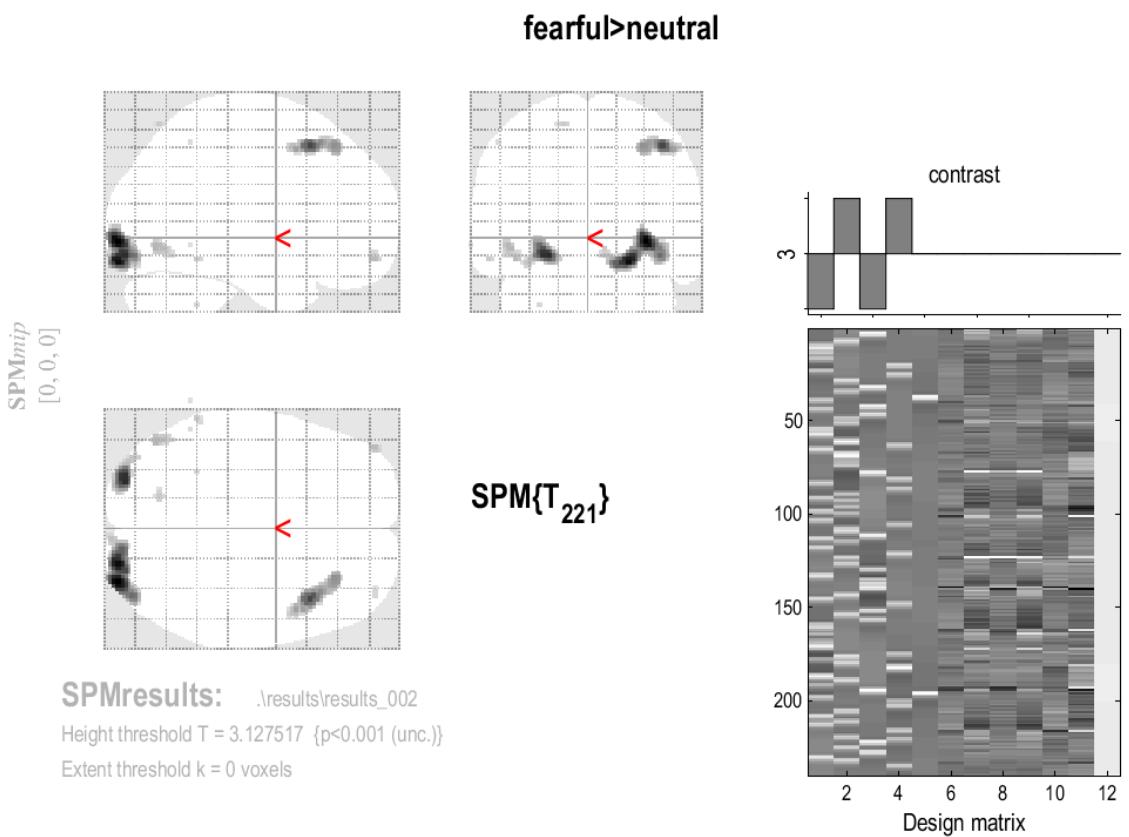
FWEp: 4.922, FDRp: 4.939, FWEc: 180, FDRc: 180

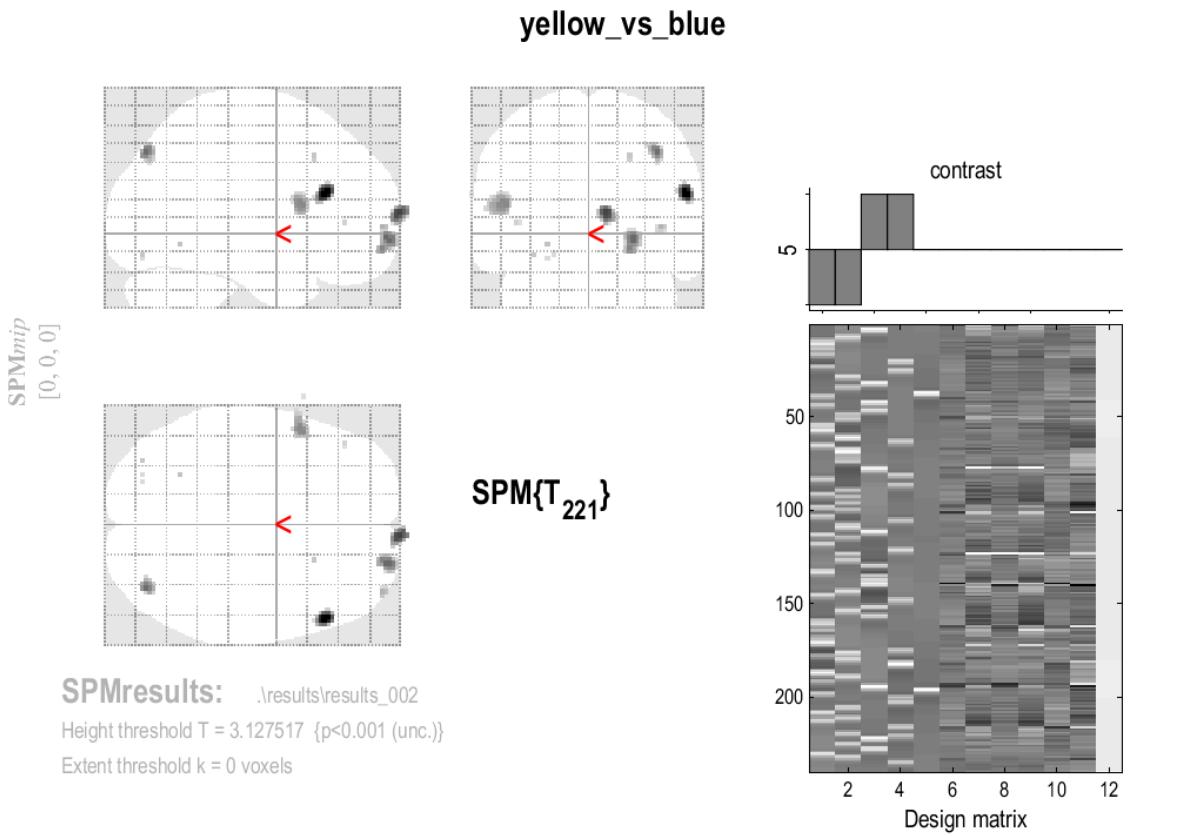
Degrees of freedom = [1.0, 221.0]

FWHM = 10.4 10.4 9.8 mm mm mm; 5.2 5.2 4.9 {voxels}

Volume: 1781184 = 222648 voxels = 1560.6 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 131.88 voxels)





Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	54	28	22
0.851	13	0.531	0.180	62	0.045	0.302	0.186	4.41	4.31	0.000	54	28	22
		0.604	0.180	56	0.055	0.778	0.480	3.99	3.92	0.000	8	70	10
		0.331	0.156	82	0.024	0.939	0.564	3.79	3.73	0.000	24	62	-4
		0.856	0.301	36	0.116	0.948	0.564	3.77	3.71	0.000	38	-78	44
		0.271	0.156	90	0.019	0.990	0.564	3.62	3.56	0.000	-50	14	14
		0.999	0.821	9	0.425	1.000	0.922	3.39	3.34	0.000	40	60	2
		1.000	0.821	1	0.821	1.000	0.951	3.29	3.24	0.001	-26	-58	-8
		1.000	0.821	1	0.821	1.000	0.951	3.28	3.24	0.001	-34	-80	-14
		1.000	0.821	2	0.731	1.000	0.951	3.27	3.23	0.001	22	20	42
		1.000	0.821	2	0.731	1.000	0.987	3.21	3.17	0.001	-44	38	4
		1.000	0.821	1	0.821	1.000	0.987	3.17	3.13	0.001	-22	-80	-14
		1.000	0.821	1	0.821	1.000	0.987	3.16	3.12	0.001	-26	-80	-16
		1.000	0.821	1	0.821	1.000	0.990	3.13	3.09	0.001	-70	14	20

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.13, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 15.128$

Expected number of clusters, $\langle c \rangle = 16.73$

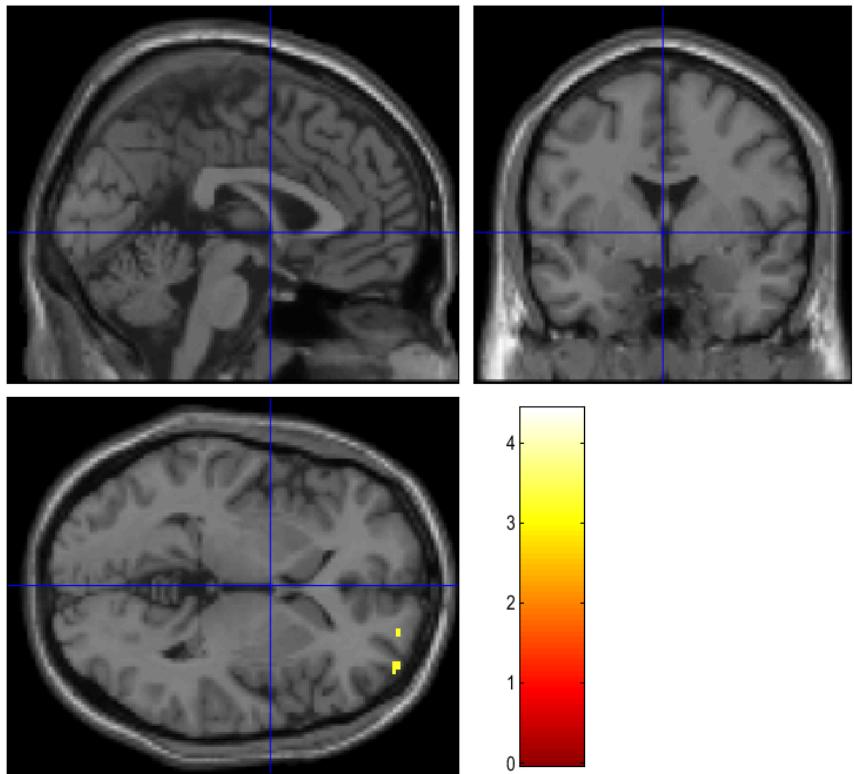
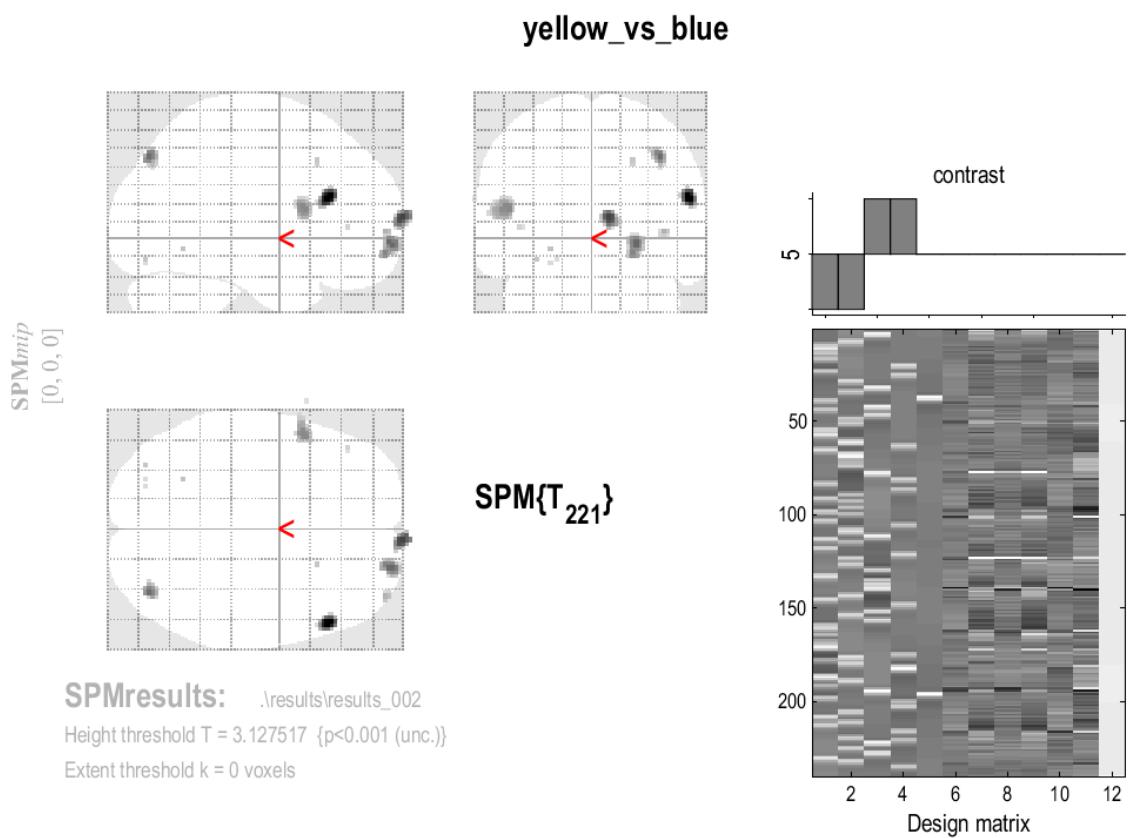
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 221.0]

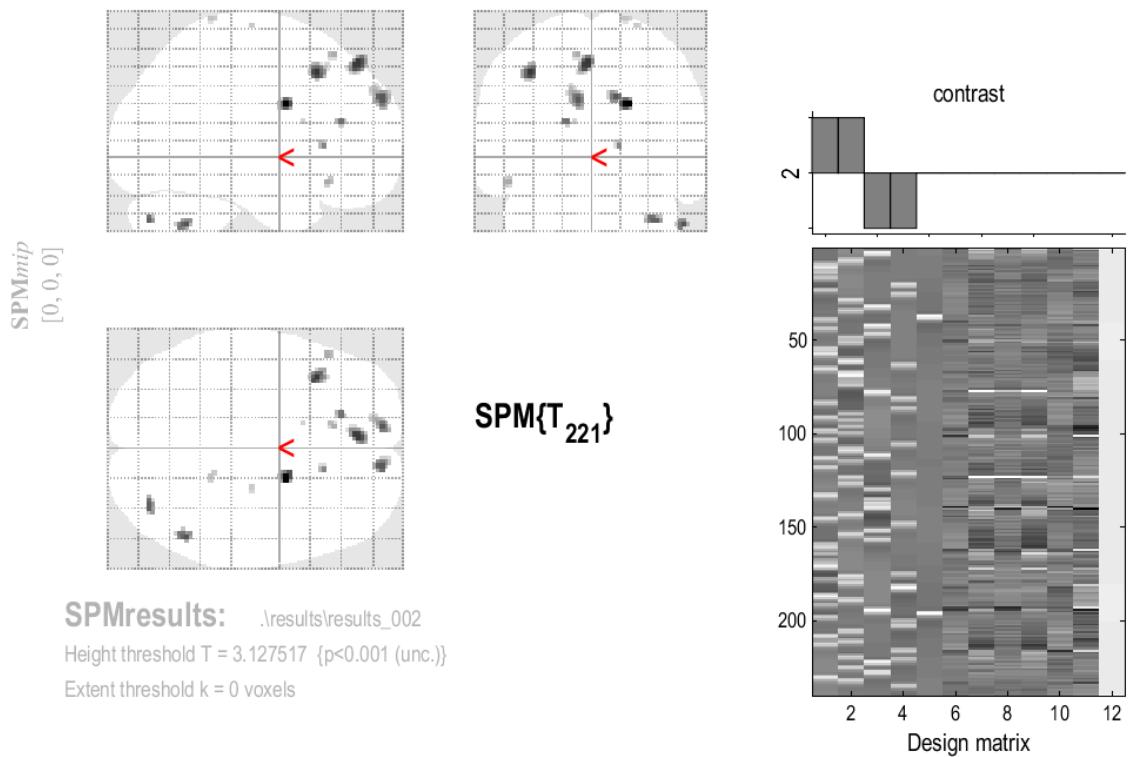
FWHM = 10.4 10.4 9.8 mm mm mm; 5.2 5.2 4.9 {voxels}

Volume: 1781184 = 222648 voxels = 1560.6 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 131.88 voxels)



blue_vs_yellow



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	20	2	28
0.781	14	0.948	0.541	26	0.176	0.754	0.515	4.01	3.94	0.000	20	2	28
		0.669	0.331	51	0.066	0.945	0.515	3.78	3.72	0.000	-4	44	52
		0.695	0.331	49	0.071	0.948	0.515	3.77	3.71	0.000	-38	20	46
		0.980	0.544	20	0.233	0.982	0.515	3.67	3.61	0.000	52	-56	-40
		0.960	0.541	24	0.193	0.985	0.515	3.66	3.60	0.000	12	58	32
		0.656	0.331	52	0.064	0.987	0.515	3.64	3.58	0.000	-10	58	30
		0.995	0.634	14	0.317	0.990	0.515	3.62	3.57	0.000	36	-78	-38
		1.000	0.663	6	0.521	0.996	0.560	3.55	3.50	0.000	-16	34	18
		1.000	0.663	7	0.485	0.999	0.607	3.48	3.43	0.000	14	24	4
		0.999	0.663	8	0.454	1.000	0.780	3.33	3.29	0.001	-50	26	-16
		1.000	0.663	6	0.521	1.000	0.780	3.32	3.28	0.001	-10	30	56
		1.000	0.709	4	0.608	1.000	0.871	3.24	3.20	0.001	18	-42	76
		1.000	0.787	2	0.731	1.000	0.871	3.21	3.17	0.001	24	-18	72
		1.000	0.821	1	0.821	1.000	0.883	3.18	3.14	0.001	-12	12	18

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.13, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 15.128$

Expected number of clusters, $\langle c \rangle = 16.73$

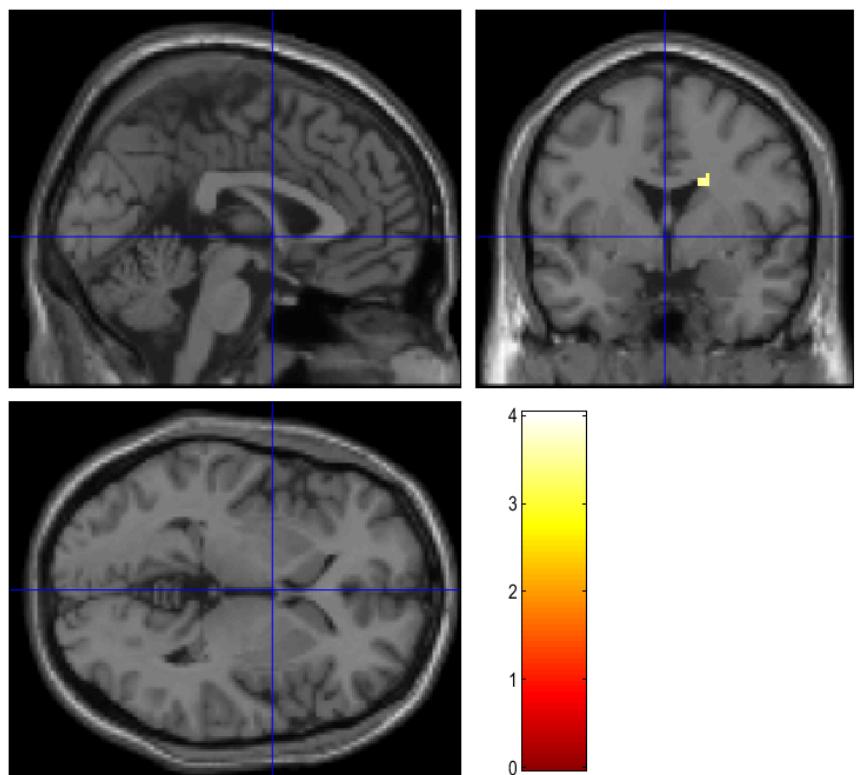
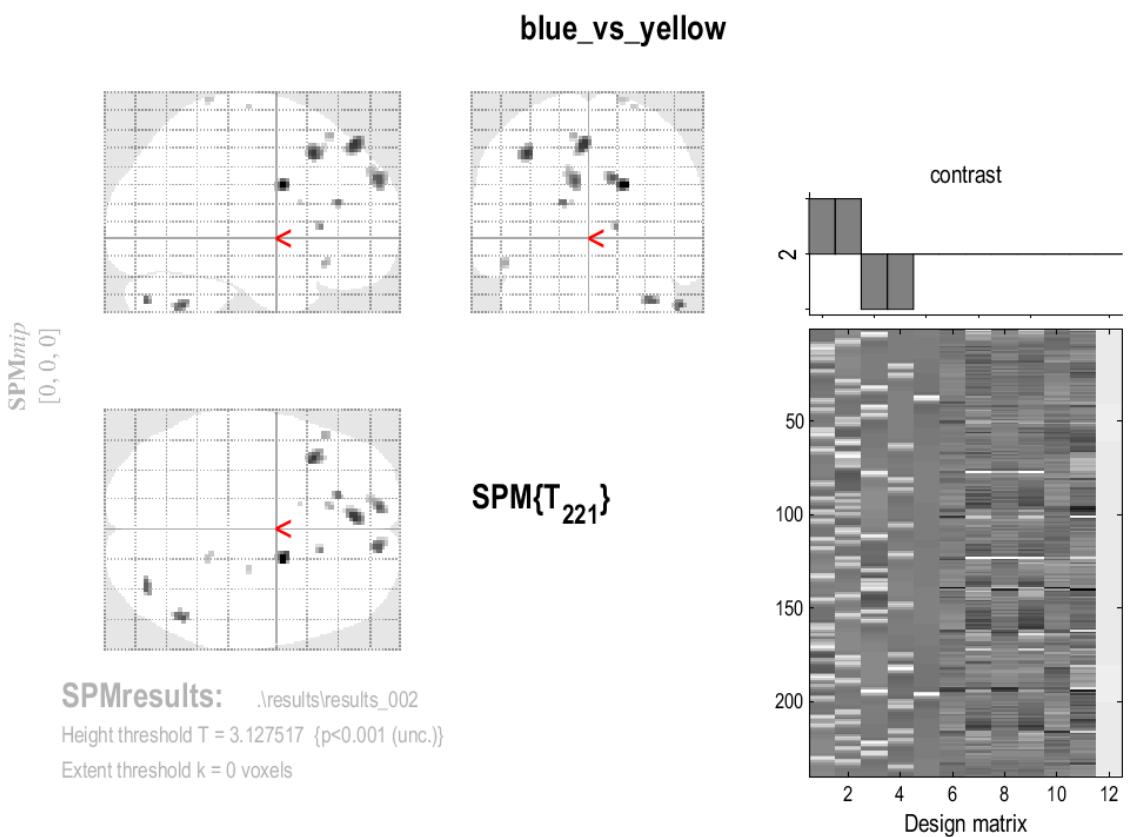
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 221.0]

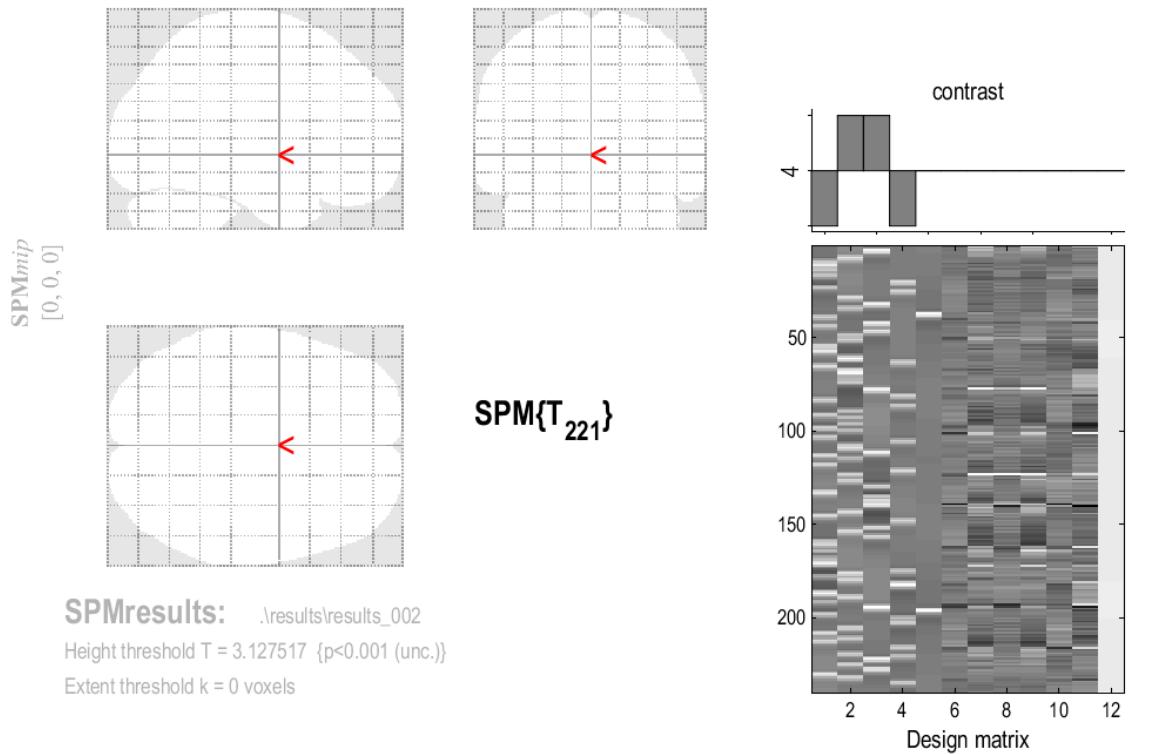
FWHM = 10.4 10.4 9.8 mm mm mm; 5.2 5.2 4.9 {voxels}

Volume: 1781184 = 222648 voxels = 1560.6 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 131.88 voxels)



interaction



Statistics: p-values adjusted for search volume

set-level		cluster-level			peak-level				mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	

no suprathreshold clusters

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.13, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 15.128$

Expected number of clusters, $\langle c \rangle = 16.73$

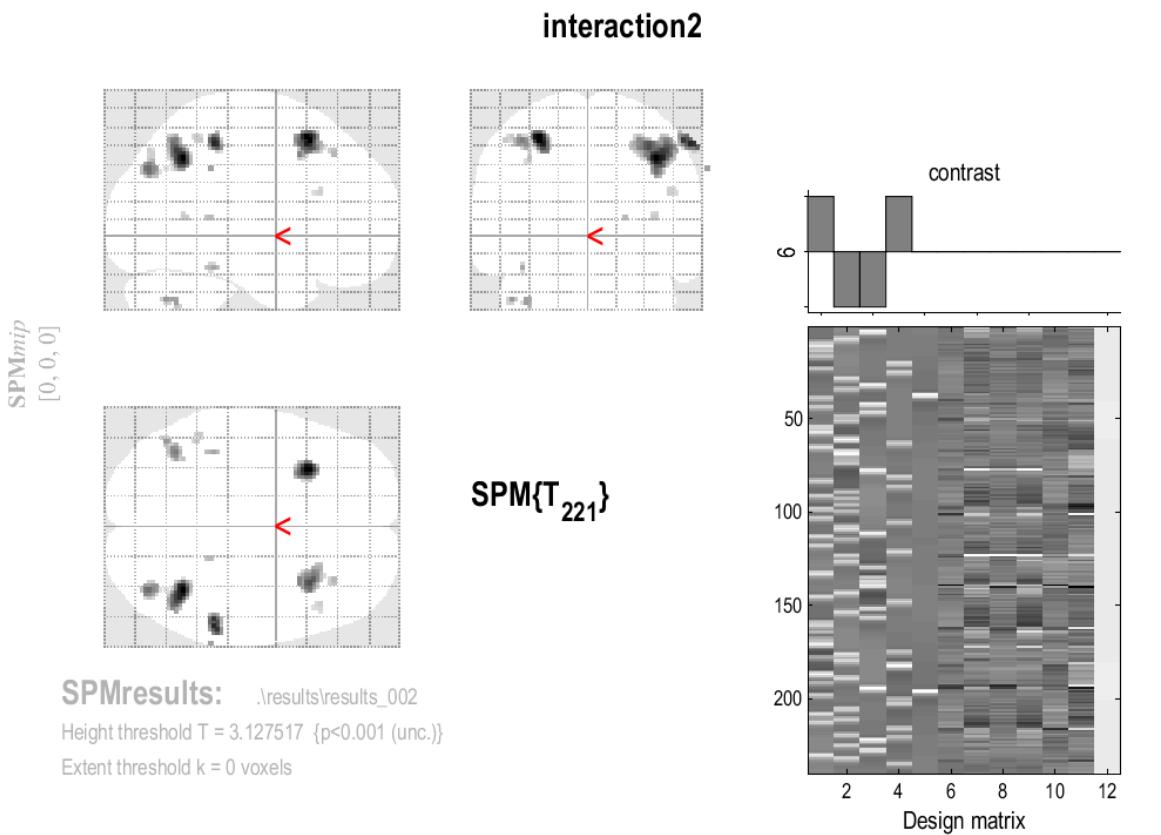
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 221.0]

FWHM = 10.4 10.4 9.8 mm mm mm; 5.2 5.2 4.9 {voxels}

Volume: 1781184 = 222648 voxels = 1560.6 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 131.88 voxels)



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}			
0.506	17	0.014	0.014	216	0.001	0.433	0.432	4.28	4.19	0.000	38	-56	42
		0.100	0.036	130	0.006	0.453	0.432	4.26	4.18	0.000	-30	18	52
		0.567	0.209	59	0.050	0.721	0.458	4.04	3.97	0.000	58	-38	50
		0.102	0.036	129	0.006	0.906	0.566	3.85	3.78	0.000	34	18	48
						1.000	0.855	3.27	3.23	0.001	28	16	54
		0.643	0.209	53	0.061	0.974	0.747	3.70	3.64	0.000	38	-76	36
						1.000	0.855	3.21	3.17	0.001	34	-76	46
		0.785	0.260	42	0.092	0.991	0.816	3.62	3.56	0.000	-40	-60	52
		1.000	0.734	7	0.485	0.999	0.825	3.48	3.43	0.000	-46	-66	-38
		0.999	0.734	9	0.425	0.999	0.825	3.47	3.42	0.000	-38	-60	-40
		1.000	0.821	1	0.821	0.999	0.825	3.45	3.40	0.000	68	-40	36
		1.000	0.734	7	0.485	1.000	0.825	3.42	3.37	0.000	-40	-38	-20
		1.000	0.734	6	0.521	1.000	0.855	3.34	3.30	0.000	32	32	44
		1.000	0.821	2	0.731	1.000	0.855	3.31	3.27	0.001	20	-40	8
		1.000	0.738	4	0.608	1.000	0.855	3.27	3.23	0.001	-36	-56	42
		0.999	0.734	8	0.454	1.000	0.855	3.24	3.20	0.001	-46	-48	54
		1.000	0.734	5	0.561	1.000	0.855	3.24	3.20	0.001	36	-56	8
		0.999	0.734	8	0.454	1.000	0.855	3.22	3.18	0.001	46	24	22
		1.000	0.821	1	0.821	1.000	0.990	3.13	3.09	0.001	38	-66	36

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.13, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 15.128$

Expected number of clusters, $\langle c \rangle = 16.73$

FWEp: 4.922, FDRp: Inf, FWEc: 216, FDRc: 129

Degrees of freedom = [1.0, 221.0]

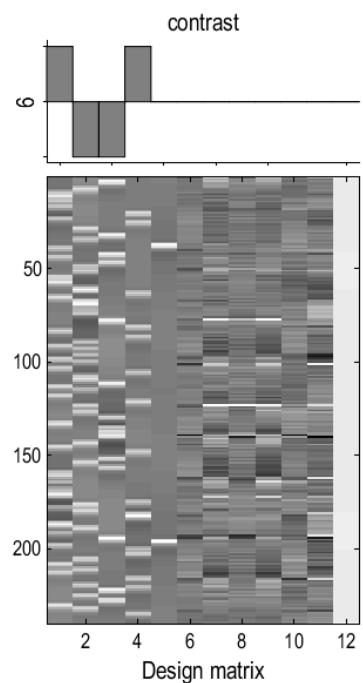
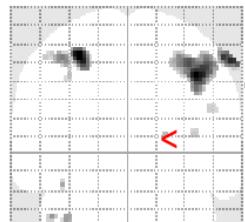
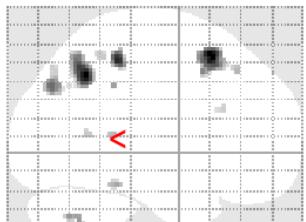
FWHM = 10.4 10.4 9.8 mm mm mm; 5.2 5.2 4.9 {voxels}

Volume: 1781184 = 222648 voxels = 1560.6 resels

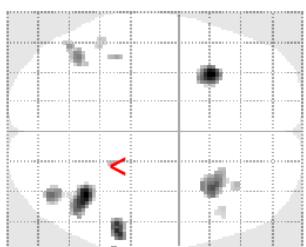
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 131.88 voxels)

interaction2

SPMminip
[20, -40, 8]



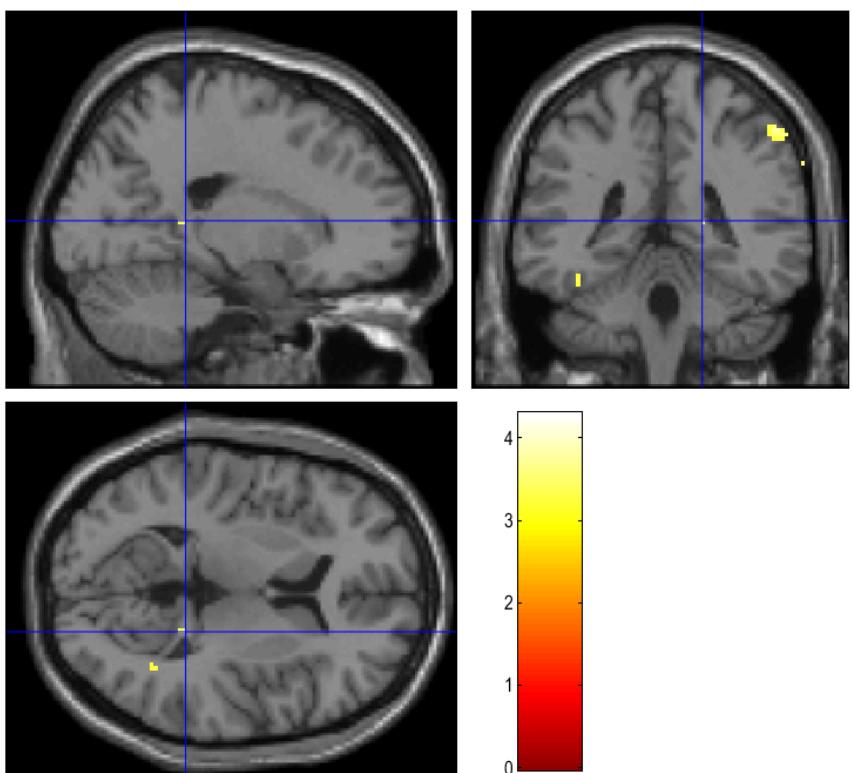
SPM{ T_{221} }



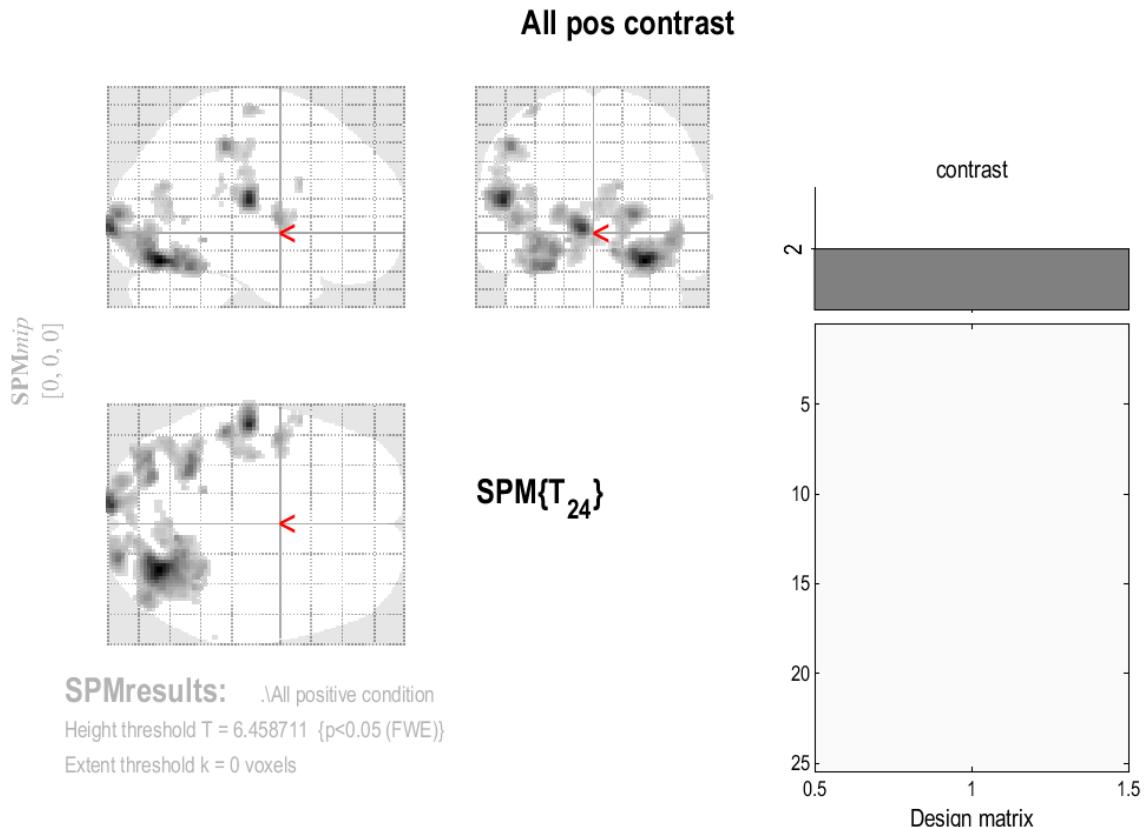
SPMresults: .\results\results_002

Height threshold T = 3.127517 {p<0.001 (unc.)}

Extent threshold k = 0 voxels



3.a all positive condition



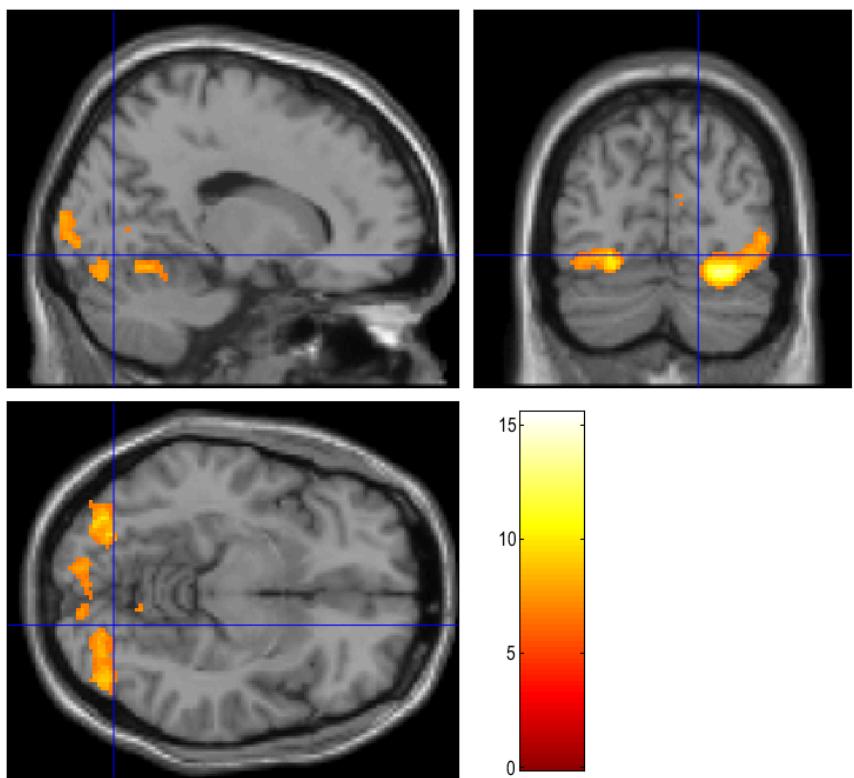
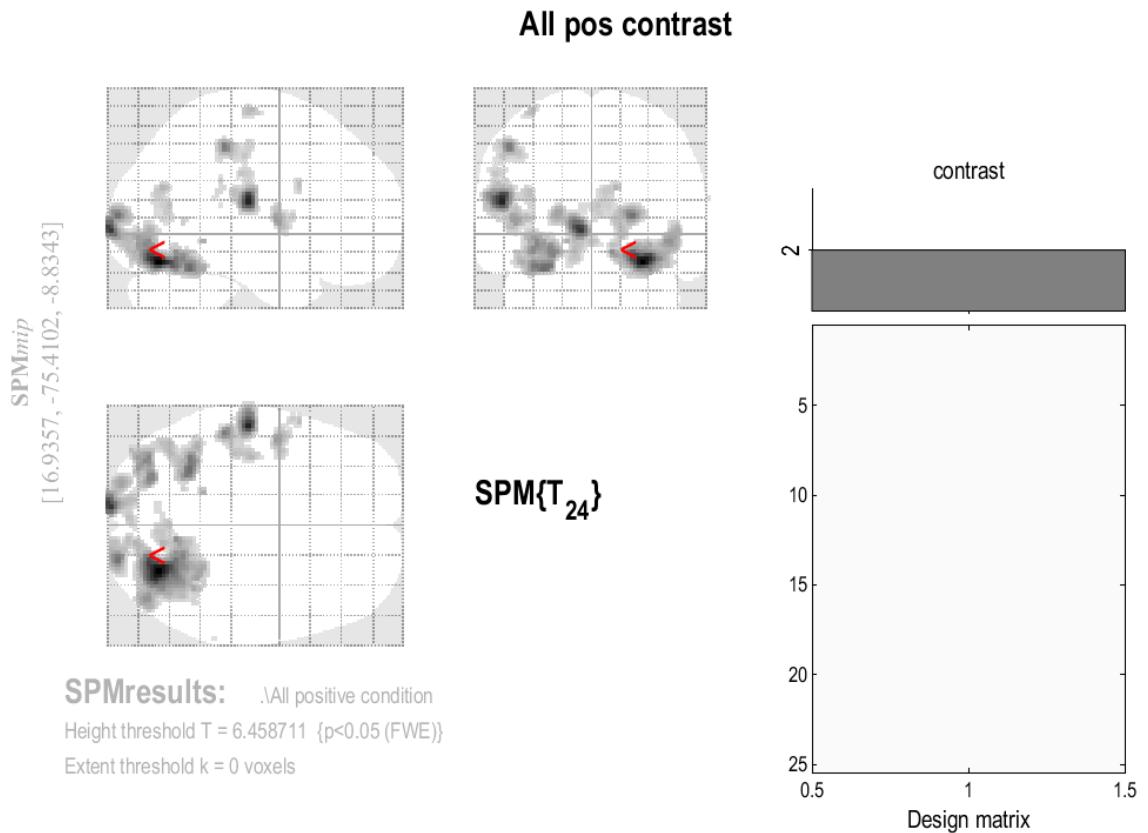
Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	p _{FWE-corr}	q _{FDR-corr}	k _E	p _{uncorr}	p _{FWE-corr}	q _{FDR-corr}	T	(Z _≥)	p _{uncorr}			
0.023		0.451	0.451	1	0.451	0.041	0.834	6.56	4.92	0.000	-52	-36	14
	0.023	0.451	0.451	1	0.451	0.042	0.836	6.55	4.91	0.000	66	-40	18

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 6.46, p = 0.000 (0.050)
Extent threshold: k = 0 voxels
Expected voxels per cluster, <k> = 1.871
Expected number of clusters, <c> = 0.05
FWEp: 6.459, FDRp: 8.621, FWEc: 1, FDRc: 11

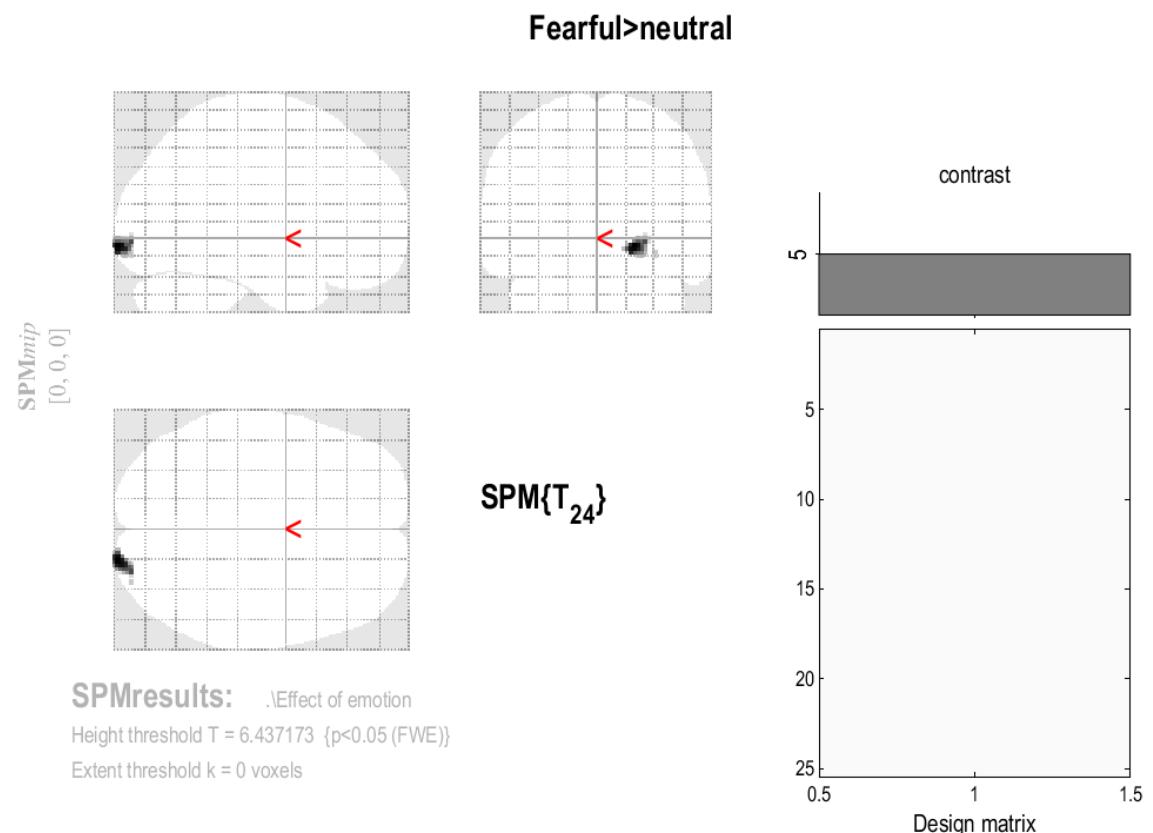
Degrees of freedom = [1.0, 24.0]
FWHM = 10.5 10.5 10.3 mm mm mm; 5.3 5.2 5.1 {voxels}
Volume: 1341864 = 167733 voxels = 1074.1 resels
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 142.45 voxels)
Page 2/2



All positive

In order to find all brain areas affected by the experiment, we use an all positive contrast. This means that there are ones [1 1 1 1] in all four slots, so that any activation, no matter the condition, will become visible. Looking at the overlay, we see activation around the band between OFA and FFA. It seems as though there is activation around STS as well, which is also a face-selective region. Lastly, there is some activation in the caudal part of the occipital lobe, which might be due to the visual stimuli.

3.b effect of emotion



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	20	-100	-6
0.001	2	0.000	0.000	84	0.000	0.002	0.105	8.20	5.61	0.000	20	-100	-6
						0.006	0.171	7.56	5.36	0.000	26	-92	-4
		0.015	0.295	2	0.295	0.035	0.702	6.62	4.95	0.000	32	-92	-10

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 6.44, p = 0.000 (0.050)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 1.974$

Expected number of clusters, $\langle c \rangle = 0.05$

FWEp: 6.437, FDRp: Inf, FWEc: 2, FDRc: 84

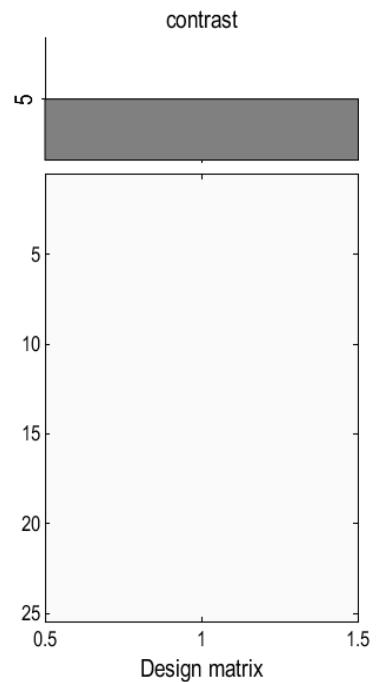
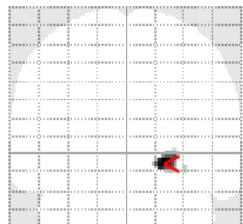
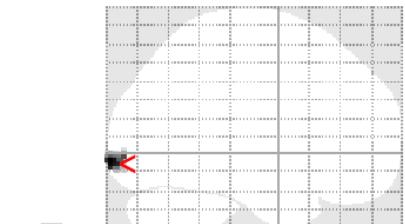
Degrees of freedom = [1.0, 24.0]

FWHM = 10.7 10.6 10.5 mm mm mm; 5.3 5.3 5.3 {voxels}

Volume: 1341864 = 167733 voxels = 1027.9 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 148.85 voxels)

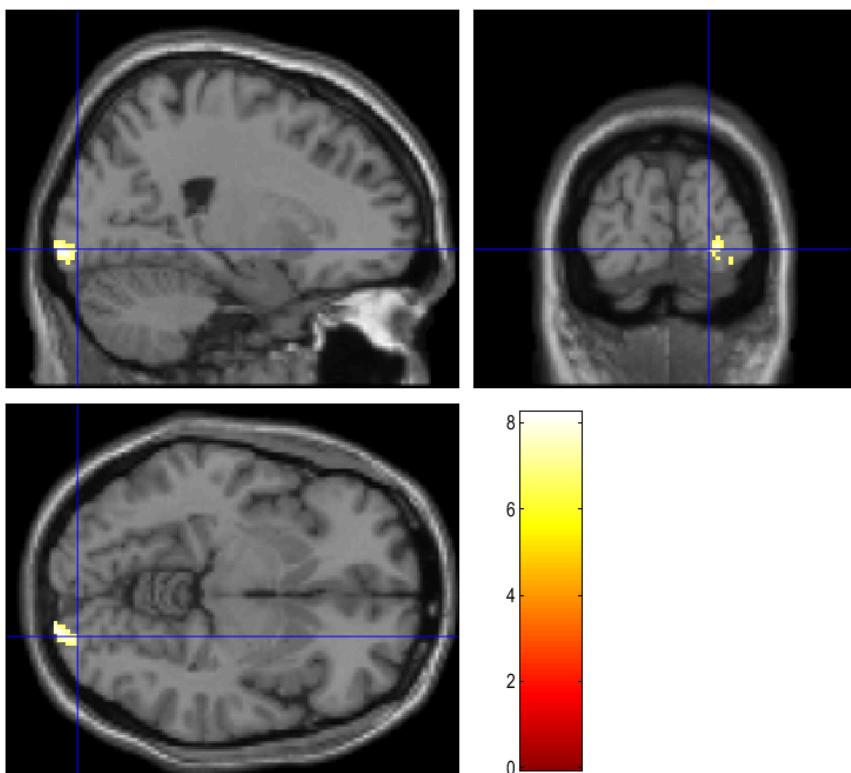
Fearful>neutral



SPMresults: .\Effect of emotion

Height threshold T = 6.437173 {p<0.05 (FWE)}

Extent threshold k = 0 voxels

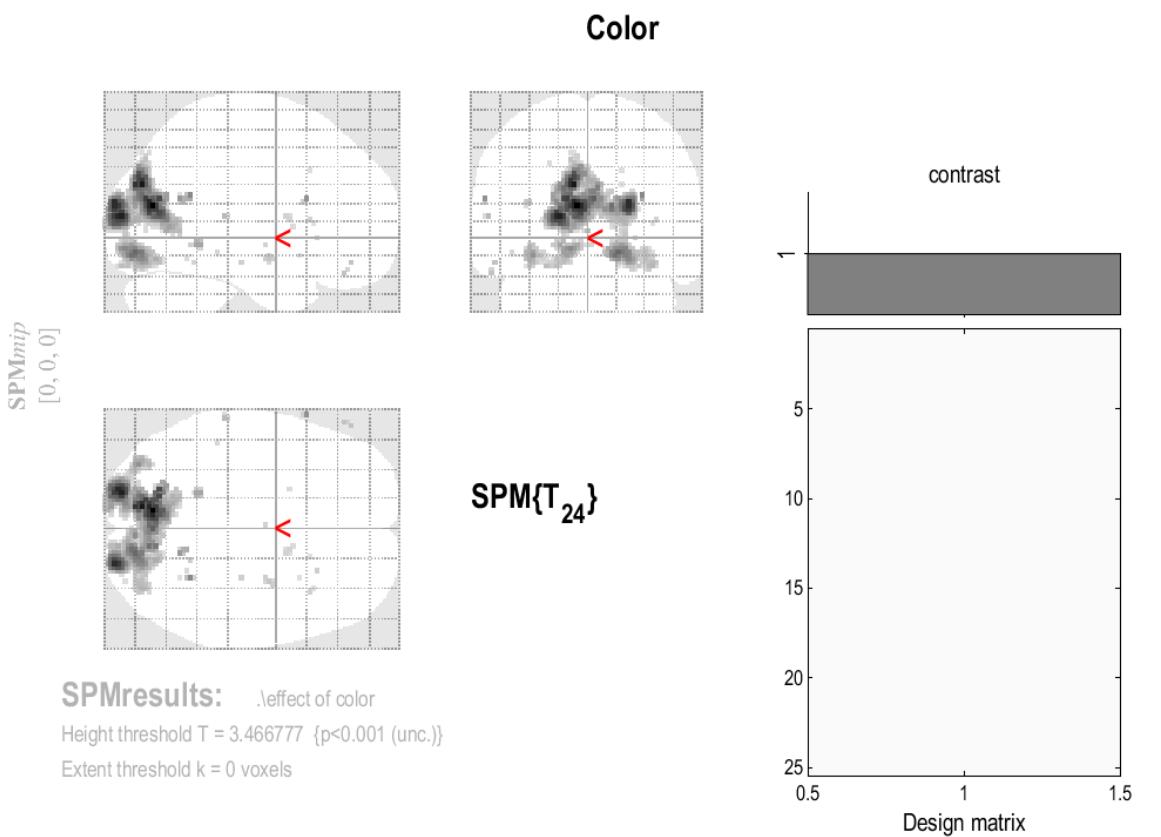


Fearful over neutral

When looking at what brain areas are more active when watching fearful faces compared to neutral, we see two statistically significant areas, which are more or less coinciding spatially. We consider whether this area might be the OFA Occipital Face Area. Looking at a standardized atlas of the brain, we see that the OFA is usually situated a little further from v1 than on our overlay of the brain, but this might be due to natural structural differences. The OFA should be more prominent in the right side of the brain, which is consistent with the right lateralized activation we see.

If we assume, the activated area is the OFA, our results show that the OFA will be more active when viewing a fearful face expression than a neutral one, p-value < .002. We only report p-values for clusters over 10 voxels, and the most significant peak for each cluster.

3.c effect of color



Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level						mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}				
0.000	34	0.000	0.000	2158	0.000	0.000	0.919	9.63	3.86	0.000	38	-18	18	
		1.000	0.844	1	0.844	1.000	0.941	3.61	3.19	0.001	-30	-50	-16	
		1.000	0.844	1	0.844	1.000	0.941	3.59	3.18	0.001	-10	-98	±6	
		0.999	0.844	3	0.802	0.808	0.883	3.59	3.18	0.000	-86	-58	-20	
		0.999	0.885	221	0.002	0.808	0.983	3.56	3.11	0.000	-20	-80	-8	
		1.000	0.844	1	0.844	1.000	0.986	3.52	3.18	0.001	-20	-86	-10	
		1.000	0.844	1	0.844	1.000	0.986	3.52	3.13	0.001	-24	-82	-40	
		1.000	0.844	2	0.864	0.908	0.988	3.82	3.58	0.000	-52	18	20	
		0.996	0.844	8	0.808	0.907	0.988	3.58	3.52	0.000	-18	-58	26	
		0.999	0.844	11	0.823	0.907	0.988	3.65	3.52	0.000	-28	-58	-16	
		0.999	0.844	3	0.802	0.905	0.988	3.86	3.32	0.000	-58	-18	16	
		1.000	0.844	2	0.864	0.906	0.988	3.80	3.38	0.000	32	-52	-12	
		1.000	0.844	2	0.864	0.909	0.988	3.49	3.32	0.000	-88	-38	-22	
		0.998	0.844	5	0.840	0.908	0.988	3.48	3.30	0.000	-36	-20	18	
		1.000	0.844	1	0.844	0.908	0.989	3.48	3.30	0.000	-58	-80	-16	
		0.996	0.844	1	0.848	0.908	0.989	3.48	3.29	0.000	32	-76	-28	
		1.000	0.844	1	0.844	0.909	0.989	3.67	3.28	0.001	20	-58	28	
		1.000	0.844	1	0.844	0.999	0.941	3.66	3.23	0.001	30	-22	-16	
		1.000	0.844	1	0.844	1.000	0.941	3.64	3.22	0.001	26	22	-2	
		0.999	0.844	4	0.653	1.000	0.941	3.62	3.20	0.001	16	4	10	
		1.000	0.844	1	0.844	1.000	0.941	3.62	3.20	0.001	-10	-74	-12	

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.47, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 19.086$

Expected number of clusters, $\langle c \rangle = 10.36$

FWEp: 6.246, FDRp: Inf, FWEc: 221, FDRc: 221

Degrees of freedom = [1.0, 24.0]

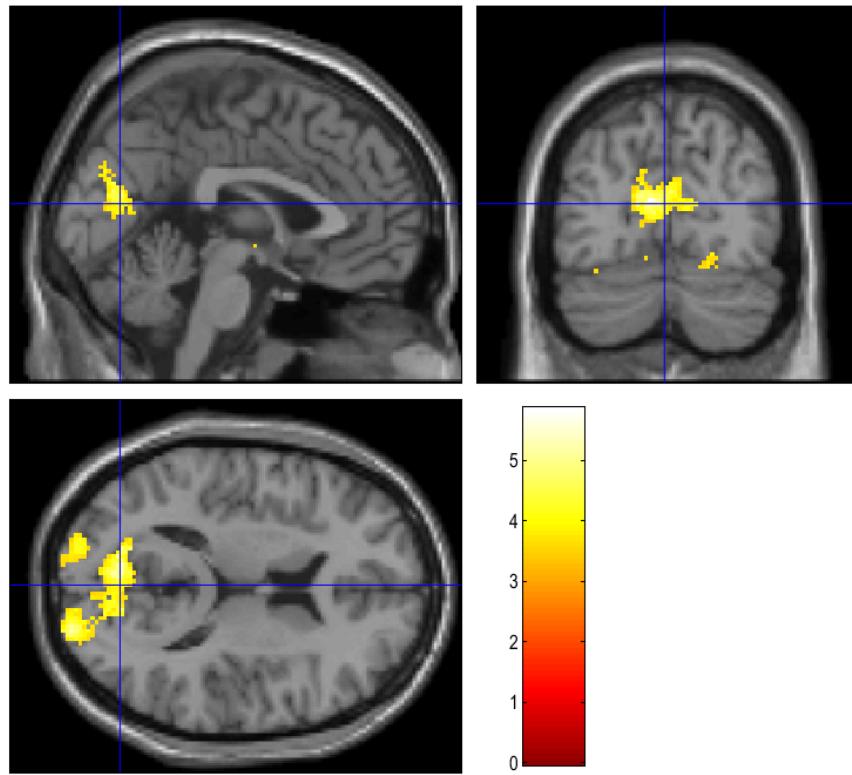
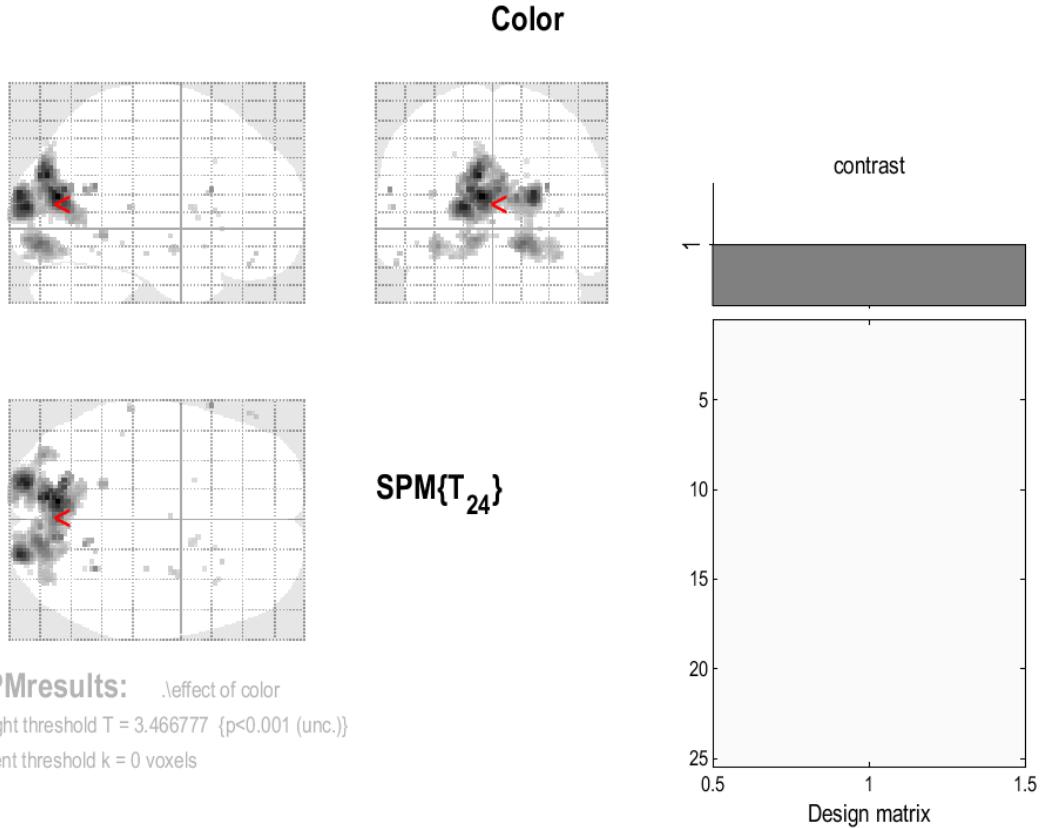
FWHM = 12.1 12.1 12.0 mm mm mm; 6.0 6.1 6.0 {voxels}

Volume: 1341864 = 167733 voxels = 694.6 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 220.29 voxels)

Page 2/2

SPMminip
[-0.526163, -73.3034, 13.7661]



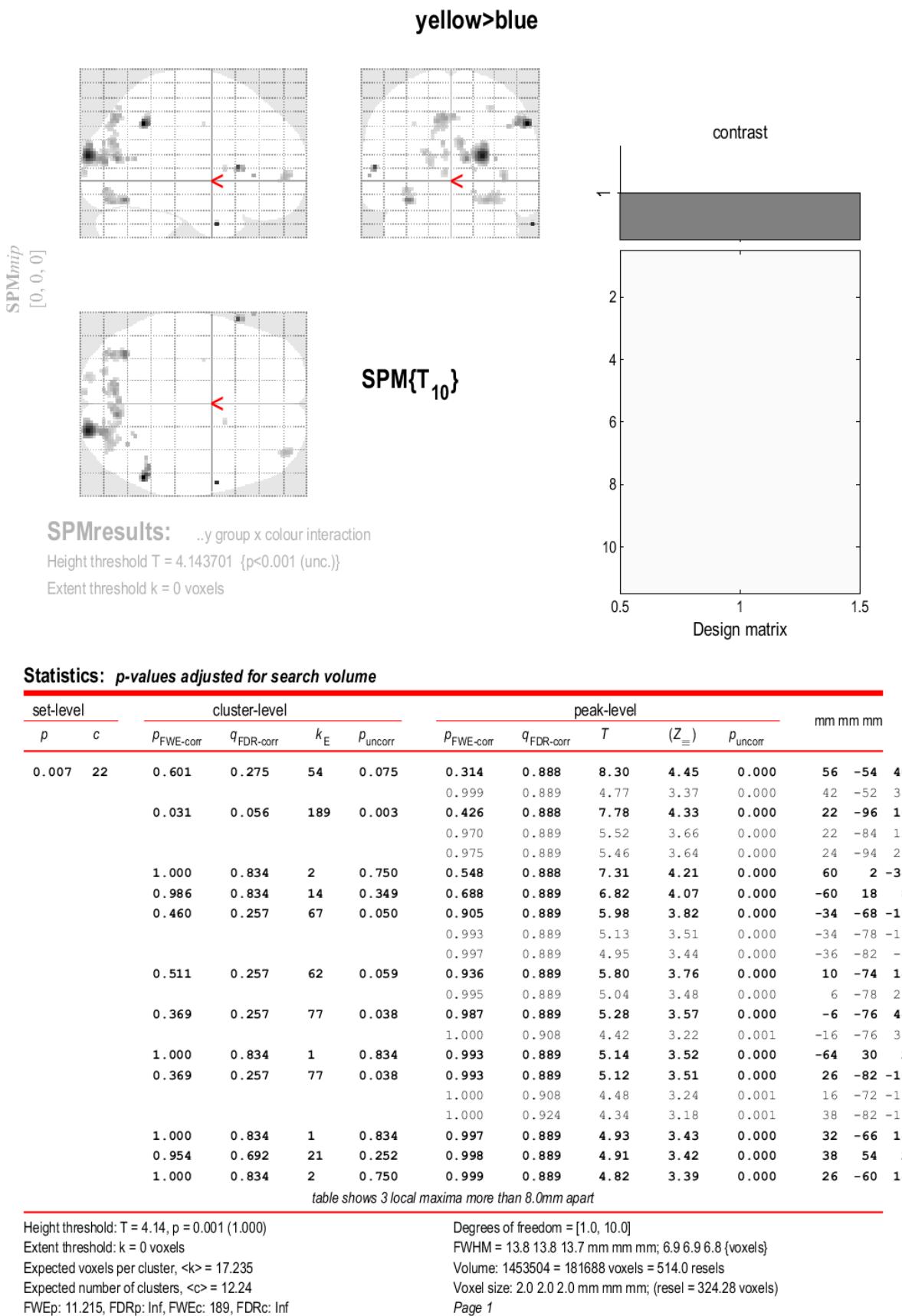
Yellow over blue

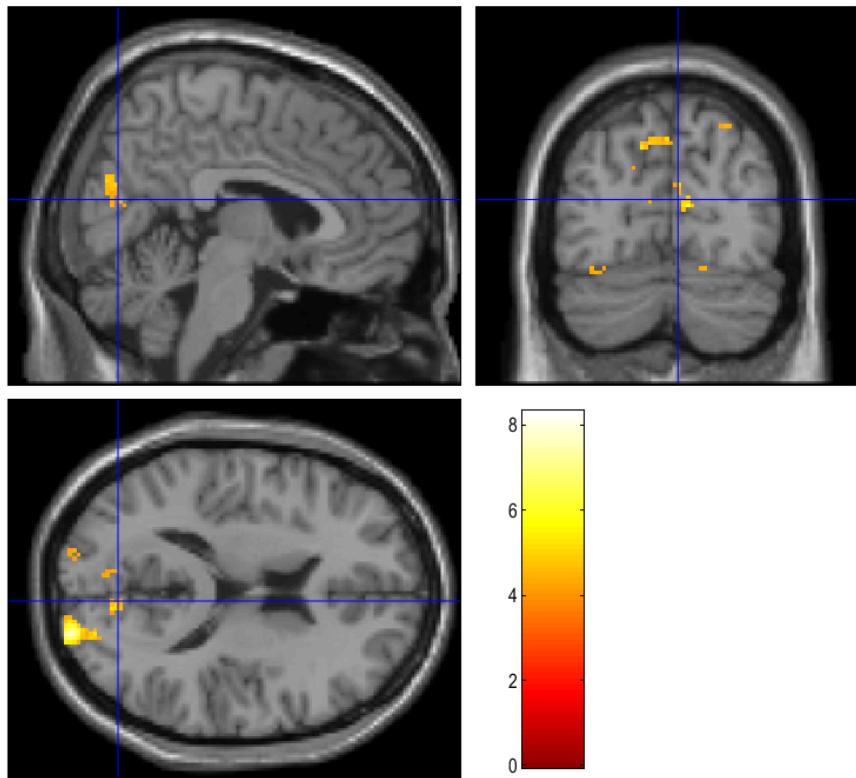
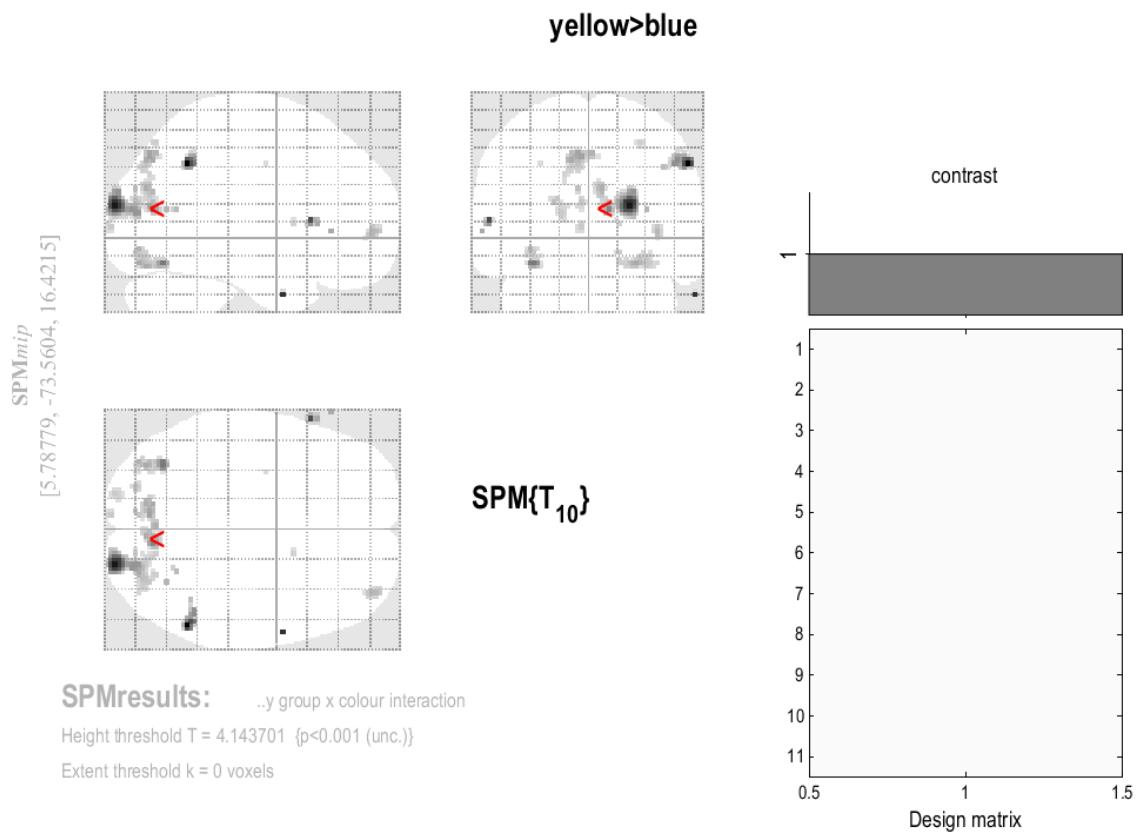
Looking at the results from the analysis, we see many significant clusters of activation around the occipital lobe. These spots are not part of our expected results. We expect to see more activation in the motor cortex area, when pressing yellow since the use of the middle finger should be less procedural than the index finger do to common use. This is not the case, however, though it is hard to say, why the yellow stimuli and/or the use of middle finger should activate the occipital lobe more than blue stimuli/index finger.

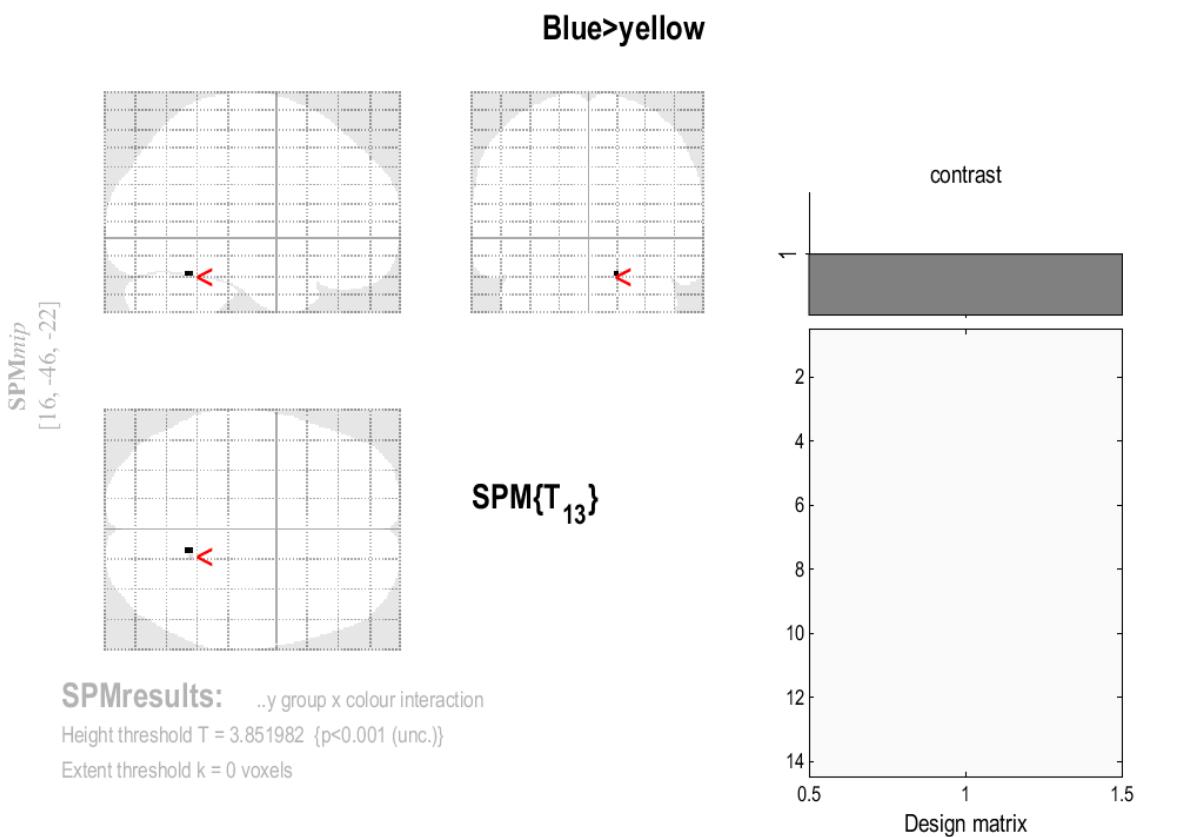
3.d frequency group x color interaction

There are clearly some interaction effects when looking at the frequently yellow group. We see that there is a higher activation for the infrequent stimuli, which goes well with the H4 hypothesis. The effects are distributed, mainly in the caudal part of the brain, but not limited to this area.

There seems to be only two significant cluster for the frequently blue group. The ventral cluster might be coinciding with the FFA, while the more dorsal cluster is harder to interpret. Both clusters are very small, and we must therefore be a bit more cautious with proclaiming too much on account of our results.







Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level						mm mm mm		
p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	k_E	p_{uncorr}	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	(Z_{\equiv})	p_{uncorr}	14	-54	-22	
1.000	2	1.000	0.766	2	0.656	0.998	0.610	4.85	3.60	0.000	14	-54	-22	
		1.000	0.766	1	0.766	1.000	0.766	4.08	3.22	0.001	18	-52	-22	

table shows 3 local maxima more than 8.0mm apart

Height threshold: T = 3.85, p = 0.001 (1.000)

Extent threshold: k = 0 voxels

Expected voxels per cluster, $\langle k \rangle = 9.689$

Expected number of clusters, $\langle c \rangle = 19.87$

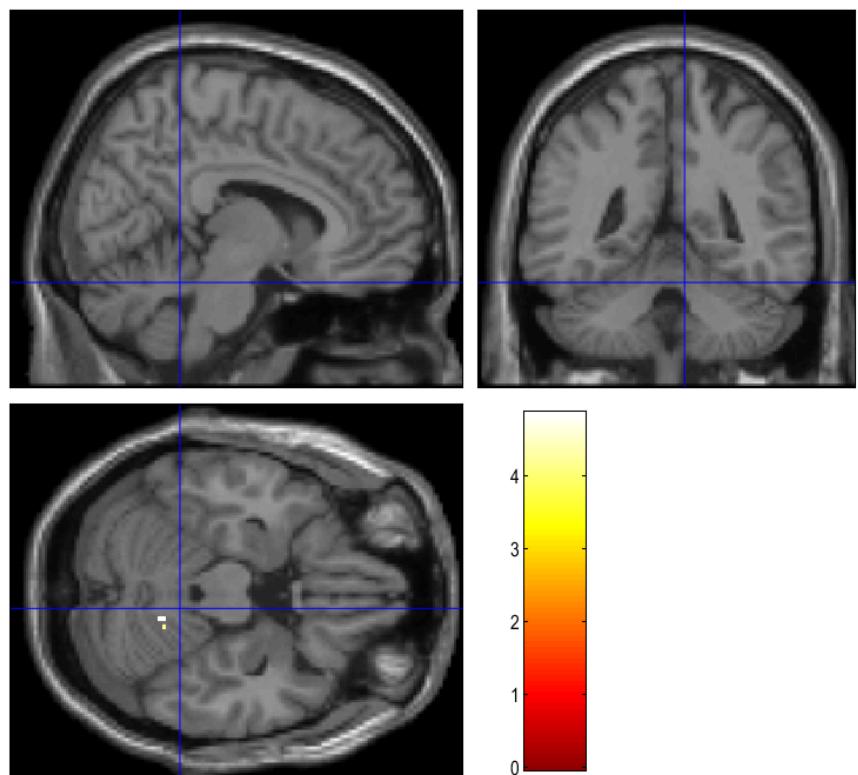
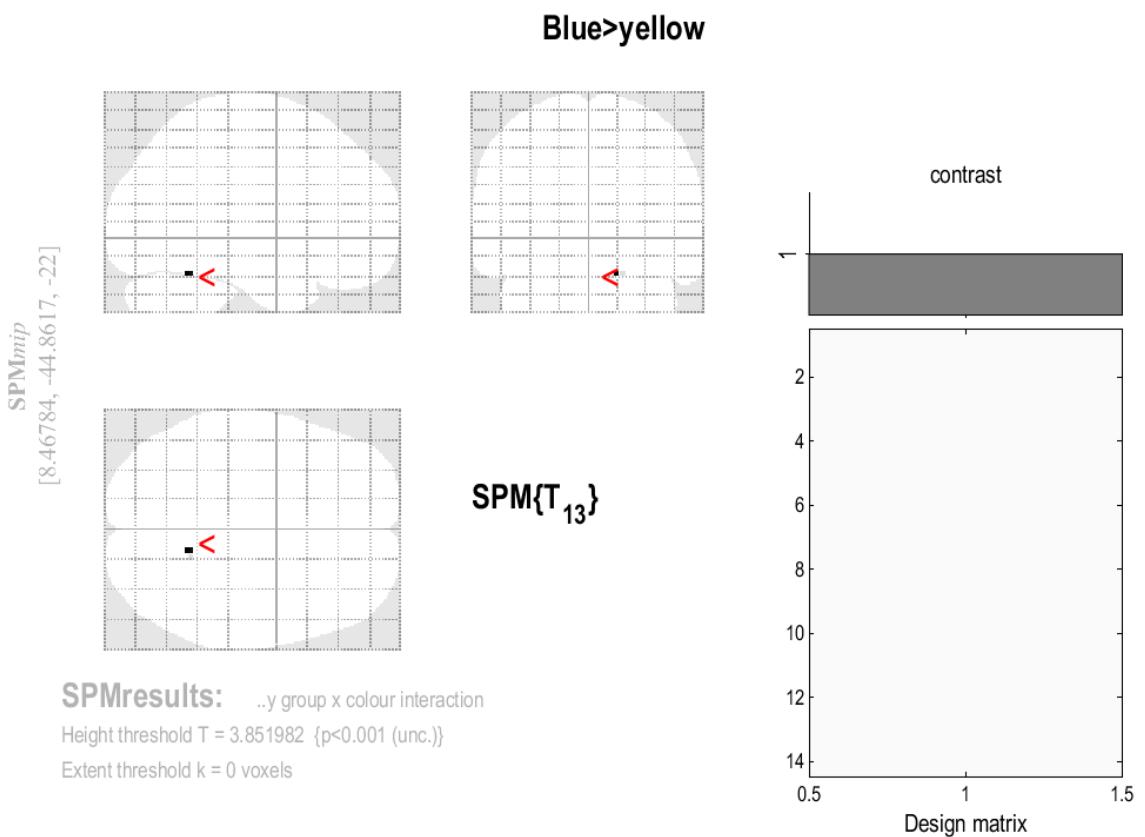
FWEp: 9.023, FDRp: Inf, FWEc: Inf, FDRc: Inf

Degrees of freedom = [1.0, 13.0]

FWHM = 10.6 10.7 10.5 mm mm mm; 5.3 5.3 5.3 {voxels}

Volume: 1367472 = 170934 voxels = 1043.8 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 149.21 voxels)



Assignment 10

Group 9 - Peter Mikkelsen Thramkrongart, Thea Hølmkjær Kaad, and Bella Terragni

17 maj 2019

0. Set up

```
#setting working directory
setwd("C:/Users/bella/Desktop/Aarhus Universitet/CogSci/Experimental Methods 2/Portfolios/10")

#install packages
library(pacman)
p_load(tidyverse, psych, corrgram, e1071)

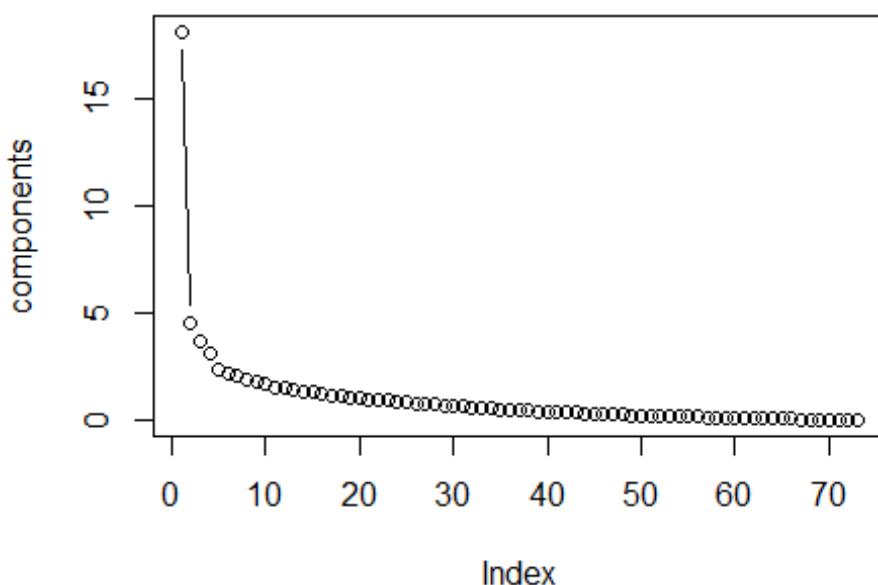
#Loading the data
data <- read.csv(file = "emp_all_all(1).csv" , sep = ",")
```

1. Determining number of relevant factors

```
#running the first PCA with all factors
PCA_1 <- principal(data, nfactors = 73)

#finding the point of inflection using a scree-plot
PCA_1_plot <- plot(PCA_1$values, type = "b", main = "Scree-plot", ylab = "components")
```

Scree-plot



```
# Above figure is a scree-plot that we use to determine the number of factors we want to include in our principal factor analysis. The point of inflection, indicates to us, at what point, adding additional factors to the analysis, would not improve the pc analysis. We judge that the point of inflection is at 4. Therefore we will include 4 factors in our analysis.
```

2. Finding the lowest and highest scoring participants for each component

```
#running the second PCA with the selected five factors
PCA_2 <- principal(data, nfactors = 4)

#assigning a variable to the participant scores from the PCA output list
scores <- as.data.frame(PCA_2$scores)

#finding max and minimum in each column
which.max(scores$RC1)

## [1] 82

# The participant with the highest score in component RC1 is nr. 82
which.min(scores$RC1)

## [1] 84

# The participant with the lowset score in component RC1 is nr. 84

which.max(scores$RC2)

## [1] 52

# The participant with the highest score in component RC2 is nr. 52
which.min(scores$RC2)

## [1] 33

# The participant with the lowest score in component RC2 is nr. 33

which.max(scores$RC3)

## [1] 101

# The participant with the highest score in component RC3 is nr. 101
which.min(scores$RC3)

## [1] 31

# The participant with the lowest score in component RC3 is nr. 31

which.max(scores$RC4)
```

```

## [1] 12
# The participant with the highest score in component RC4 is nr. 12
which.min(scores$RC4)

## [1] 82
# The participant with the lowest score in component RC4 is nr. 82

```

3. Determining whether the PEST test judges novel features compared to other tests

```

#creating a correlation matrix that examines correlated answers
cor_m <- cor(data)
corrgram(cor_m, col.regions = colorRampPalette(c("dodgerblue4", 'dodgerblue','white', 'gold',"firebrick4")),cor.method='pearson')

```



Looking at the correlation matrix, we can see that the PEST test does not seem to be overly correlated with the other questionnaires compared to overall correlation. However, the test is internally correlated and more so than the other tests. We

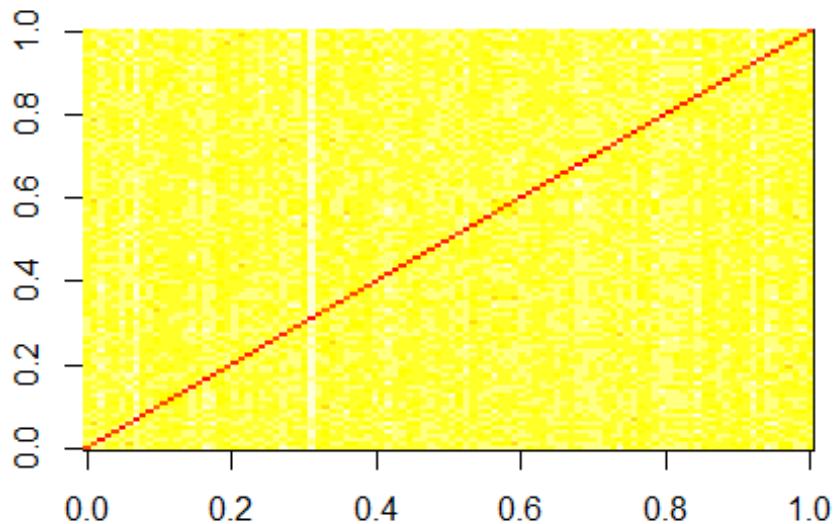
would therefore suggest that some of the PEST questions might be redundant, even though the PEST test generally seems to examine new variables of empathy.

4. Creating a couple of nice extra plots to please our employer

If our employer were interested in finding a group of similar people, since these e.g. might work together in a team, we would suggest using the Hamming Distance to group clusters of two into a Cluster Dendrogram

```
#calculating hamming distance between participants
hamming <- hamming.distance(as.matrix(data))
```

```
# first we get a quick overview using a heat map to indicate the hamming distance
# between participants
image(scale(hamming))
```



```
#making a dendrogram firstly by defining participants id
rownames(hamming) <- data[,0]
```

Looking at the heat map, there doesn't seem to be any suspiciously highly correlated participants. This indicates that there hasn't been a mistake in data collection where somebody e.g. got their answers entered twice into the tabel.

```
#clustering variables  
tree <- hclust(dist(hamming), method = "ward.D")  
# plotting results  
plot(tree)
```

Cluster Dendrogram



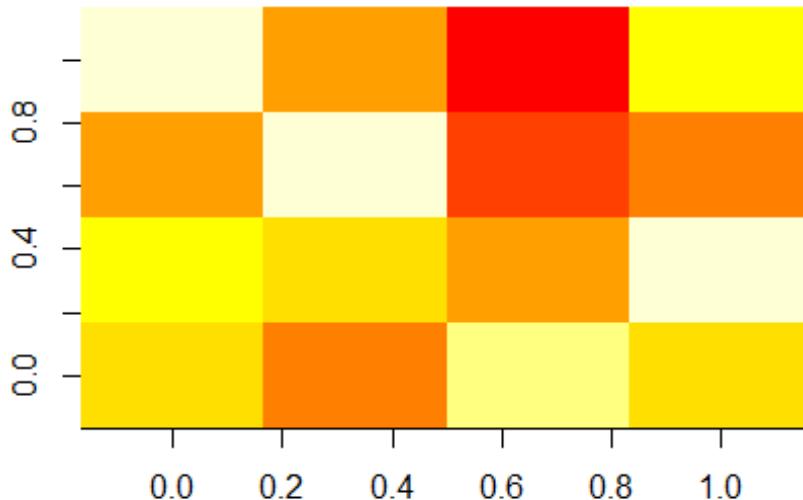
```
# We know from our principal component analysis under section 2 that participants
82, 52, 101 and 12 scored the highest, one on each component. If we want to investi-
gater these candidates further, we might make a heat map of their scores on all f
our components.
```

```
# creating data frame with the four candidates
best_candidates <- rbind(scores[82,], scores[52,], scores[101,], scores[12,])
```

```
#transforming data structure from dataframe to matrix
best_candidates_m <- as.matrix(best_candidates)
```

```
#creating a heat map with high values as white going towards comparatively low val-
ues at red
```

```
rotate <- function(x) t(apply(x, 2, rev))
image(rotate(best_candidates_m))
```



```
#printing the dataframe in order to see which variables are which (notice that com-
ponent RC3 and RC4 are switched around)
```

```
best_candidates
```

```
##          RC1        RC2        RC4        RC3
## 82  2.4396812 -0.9041585 -3.4089559  0.8656098
## 52 -0.5877093  2.3390467 -2.4108841 -1.3233804
## 101  0.5809308  0.4269010 -0.6746192  2.0449843
## 12   0.3117626 -1.4507434  1.9435316  0.3383456
```

Eyeballing the heat map, it seems as though the third row, candidate 101, has generally the best scores we can test this easily by obtaining the means. These results shoecasing the score in each component are, however, not interpretable 1:1, since the components are weighted. If we want to find out who has the biggest capacity for empathy, we need to take the weights into account.

```
# investigating weights or Loadings
PCA_2$loadings

## 
## Loadings:
##      RC1    RC2    RC4    RC3
## X2     0.263      0.385  0.175
## X3           0.226  0.150
## X4     0.292  0.581  0.266
## X1     0.490      0.376
## X2.1   0.377  0.337      0.150
## X2.2   0.226      0.356 -0.140
## X3.1   0.255  0.454  0.143  0.470
## X2.3   0.289  0.377  0.141 -0.227
## X0     0.395      0.382
## X3.2   0.494      0.290  0.119
## X.1    0.183  0.435  0.313  0.144
## X3.3   0.169  0.614  0.138
## X2.4   0.311  0.397  0.315  0.276
## X1.1   0.272  0.345  0.317
## X3.4   0.161  0.296  0.447
## X4.1   0.604      0.299  0.115
## X3.5   0.351  0.602  0.158  0.113
## X0.1   0.372  0.152  0.546
## X2.5   0.192  0.159  0.567  0.248
## X2.6   -0.141      0.497 -0.146
## X3.6   0.353  0.446  0.289  0.311
## X.1.1  0.134  0.151  0.581
## X1.2   0.639
## X3.7   0.209  0.318 -0.368
## X2.7   0.193  0.612  0.425
## X2.8   0.609      0.454
## X.1.2  0.398  0.444
## X4.2   0.201  0.111  0.291  0.226
## X2.9   0.371  0.483  0.170 -0.118
## X3.8   0.364  0.497 -0.212
## X4.3   0.595  0.195  0.102
## X3.9   0.629  0.239  0.226
## X3.10  0.448  0.100  0.318
## X2.10  0.400      0.170  0.194
## X2.11  0.357  0.459
## X4.4   0.177 -0.140  0.472 -0.190
```

```

## X2.12  0.691  0.313  0.116
## X4.5   0.330  0.595  0.164
## X3.11  0.312  0.489  0.242
## X3.12  0.140  0.431  0.293
## X1.3   0.296  0.355  0.503  0.278
## X2.13  0.155  0.407  0.453
## X2.14  0.194  0.520  0.543  0.118
## X2.15  0.417  0.371  0.566
## X2.16          0.761
## X.1.3           -0.384
## X2.17  0.125  0.610
## X2.18  0.126  0.710 -0.112
## X1.4    0.260 -0.118 -0.379
## X1.5    0.474 -0.103  0.136
## X0.2    0.672  0.155 -0.116
## X3.13  0.302  0.249      0.416
## X2.19          0.644
## X2.20  0.133  0.122      0.809
## X1.6   -0.122          0.339
## X.2    0.634
## X.1.4  0.260  0.135      0.373
## X0.3   0.178 -0.157  0.124  0.645
## X4.6   0.709  0.255
## X4.7   0.671      0.109  0.144
## X4.8   0.777  0.120
## X3.14  0.672  0.322
## X2.21  0.509  0.276  0.311
## X1.7   0.536  0.313  0.206  0.232
## X4.9   0.313
## X4.10  0.552  0.189
## X3.15  0.624  0.356  0.173
## X1.8   0.614  0.184
## X3.16  0.574  0.306  0.210  0.120
## X4.11  0.597      0.197
## X3.17  0.480  0.230  0.178
## X3.18  0.612  0.188      0.231
## X3.19  0.604  0.182  0.273  0.175
##
##                  RC1    RC2    RC4    RC3
## SS loadings    10.687 8.405 6.307 4.109
## Proportion Var  0.146 0.115 0.086 0.056
## Cumulative Var  0.146 0.262 0.348 0.404

# creating a matrix/vector for each component
RC1 <- best_candidates$RC1*10.687
RC2 <- best_candidates$RC2*8.405
RC3 <- best_candidates$RC3*4.109
RC4 <- best_candidates$RC4*6.307

```

```

#creating a dataframe with the weighted scores. We will keep the order RC1, RC2, R
C4, RC3 for comparability.
weighted <- cbind(RC1, RC2, RC4, RC3)

# creating a column in the dataframe for grand score. We will simply calculate gra
nd score from the summed values.

grand_score <- rbind(sum(weighted[1,]), sum(weighted[2,]), sum(weighted[3,]), sum(
weighted[4,]))
grand_score

##          [,1]
## [1,]  0.5299264
## [2,] -7.2643781
## [3,] 13.9445274
## [4,]  4.7864252

# Looking at the grand scores it seems as though candidate 101, column 3, is in fa
ct the best scoring person in the group of four. This does not inform us about whe
ther someone else's grand score may be higher.

# If we want to, say, hire three people who are all relatively empathetic and have
rather similar answers to their questioners - in order to promote easy communicat
ion and cooperation - we could look at the two closest candidates to nr. 101. Acco
rding to the cluster diagram these are nr. 21 and 89. In the following we will exa
mine how these two candidates scored highly across components as nr. 101 succeeded
in doing.

# Find grand scores for nr. 21 and 89

#inserting data from candidate 21 and 89 in separate dataframe
two_candidates <- rbind(scores[21,], scores[89,])

#calculating the weighted scores for each component
RC1w <- two_candidates$RC1*10.687
RC2w <- two_candidates$RC2*8.405
RC3w <- two_candidates$RC3*4.109
RC4w <- two_candidates$RC4*6.307

#binding weighted scores by columns
weighted2 <- cbind(RC1w, RC2w, RC4w, RC3w)

#summing candidates' scores to grand score and insert in new dataframe
grand_score2 <- rbind(sum(weighted2[1,]), sum(weighted2[2,])))
grand_score2

```

```
##      [,1]
## [1,] 12.53523
## [2,] 12.40124
```

We see that the three candidates have very similar scores 101: 13.94, 21: 12.54 and 89: 12.40. This is of course not surprising since we saw their answers were highly similar using the hamilton distance.

If our employer were to ask us, whom we believe would become an empathetic, diligent and well-working team; we would feel pretty confident recommending candidates 21, 89, and 101. This might not be the very best group, since we haven't looked too closely at the candidates without a highscore, but the group certainly has the foundation for successful cooperation.